Graph Representation Learning for Biometric and Biomedical Images Analysis

Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy by

Yanda Meng

October 2022

Abstract

The theme of this dissertation is deep learning with graph-structured representations on the tasks of biomedical and biometric image analysis. Graphs are general representations of signal structure that are crucial in a wide variety of domains, including attributes extraction, features aggregation, and information propagation. Thus, developing machine learning algorithms capable of effectively learning from graph-structured representations is an important computer vision and image analysis topic.

The overarching aim of this dissertation is to structure the representations and computations of neural network-based models in the form of a graph, which allows for improved generalisability of the neural network when learning from data with both implicit and explicit graph structures. This entails several research directions in the task of biometric and biomedical image analysis.

Firstly, I explored the geometric association/consistency between objects' region and boundary via implicit graph data representation learning; and proposed different graph-based novel methods to exploit the underlying complementary spatial relationships. I addressed the rarely discussed issues of the underlying relationship between the region and boundary characteristics in segmentation tasks and data efficiency learning researches. I have applied this new method to five large-scale fundus image datasets for optic disc and cup segmentation in both fully supervised and semi-supervised learning paradigms and five challenging datasets of colonoscopic endoscopy images for polys segmentation in fully supervised mechanism, and the results demonstrated an average 4.1% Dice score improvement over the previous cutting-edge segmentation methods in both learning tasks.

Secondly, I studied the context pattern fusion of various forms of granularity information using inner-domain and cross-domain implicit graph data representation learning. I proposed several novel graph-based methods for hybrid information fusion and addressed the contextual dependency difficulties of multi-granularity features during graph reason-

ing. The methods were applied to the three largest 3D CT-based COVID-19 diagnosis datasets and five challenging 2D image-based crowd counting datasets and achieved superior performance to previous state-of-the-art methods in both tasks. Significantly, my proposed graph model can outperform other compared methods by a large margin in the generalisation ability evaluation experiments of the COVID-19 diagnosis task.

Thirdly, I attempted to model the geometric structure of explicit graph data representations regarding objects' boundaries. I introduced a novel graph-based segmentation paradigm and addressed the difficulties of direct feature learning on objects' boundary locations by previous convolutional neural network-based methods. I applied the proposed methods to several biomedical image segmentation tasks, such as fundus image based optic disc and cup segmentation and ultrasound image based fetal head segmentation tasks. Results on two types of challenging datasets have demonstrated my model's superiority.

Fourthly, I researched the explicit graph data representation learning of dense vertices regression task. I proposed a multi-level aggregated graph convolutional network and addressed the challenges of loss of semantic and spatial information in classic graph convolutional network-based methods. The proposed model was applied to two large 3D face reconstruction datasets; excellent results have achieved, demonstrating my model's reconstruction accuracy and ability to tackle a large number of vertices.

In conclusion, I have proposed several novel methods on the basis of graph-based deep learning with explicit and implicit representations in different biomedical and biometric image analysis tasks. I have demonstrated the robustness and generalisability of the aforementioned proposed methods in various biomedical and biometric image analysis tasks. All of my approaches are anticipated to be widely applicable to real-world applications. Future works can combine the benefits of explicit and implicit graph representation learning and tackle more complicated problems in graph structure, such as protein analysis and drug discovery.

Acknowledgements

Throughout the process of writing my thesis, it brought me a great deal of happiness, particularly as I remembered the wonderful individuals at Liverpool who have been with me, have given me with enormous assistance, and continue to do so. To me, they are brilliant stars in the sky that illuminate my way.

Prof. Yalin Zheng, my respected supervisor, offered me the chance to join this amazing field of research and provided me with significant advise, support, and encouragement during my Ph.D. studies. Thank you for sharing your vast knowledge and vast experience with me on a professional and, when necessary, a more personal level. Thank you for encouraging me to independently explore my scholarly interests. No matter how busy you are, you always make time for every student, post-doc, and staff member, giving them with varying degrees of advice or aid with their work. Your ideals, perspective on science, and philosophy of life have been an inspiration to me.

I would also want to thank Prof. Xiowei Huang, my second adviser and another fantastic individual. It is a delight to work with him, since he is always helpful and provides me with assistance and advice anytime I need it. In addition, he is very generous with his incisive comments, recommendations, and practical advice on experimental design, academic writing, and presentation. In addition to being my boss, he is also my mentor, friend, and role model!

This thesis was funded by studentships from China Science IntelliCloud Technology Co., Ltd. and Remark AI UK LIMITED. Thank you very much, Dr. Xiaoyun Yang, the director of my sponsor, for the financial assistance that has allowed me to concentrate on research. My close partners, Prof. Yitian Zhao, Dr. Alam Uzaman, Dr. Dongxu Gao, Preston Frank, and pals in our YobiLab group, Dr. Xu Chen, Hongrun Zhang, Dr. Wenyue Zhu, Joshua Bridge, and others, all deserve special recognition. Their assistance and support have made my studies and life in Liverpool a joy.

Lastly, I would want to convey my profound gratitude to my parents and wife for their unwavering love and support. Without their support, it is impossible for me to pursue a Ph.D. and study overseas. This doctoral dissertation is devoted completely to them.

List of Publications

This thesis is based on the following publications, * denotes the first two authors contributed equally:

0.1 Peer-reviewed Journal Publications

- **Meng, Y.**, Zhang, H., Zhao, Y., Gao, D., Hamill, B., Patri, G., Peto, T., Madhusudhan, S. and Zheng, Y., 2022. Dual Consistency Enabled Weakly and Semi-Supervised Optic Disc and Cup Segmentation with Dual Adaptive Graph Convolutional Networks. IEEE Transactions on Medical Imaging. (accepted)

- **Meng, Y.**, Zhang, H., Zhao, Y., Yang, X., Qiao, Y., MacCormick, I.J., Huang, X. and Zheng, Y., 2021. Graph-based region and boundary aggregation for biomedical image segmentation. IEEE Transactions on Medical Imaging, 41(3), pp.690-701. DOI: 10.1109/TMI.2021.3123567

- *Preston, F.G., ***Meng, Y.**, Burgess, J., Ferdousi, M., Azmi, S., Petropoulos, I.N., Kaye, S., Malik, R.A., Zheng, Y. and Alam, U., 2022. Artificial intelligence utilising corneal confocal microscopy for the diagnosis of peripheral neuropathy in diabetes mellitus and prediabetes. Diabetologia, 65(3), pp.457-466. DOI: 10.1007/s00125-021-05617-x

- Zhang, H., **Meng, Y.**, Zhao, Y., Qian, X., Qiao, Y., Yang, X. and Zheng, Y., 2022. 3D Human Pose and Shape Reconstruction from Videos via Confidence-Aware Temporal Feature Aggregation. IEEE Transactions on Multimedia. DOI: 10.1109/TMM.2022.3167887

- Bridge, J., **Meng, Y.**, Zhao, Y., Du, Y., Zhao, M., Sun, R. and Zheng, Y., 2020. Introducing the GEV activation function for highly unbalanced data to develop COVID-19 diagnostic models. IEEE journal of Biomedical and Health Informatics, 24(10), pp.2776-2786. DOI: 10.1109/JBHI.2020.3012383

- Preston, F.G., **Meng, Y.**, Zheng, Y., Hsuan, J., Hamill, K.J. and McCormick, A.G., 2022. Informed Consent In Facial Photograph Publishing: A Cross-sectional Pilot Study To Determine The Effectiveness Of Deidentification Methods. Journal of Empirical Research on Human Research Ethics, 17(3), pp.373-381. DOI: 10.1177/15562646221075459
- Alam, U., Anson M., **Meng, Y.**, Preston, F.G., Kirthi, V., Jackson, T.L., Nderitu, P., Cuthbertson, D.J., Malik, R., Zheng, Y., Petropoulos, I.N, 2022. Artificial Intelligence and Corneal Confocal Microscopy: the start of a beautiful relationship. Journal of Clinical Medicine. (accepted)

0.2 Peer-reviewed Conference Publications:

- **Meng, Y.**, Chen, X., Zhang, H., Zhao, Y., Gao, D., Hamill, B., Patri, G., Peto, T., Madhusudhan, S. and Zheng, Y., 2022. Shape-Aware Weakly/Semi-Supervised Optic Disc and Cup Segmentation with Regional/Marginal Consistency. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp.

524-534). Springer, Cham.

- **Meng, Y.**, Ferdousi, M., Petropoulos, I.N., Malik, R.A., Zhao, Y., Alam, U. and Zheng, Y., 2022, May. Diagnosis of Diabetic Neuropathy by Artificial Intelligence using Corneal Confocal Microscopy. In EUROPEAN JOURNAL OF OPHTHALMOLOGY (Vol. 32, No. 1 SUPPL, pp. 11-12).

- **Meng, Y.**, Zhang, H., Gao, D., Zhao, Y., Yang, X., Qian, X., Huang, X. and Zheng, Y., 2021, October. BI-GCN: Boundary-Aware Input-Dependent Graph Convolution Network for biomedical image segmentation. In British Machine Vision Conference.

- **Meng, Y.**, Zhang, H., Zhao, Y., Yang, X., Qian, X., Huang, X. and Zheng, Y., 2021. Spatial uncertainty-aware semi-supervised crowd counting. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 15549-15559). DOI: 10.1109/iccv48922.2021.01526

- **Meng, Y.**, Meng, W., Gao, D., Zhao, Y., Yang, X., Huang, X. and Zheng, Y., 2020, August. Regression of instance boundary by aggregated CNN and GCN. In European Conference on Computer Vision (pp. 190-207). Springer, Cham. DOI: 10.1007/978-3-030-58598-3_12

- **Meng, Y.**, Wei, M., Gao, D., Zhao, Y., Yang, X., Huang, X. and Zheng, Y., 2020, October. CNN-GCN aggregation enabled boundary regression for biomedical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 352-362). Springer, Cham. DOI: 10.1007/978-3-030-59719-1_35

- Zhang, Y., **Meng, Y.**, and Zheng, Y., 2022. Automatically Segment the Left Atrium and Scars from LGE-MRIs Using a Boundary-focused nnU-Net. In LAScarQS 2022 (MICCAI 2022 Workshop, accepted) .

- Zhang, H., **Meng, Y.**, Zhao, Y., Qiao, Y., Yang, X., Coupland, S.E. and Zheng, Y., 2022. DTFD-MIL: Double-Tier Feature Distillation Multiple Instance Learning for Histopathology Whole Slide Image Classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 18802-18812).

- Chen, X., **Meng, Y.**, Zhao, Y., Williams, R., Vallabhaneni, S.R. and Zheng, Y., 2021, September. Learning unsupervised parameter-specific affine transformation for medical images registration. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 24-34). Springer, Cham. DOI: 10.1007/978-3-030-87202-1_3

- Zhang, H., **Meng, Y.**, Qian, X., Yang, X., Coupland, S.E. and Zheng, Y., 2021, August. A regularization term for slide correlation reduction in whole slide image analysis with deep learning. In Medical Imaging with Deep Learning (pp. 842-854). PMLR.

- Deng, K., **Meng, Y.**, Gao, D., Bridge, J., Shen, Y., Lip, G., Zhao, Y. and Zheng, Y., 2021, September. Transbridge: A lightweight transformer for left ventricle segmentation in echocardiography. In International Workshop on Advances in Simplifying Medical Ultrasound (pp. 63-72). Springer, Cham. DOI: 10.1007/978-3-030-87583-1_7

0.3 Under Review/Preprint:

- **Meng, Y.**, Chen, X., Gao, D., Zhao, Y., Yang, X., Qiao, Y., Huang, X. and Zheng, Y., 2022. 3D Dense Face Alignment with Fused Features by Aggregating CNNs and GCNs. arXiv preprint arXiv:2203.04643.

- **Meng, Y.**, Bridge, J., Ren, S., Addison, C., Wang, M., Merritt, C., Franks, S., Mackey, M., Messenger, S., Sun, R., Zhao, Y., Zheng, Y., 2022. Bilateral Adaptive Graph Convolutional Network on CT based COVID-19 Diagnosis with Uncertainty-Aware Consensus-Assisted Multiple Instance Learning. **Minor Revision** in Medical Image Analysis (MedIA).

- **Meng, Y.**, Bridge, J., Zhao, Y., Joddrell, M., Qiao, Y., Yang, X., Huang, X. and Zheng, Y., 2022. Transportation Object Counting with Graph-Based Adaptive Auxiliary Learning. **Major Revision** in IEEE Transactions on Intelligent Transportation Systems (IEEE-TITS).

Contents

List of Figures

List of Tables

List of Abbreviation

This thesis contains the abbreviations as follows:

- CNN - Convolutional Neural Network
- GNN - Graph Neural Network
- GCN - Graph Convolution Network
- OD - Optic Disc
- OC - Optic Cup
- COVID-19 - Coronavirus disease
- CT - Computerised Tomography
- ACM - Activate Contour Model
- MLP - multi-layered perceptrons
- ReLU - Rectified Linear Unit
- AI - Artificial Intelligence
- MRI - Magnetic Resonance Imaging
- SVM - Support Vector Machine
- MRF - Markov Random Field
- GGO - Ground Glass Opacity
- 3DMM - 3D Morphable Models
- PCA - Principle Component Analysis

- NMF - Non-negative Matrix Factorisation
- AEM - Attention Enhancement Module
- BIoU - Boundary Intersection-over-Union
- SDF - Signed Distance Function
- GT - Ground Truth
- B-Acc - Balanced Accuracy
- vCDR - vertical Cup to Disc Ratio
- PM - Probability Map
- DAGCN - Dual Adaptive Graph Convolutional Network
- FAM - Feature Aggregation Module
- mSDF - Modified Signed Distance Function
- UKBB - UK Biobank
- MAE - Mean Absolute Error
- Corr - Pearson's Correlation Coefficients
- FLOPs - Floating-point Operations
- AUROC - Area Under the Receiver Operating Characteristic
- UC-MIL - Uncertainty-aware Consensus-assisted Multiple Instance Learning
- BA-GCN - Bilateral Adaptive Graph-based Neural Network
- CP - Common Pneumonia
- NCP - Novel Coronavirus Pneumonia
- MIL - Multiple Instance Learning
- NAS - Neural Architecture Search
- MF-Net - Multi-Fiber Network
- Grad-CAM - Gradient-weighted Class Activation Mapping
- L1 - Least Absolute Error

- L2 - Least Square Error
- FFB - Feature Fuse Block
- GAP - Global Average Pooling
- BN - Batch Normalization
- RMSE - Root Mean Square Error
- GAME - Grid Average Mean Absolute Error
- IoU - Intersection over Union
- ms - millisecond
- MOT - multiple Objects Tracking
- ARM - Attention Refinement Module
- FCN - Fully Convolution Neural Network
- DGCNet - Dual Graph Convolutional Network
- FH - Fetal Head
- PNCC - Projected Coordinate Code
- DenseNet - Densely Connected Network
- FPN - Feature Pyramid Network
- DenseGCN - Dense Graph Convolution Block
- MFF - Multi-Feature Fitting
- BFM - Basel Face Model

Chapter 1

Introduction

1.1 Overview

A natural way to represent information in structured form is as a graph. A graph is a data structure that represents a collection of items (vertices) and their pairwise associations (edges) [79,82,121]. Graph-structured data is ubiquitous throughout the natural and social sciences, from fundamental physical interactions to emergent structures such as molecules, societal networks, ecosystems, the world wide web, *etc.*. The pervasiveness of this graph-structured representation of information necessitates the development of efficient ways for using and learning to interpret data presented in this structural form. To this end, it is critical to build relational inductive biases in graph representation learning if we create systems capable of learning, reasoning, and generalising from this kind of data to novel and unforeseeable circumstances. Research on graph representation learning has accelerated in recent years, including methods for deep graph embeddings, expansions of convolutional neural networks (*CNNs*) to graph-structured data, and neural message-passing systems inspired by belief propagation. These advancements in graph representation learning have

resulted in new state-of-the-art results in computer vision and image analysis domains, including 3D vision, surveillance, and biomedical image-based segmentation and diagnosis.

The field of graph representation learning is involved with the study and design of operations that make use of the graph structure inherent in data [343, 362]. As a result, graph representations are perfect mathematical objects for defining the structure of networks, and hence the optimum framework for handling network data is graph data processing. Network data is represented as two distinct objects in graph data processing: graph data and graph adjacency matrix. The graph data provides a value to each node, while the graph adjacency matrix records the underlying network structure for usage with the graph data. Indeed, the interaction between the adjacency matrix and the graph data representations enables the development of graph filters, most notably the graph convolution operation [99, 182, 346]. This generalises the conventional graph convolution process. The concept of graph filtering serves as the foundation for the graph neural network (GNN) models [17, 85, 118, 123, 220, 280, 342, 344]. GNNs handle graph representations by using the adjacency matrix's graph structure; they are good at tackling Non-Euclidean data structure with the benefits of defined associations among vertices [182, 346]. This thesis focuses on researching graph representation learning on implicit (hidden graph data) and explicit graph-structured data, such as the datasets that are given to us in the form of entities and their relations to biometric and biomedical image analysis tasks.

1.2 Scope and Research Questions

The works presented in this thesis are on graph representation learning, which is one of the most actively researched areas of machine learning research. Machine learning is concerned with the question of developing systems and algorithms that are capable of learning from data and experience. The learning challenge is often addressed by fitting a model to data

to generalise the learned model to new data or experiences. This thesis's primary goal is to structure the representations and computations of neural network-based models in the form of a graph, allowing for increased generalisability of neural networks while learning from data using both explicit and implicit graph structures on the task of biometric and biomedical image analysis. This thesis is generally structured into two parts: Part 1 explores the implicit structure of graph representation learning in geometry-aware and cross-granularity inductive biases, as applied to biometric and biomedical image analysis tasks. Typically, the overtly graph-structured datasets are not given but rather to create models that infer or exploit latent graph structures in the data or high-dimensional features. Part 2 researches deep graph neural network models for various data-driven regression tasks using explicitly graph-structured data. The graph information in the form of entities and their relations are given. Specifically, the dense geometric data modelling, spatial information supplement and the novel segmentation paradigm via geometric structure are explored.

The following research questions lead to the contributions of this thesis:

Research Question 1: *Can graph neural networks exploit the underlying spatial and semantic relationships between objects' region and boundary characteristics?*

- I proposed two methods based on GNNs and GCNs to address this question. They are introduced in Chapter 3. In brief, I proposed a *GRBNet* [286] for reasoning the spatial association of objects' regions and boundaries with implicit graph data representation learning in the forms of GNNs. My methods explicitly leveraged both region and boundary characteristics during graph-based information propagation. They specifically modelled and reasoned about the boundary-aware region-wise cor-

relations of different classes through learning implicit graph representations, which can manipulate long-range semantic reasoning across various regions with the spatial enhancement along the object's boundary. My models were well-suited to obtain global semantic region information while simultaneously accommodating local spatial boundary characteristics. I have addressed the rarely discussed issues that previous approaches usually overlooked the underlying relationship between the region and boundary characteristics while segmenting biomedical images. Such geometric associations can boost the model's segmentation performance, specifically for boundary accuracy. I evaluated the proposed methods in five large-scale colour fundus image datasets on optic disc and cup segmentation and five large-scale colonoscopic endoscopy images for polys segmentation tasks.

Research Question 2: *Can graph neural networks exploit the geometric consistency within and between objects' region and boundary, and contribute to the semi-supervised learning mechanism?*

- To address the question, I proposed a *DC-GCN* [282] under a semi-supervised learning mechanism to exploit the boundary and region's inherent geometric consistency via an implicit graph data representation learning. The enforced consistency on regional and marginal predictions leads the learned model to a generalisable characteristic learning via leveraging a large amount of unlabeled data. Specifically, the consistent regional regularisation between different formats of region graph representations advanced semi-supervised learning and exploited the inherent geometric consistency in many unlabeled data. I demonstrated robustness and generalisability of the proposed networks in several biomedical image analysis tasks, such as optical

disc and cup segmentation and vertical cup to disc ratio estimation of colour fundus images, under semi-supervised learning mechanisms, respectively. The performance is superior to previous cutting-edge semi-supervised segmentation methods. They are introduced in Chapter 4.

Research Question 3: *Can graph neural networks discover and build effective context patterns of cross-granularity multi-domain representations?*

- To address the question, I proposed a *BAGCN* [271] for fusing various forms of granularity information using implicit graph data representations in a three-dimensional (3D) chest computed tomography (CT) based COVID-19 diagnosis task. They are introduced in Chapter 5. Specifically, my proposed method represented inner-domain and cross-domain multi-granularity features as vertices in the proposed graph. In this way, the contextual dependency and properties among vertices with multi-granularity are exploited during graph reasoning. On the other hand, such cross-granularity information can supplement each other vertices through graph-based propagation for the aimed task. For example, *BAGCN* aggregated information and exchanged messages between bilateral cross-domain vertices in terms of 2-dimensional (2D) and 3D levels. This helps the proposed method consider features at both levels of the given volume data when making inferences, thus improving the proposed model's diagnosis performance. Apart from that, *BAGCN* addressed the shortcomings of hand-crafted or randomly initialised graph structures by prior GCN-based approaches. As a result of this challenge, the model tends to develop a specific context pattern that is less generalisable to other domains of similar data. *BAGCN*, on the other hand, adaptively learned the graph structure and edge relationships between vertices via

initialising adjacency matrix according to the cross-granularity vertices' own properties. The data-dependent way of vertices reasoning enables the *BAGCN* to learn an input-related long-range context pattern, which improves the generalisation ability of graph representation learning. I demonstrated the resilience and generalisability of the proposed methods in the three largest CT-based COVID-19 diagnosis datasets. Specifically, my model outperformed other state-of-the-art methods by a large margin in evaluating generalisation ability with external test data.

Research Question 4: *Can graph neural networks adaptively exploit the supplementary information of different auxiliary tasks and contribute to the main task in the multi-task based learning mechanism?*

- To address the question, I proposed an adaptive auxiliary task learning-based approach for supplementing the complementary information of different auxiliary tasks to the main task of crowd counting with biometric images. I proposed an AAL-Net [272] that consisted of both CNN and GCN for feature extraction and feature reasoning among different domains of auxiliary tasks. My approach gained enriched contextual information by iteratively and hierarchically fusing the features across different task branches of the adaptive CNN backbone. The framework paid special attention to the objects' spatial locations and varied density levels, informed by crowd segmentation and density level segmentation auxiliary tasks via the proposed adaptive GCN module. In other words, the proposed adaptive GCN module can tackle different auxiliary tasks information and supply it to the main task through graph-based information propagation and update. In other words, I proposed a new vertices-edges connection paradigm in the graph that contains rich auxiliary tasks in-

formation. The method is introduced in Chapter 6. Experiments on five challenging multi-domain datasets demonstrate that my method is superior to the state-of-the-art auxiliary task learning-based counting methods.

Research Question 5: *Can deep neural networks exploit the geometric structure of explicit graph representations and directly learn features on objects' boundary locations while tackling segmentation tasks?*

- To address the question, I introduced a novel image segmentation paradigm [276] and addressed the difficulties of direct features learning on objects' boundary locations by previous CNN based methods while tackling segmentation tasks. The model is introduced in Chapter 7. I did not follow the traditional segmentation way of dense pixel-wise classification or Activate Contour Model (ACM) based iteration methods but proposed directly regressing the objects' boundary location via a graph-based learning mechanism. I explicitly defined the vertices and the edges within the proposed graph along the objects' boundaries. On the other hand, loss of spatial information and limitation of CNN's receptive field makes direct feature learning on the objects' boundary location difficult. In contrast, I directly exploited the GCNs' long-range information propagation ability on objects' boundary locations to address the challenge. Such a straightforward and intuitive segmentation method leads to a new paradigm of biomedical image segmentation tasks because boundary information and accuracy are more critical than pixel-wise converge in the biomedical image segmentation tasks. Experiments demonstrate that my method achieved comparable segmentation performance with state-of-the-art approaches but is able to give more interpretative and accurate boundary predictions on the segmentation of fetal head

in ultrasound images and segmentation of optic disc and optic cup in colour fundus images.

Research Question 6: *Can deep neural network based models efficiently tackle large-scale nodes' (vertices) location tasks with graph-structured datasets?*

- My main contribution to addressing this question is to propose a multi-level multi-stage aggregation mechanism that seamlessly combines classic CNNs and Graph Convolution networks (GCNs) [182]. Such aggregation mechanisms contributed to supplementing spatial and semantic information loss due to the difficulty of dense vertices' information gain. Specifically, I proposed a *MA-GCN* [273], for dense vertices regression of explicit graph data representation learning task. They are introduced in Chapter 8. *MA-GCN* is a form of graph neural network that performs parameterised message-passing operations inside a graph. It is represented as a first-order approximation to spectral graph convolutions. Specifically, multi-level and multi-stage aggregation paradigms are proposed for sufficient information gain and feature propagation of GCNs. They efficiently address the loss of spatial information challenges in single CNNs or GCNs based methods on dense vertices regression tasks. The aggregation paradigm benefits the advantages of both CNNs and GCNs in terms of semantic and spatial feature extraction ability, resulting in superior performance. I demonstrated its state-of-the-art performance and outperformed previous cutting-edge methods by a large margin in graph-structured datasets, such as 2D-3D face mesh reconstruction.

Chapter 2

Background

Deep learning models are essentially deep artificial neural networks. This chapter aims to provide a formal introduction and definition of the concepts, methodologies, and architectures of deep learning.

2.1 Fundamentals of Deep Learning

2.1.1 Neural Network Neurons

Neural networks are a learning algorithm that serves as the foundation for most deep learning methods. A neural network is comprised of neurons or units with some activation and parameters $\Theta = \{W, b\}$, where W is a set of weights and b a set of biases. The activation represents a linear combination of the input x to the neuron and the parameters, followed by an element-wise non-linearity $\sigma(\cdot)$, referred to as a transfer function:

$$a = \sigma(W^T x + b). \tag{2.1}$$

Typical transfer functions for traditional neural networks are the sigmoid and hyperbolic tangent function. The multi-layered perceptrons (MLPs), the most well-known of the classic neural networks, have several layers of the following transformations:

$$f(x;,\Theta) = \sigma(W^L\sigma(W^{L-1}...\sigma(W^0x + b^0) + b^{L-1}) + b^L). \tag{2.2}$$

Here, W^n is a matrix containing rows w_k, associated with activation k in the output. The symbol n represents the number of the current layer, with L indicating the ultimate layer. Typically, layers between the input and output are known as 'hidden' layers. When a neural network contains multiple hidden layers, it is often referred to as a 'deep' neural network, thus the phrase 'deep learning'. Typically, the activation of the final layer of the network is mapped to a distribution over classes $P(y|x;\Theta)$ through an activation function, such as *softmax*.

2.1.2 Non-linear Activation Function

There are a variety of nonlinear activation function types. The most prevalent are detailed below. Figure. 2.1 depicts the S-shaped appearance of the *Sigmoid* activation function. The range of *sigmoid* outputs (prediction probability) is between 0 and 1. The definition of *Sigmoid* activation function is as follows:

$$Sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{2.3}$$

The activation function of *Tanh* is also represented in Figure. 2.1 as an S-shape. The *Tanh* function has a range of between and (-1 to 1). The *Tanh* function has the virtue of

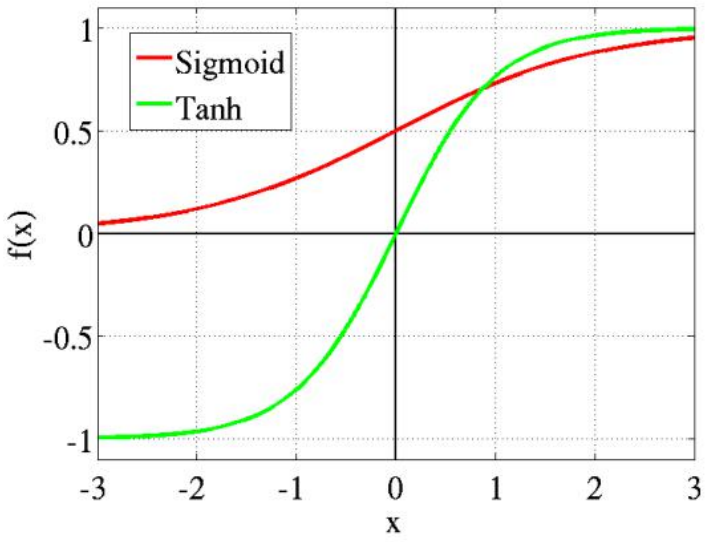


Figure 2.1: Representation of the *Sigmoid* and *Tanh* activation function.

mapping negative inputs substantially negative and zero inputs near zero.

$$Tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.4}$$

Both the *Sigmoid* and *Tanh* activation functions are mostly used for classifying between two groups. The *softmax* function is a comparable activation function that is one of the most often utilised activation functions in machine learning. *Softmax* is a more generalised logistic activation function used for multi-class problems.

Rectified Linear Unit ($ReLU$) [300] was recently proposed to tackle the problem mentioned above, shown in Figure. 2.2. $ReLU$ is defined by the formula below,

$$ReLU(x) = max\{0, x\} \tag{2.5}$$

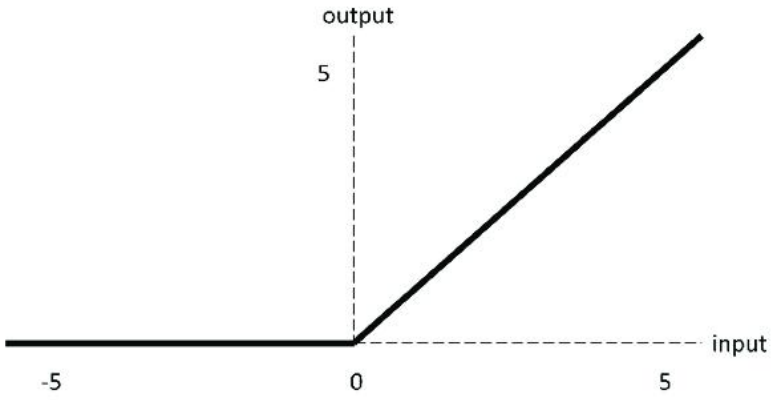


Figure 2.2: Representation of the *ReLU* activation function.

ReLU returns the input value directly, or 0.0 if the input is less than 0.0, in contrast to the *Tanh* and *sigmoid* activation functions, which need an exponential computation. An important advantage of the *ReLU* is that it may output a true zero value, unlike the *sigmoid* activation function, which learns to approximate a zero output, *e.g.* a number extremely near to 0.0, but not a true zero.

2.1.3 Convolution Layers

The convolutional layer [195] is the fundamental building element for CNNs, although it is constrained by sluggish CPUs. The convolutional layer may be parameterized using filters convolved across the image's width and height. The neurons in the convolutional layer are connected to local input areas and calculate their outputs depending on these local regions.

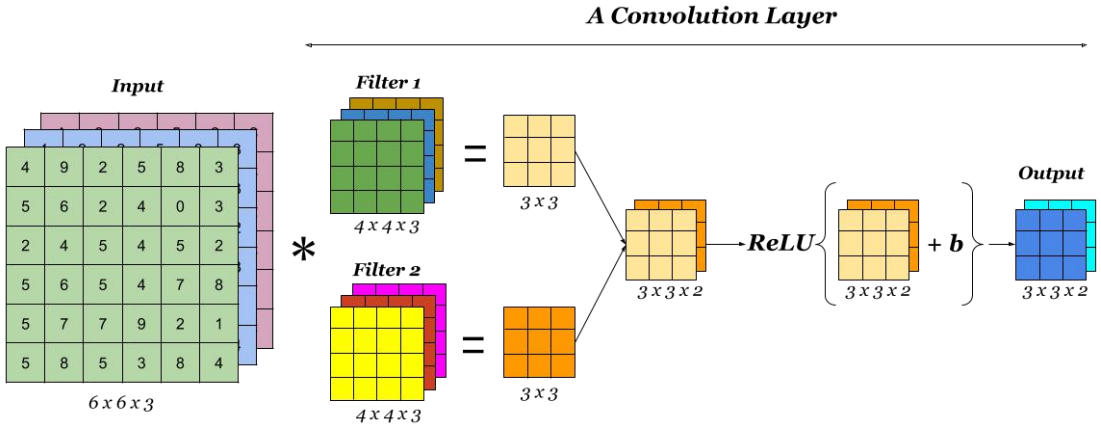


Figure 2.3: Diagram shows convolution operation with R$ReLU$ as the activation function.

Each filter's output is known as a feature map (see Figure. 2.3). Convolution is important for extracting information from pictures as feature maps. Transposed convolution [98, 476], dilated convolution/atrous convolution [59], and depth-wise convolution [78] are the variants of convolution.

2.1.4 Pooling Layers

The pooling layer is a spatial downsampling procedure performed after the convolution layer (see Figure. 2.4). The feature maps from convolution layers are subsampled, or pooled, with sections that do not overlap (windows). The non-overlapping windows on the feature map are relocated. The objective of the pooling layer of a CNN is to reduce the number of trainable parameters, the network's overall calculation time, and to

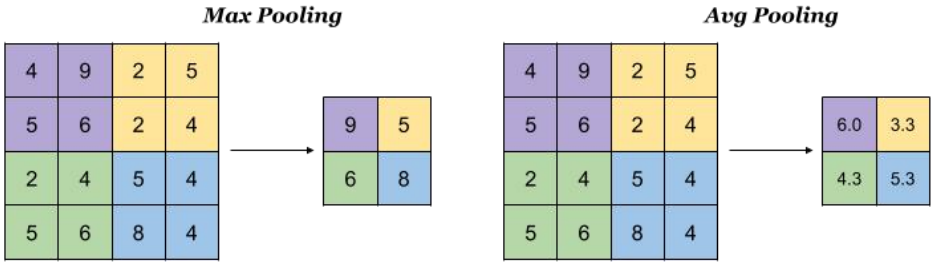


Figure 2.4: Diagram shows max and average pooling operations.

achieve translation invariance [324]. The size of windows is an empirically-definable hyper-parameter. Two main forms of pooling procedures exist. 1) max pooling: the maximum value of all pixels in the batch is chosen; 2) average pooling: the average value of all pixels in the batch is chosen.

2.2 Graph Neural Networks

Graph neural networks (GNNs) are a class of deep neural network models suited for processing graph-structured data and are of central importance to the topics covered in this thesis. The architecture of a GNN is structured according to a graph $G = (V, E)$ with a set of nodes V and a set of edges E. Generally, nodes are identified by a unique index $i \in V$ ranging from 1 to $|V|$, and directed edges from i to j are represented by an ordered

pair of nodes $(i, j) \in V \times V$. For undirected graphs, both (i, j) and (j, i) are in E if nodes i and j are connected. A GNN takes as input an instance of a graph G (*e.g.*, a sample from a dataset of many graphs), where nodes are associated with feature vectors x_i and edges may also be associated with feature vectors $x_{i,j}$. We denote hidden representations in the neural network for nodes and edges with h_i and $h_{i,j}$, respectively. As the initial node representation, we may set $h_i = x_i$. The structure of the graph G then dictates the message passing updates that are conducted sequentially to produce updated node representations h'_j and edge representations $h_{i,j}$:

$$h_{i,j} = f_{edge}(h_i, h_j, x_{i,j}), \tag{2.6}$$

$$h'_i = f_{node}(h_i, \sum_{j \in N_i} h_{j,i}, x_i). \tag{2.7}$$

N_i is the set of neighbors with an incoming edge to node i. f_{edge} and f_{node} typically are small MLPs with two or three layers which take a concatenation of the function arguments as input, but alternative options are conceivable. Multiple message passing updates can be chained by setting from h'_i to h_i after each node update given by Eq. 2.7. The parameters of f_{edge} and f_{node} do not need to be shared across changes involving message passing.

This form of GNN was introduced by *Gilmer et al.* [118] under the name *message passing neural network*, in an effort to generalize and unify earlier models, such as the *graph convolutional network* (GCN) [182] or the *interaction network* [23]. We can utilize this GNN as a function approximator on graph-based tasks trained with backpropagation, *e.g.*, in the context of graph classification by aggregating the final outputs of the GNN into a global representation $h_g = \sum_{i \in V} h_i$. For a recent study of the expressive power of this class of models in the context of function approximation, see [71].

The first GNN model is typically attributed to [123], who originated the term *graph neural network*. Their model contains many of the core concepts found in the GNN definition above but was formulated as a recurrent neural network, trained by a version of backpropagation through time [438] that demanded that message passing updates of the GNN model are a *contraction mapping*. This form of GNN further did not learn an explicit edge representation $h_{i,j}$ and the update function for a node i was conditioned on neighboring states h_j with $j \in N_i$ only (in addition to initial node feature vector x_i). *Scarselli et al.* [344] enhanced this formulation by also configuring the message passing update based on initial edge characteristics $x_{i,j}$.

The GNN definitions in Eq. 2.6 and Eq. 2.7 are not exhaustive, but they do correspond to the models analysed in this thesis. Recent extensions include *graph networks* [24], which provides a global state and update mechanism, and graph G-invariant networks [267]. Other recent related models and GNN variants can be cast as a special case of the message passing definition above, such as the *transformer* architecture [399]. Lastly, there exists a class of *spectral* methods for learning on graphs [42, 85, 147, 182], which promote the development of GNN. Among them, [182] has been widely adopted as the baseline model in the task of computer vision and image analysis.

2.2.1 Graph Convolutional Network

The start point of building a graph-based neural network classifier is the notion of a spectral graph convolution [85]. A spectral convolution on a graph can be understood as parameterized filtering operation that takes into account both node features and the structure of a graph.

The spectral convolutions on graphs defined as the multiplication of a signal $x \in \mathbb{R}^N$

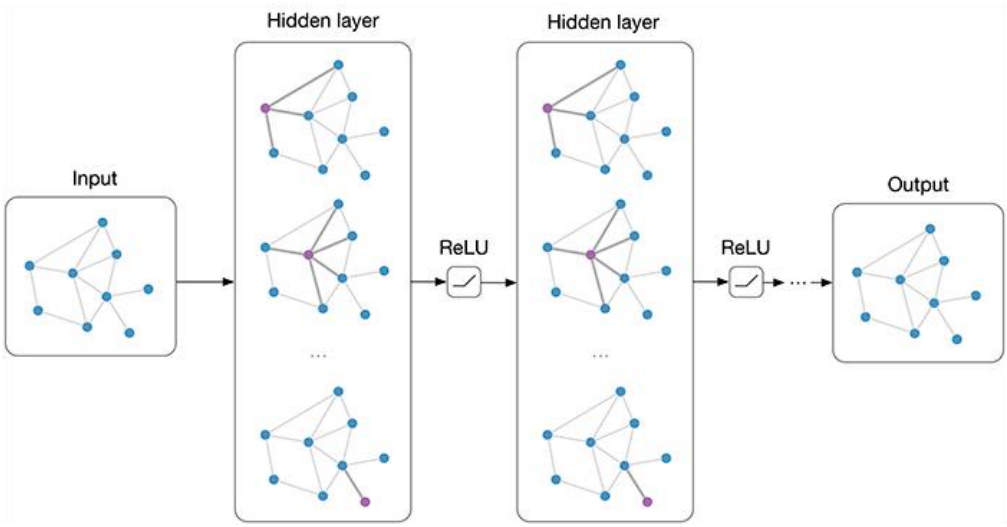


Figure 2.5: Diagram shows an example of Multi-layer Graph Convolutional Network (GCN).

with a filter $g_\theta = diag(\theta)$ parameterized by $\theta \in \mathbb{R}^N$ in the Fourier domain, *i.e.*:

$$g_\theta(x) = U g_\theta U^T x, \tag{2.8}$$

where U is the matrix of eigenvectors of the normalized graph Laplacian $L = L_N - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$, with a diagonal matrix of its eigenvalues Λ and $U^T x$ being the graph Fourier transform of x. The g_θ can be understood as a function of the eigenvalues of L, *i.e.*, $g_\theta(\Lambda)$. Eq. 2.8 is computationally expensive, as multiplication with the eigenvector matrix U is $\mathcal{O}(N^2)$. Furthermore, computing the eigendecomposition of L in the first place might be prohibitively expensive for large graphs.

To circumvent this problem, one can approximate $g_\theta(\lambda)$ by a truncated polynomial

expansion, *e.g.* using a monomial basis or, as proposed in [85], in terms of Chebyshev polynomials $T_k(x)$ up to *K-th* order:

$$g_{\theta'}(\Lambda) \approx \sum_{k=0}^{K} \theta'_k T_k(\tilde{\Lambda}), \tag{2.9}$$

with re-scaled $\tilde{\Lambda} = \frac{2}{\lambda max}\Lambda - I_N$. Λ_{max} denotes the largest eigenvalue of L. $\theta' \in \mathbb{R}^K$ is now a vector of Chebyshev coefficients. The Chebyshev polynomials are recursively defined as $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$, with $T_0(x) = 1$ and $T_1(x) = x$. The reader is referred to [85] for more details.

Going back to the definition of a convolution of a signal x with a filter $g_{\theta'}$, now the spectral graph convolution can be defined as:

$$g_{\theta'}(x) \approx \sum_{k=0}^{K} \theta'_k T_k(\tilde{L})x, \tag{2.10}$$

with $\tilde{L} = \frac{2}{\lambda_{max}}L - I_N$; as can easily be verified by noticing that $(U\Lambda U^T)^k = U\Lambda^k U^T$. Note that this expression is now *K-localized* since it is a *K-th* order polynomial in the Laplacian, *i.e.* it depends only on nodes that are at maximum K steps away from the central node (*K-th* order neighborhood), and hence it can be seens as a spatial graph filter. The complexity of evaluating Eq.2.10 is $\mathcal{O}(|\varepsilon|)$, *i.e.* linear in the number of edges.

In [182], *Kipf et al.* further simplified the graph convolution as $g_\theta = \theta(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}})$, where $\hat{A} = A + I$, $\hat{D}_{ii} = \sum_j \hat{A}_{ij}$, and θ is the only Chebyshev coefficient left. The corresponding graph *Laplacian* adjacency matrix $\hat{A}$ is hand-crafted, which leads the model to learn a specific long range context pattern rather than the input-related one [215].

2.3 Application Domain

Here we introduce the related techniques and applications to the high-level themes of our work, which will be discussed in detail in the corresponding chapters.

2.3.1 Biomedical Image Analysis

Medical imaging is often an integral aspect of the medical diagnosis and treatment process. Typically, a radiologist reviews the acquired medical images and writes a summarising report of their findings. The referring physician formulates a diagnosis and treatment plan based on the images and the radiologist's report. Often, medical imaging is ordered as part of a patient's post-therapy follow-up to confirm treatment effectiveness. In addition, images are increasingly used for surgical planning and real-time imaging during invasive surgical operations. As a specific example, consider "radiology challenge" [325, 339]. With the advancement of technology related to the image capture process during the last decade, the speed and resolution of imaging equipment have increased. For example, in 1990, a CT scanner might acquire 50–100 slices in each case; however, modern CT scanners might acquire 1000–2500 slices per case. A single whole slide digital pathology image corresponding to a single prostate biopsy core can easily occupy 10GB of space at 40x magnification. Overall, there are billions of medical imaging studies undertaken annually worldwide, and this number is growing. Most interpretations of medical images are performed by physicians and, in particular, by radiologists. However, image interpretation by humans is limited due to human subjectivity, high inter-interpreter variances, and weariness. Case-reviewing radiologists have limited time to examine an ever-increasing volume of images, resulting in missed diagnoses, lengthy turnaround times, and a dearth of numerical data or quantification. This severely hinders the medical community's capacity to progress towards more evidence-based, customised treatment. AI tools such as deep learning technology can

provide support to physicians by automating image analysis, leading to what we can term "Computational Radiology" [36, 391]. Among the automated tools that can be developed are detection of pathological findings, quantification of disease extent, characterisation of pathologies (*e.g.*, into benign vs malignant), and assorted software tools that can be broadly characterised as decision support. This technology can also extend physicians' capabilities to include the characterisation of three-dimensional and time-varying events, which are often not included in today's radiological reports because of both limited time and limited visualisation and quantification tools.

Medical image segmentation aims to make anatomical or pathological structure changes more evident in images; it often plays a crucial role in computer-assisted diagnosis and smart medicine due to the vast increase in diagnostic efficiency and accuracy. Popular medical image segmentation tasks consist of liver and liver-tumor segmentation [214, 216], brain and brain-tumor segmentation [287, 313], optic disc segmentation [274, 282], cell segmentation [337, 479], lung segmentation, pulmonary nodules [271, 426], cardiac image segmentation [57, 441], colorectal tumor or polyps segmentation [280, 286], fetal head segmentation [276, 278] etc. With the development and widespread adoption of medical imaging techniques, X-ray, CT, Magnetic Resonance Imaging (MRI), ultrasound, colour fundus, and endoscopy images have become important image assisted means to assist clinicians in diagnosing diseases, evaluating prognosis, and to plan operations in medical institutions. In practical applications, although these ways of imaging have advantages as well as disadvantages, they are useful for the medical assessment of different parts of the human body.

To aid clinicians make an accurate diagnosis, it is necessary to segment some crucial objects in medical images and extract features from segmented areas. Early approaches to medical image segmentation often rely on edge detection, template matching techniques,

statistical shape models, active contours, machine learning, *etc.* *Zhao et al.* [473] proposed a new mathematical morphology edge detection algorithm for lung CT images. *Lalonde et al.* [192] applied Hausdorff-based template matching to disc inspection, and *Chen et al.* [63] also employed template matching to perform ventricular segmentation in brain CT images. *Tsai et al.* [393] proposed a shape-based approach using horizontal sets for 2D segmentation of cardiac MRI images and 3D segmentation of prostate MRI images. *Li et al.* [203] used the activity profile model to segment liver-tumors from abdominal CT images, while *Li et al.* [212] proposed a framework for medical body data segmentation by combining level sets and support vector machines (SVMs). *Held et al.* [146] applied Markov random fields (MRF) to brain MRI image segmentation.

Even though many techniques have been described and are effective in specific conditions, image segmentation is still one of the most challenging topics in the field of computer vision due to the difficulty of feature representation. In particular, it is more challenging to extract discriminating features from medical images than standard RGB images since the former often suffers from problems of blurring, noise, low contrast, *etc.*. Due to the rapid development of deep learning techniques [189], medical image segmentation will no longer require hand-crafted features, and CNNs or GNNs successfully achieve hierarchical feature representation of images, making it the most popular research topic in image processing and computer vision. As CNNs or GNNs used for feature learning are insensitive to image noise, blur, contrast, *etc.*, they provide excellent segmentation results for medical images.

It is worth mentioning that there are currently two categories of image segmentation tasks, semantic segmentation and instance segmentation. Image semantic segmentation is a pixel-level classification that assigns a corresponding category to each pixel in an image. Compared to semantic segmentation, instance segmentation needs to achieve pixel-level classification and distinguish instances based on specific categories. There are few reports

on instance segmentation in medical image segmentation since each organ or tissue is quite different. In this thesis, we report our proposed model for both segmentation tasks.

Medical image diagnosis usually focuses on classifying the medical image into two or more classes. There are different types of classification tasks. For example, exam classification and object classification. Exam classification aims to categorise an image of a diagnostic exam as absent/present or typical/abnormal illness. As for object classification, the goal is to classify an entity that has been pre-identified (such as a Chest CT nodule) into one of two or more classes. This thesis focuses on the object classification task of COVID-19 diagnosis among common pneumonia and healthy control cases in chest CT. . For many of these tasks, both local information on radiographic diagnosis characteristics (GGO) appearance and global contextual information on GGO location are required for accurate classification. This combination is typically not possible in generic deep learning architectures. Previous COVID-19 related deep learning methods rely on the extracted features from either 2D or 3D level, for example, 2D CNN models on the selected 2D CT images ([101, 113, 149, 230, 395, 415, 433, 447, 504]) or 3D CNN models for CT volumes ([145,428,511]). This thesis presents a graph-based model that can simultaneously exploit the features on 2D and 3D levels of chest CT images.

2.3.2 Crowd Counting

Accurately estimating the number of objects in a single image is a challenging yet meaningful task and has been applied in many applications, including urban planning and public safety. In the various object counting tasks, crowd counting is particularly prominent due to its significance to social security and development. Fortunately, the development of the techniques for crowd counting can be generalised to other related fields, such as vehicle counting and environment survey, without taking their characteristics into account.

Therefore, many researchers are devoted to crowd counting, and many excellent works of literature and works have spurred out.

Over the past few decades, an increasing number of research communities, have considered the problem of object counting as their mainly research direction, as a consequence, many works have been published to count the number of objects in images or videos across wide variety of domains such as crowding counting [131, 231, 284, 306, 326], cell microscopy [269], animals [16], vehicles [127, 478, 487] and leaves [3]. In all these domains, crowd counting is of paramount importance, and it is crucial to building a more high-level cognitive ability in some crowd scenarios, such as crowd analysis [353, 503], and video surveillance [52]. As the world's population rises and urbanisation grows, crowds assemble rapidly in several settings, including parades, concerts, and stadiums. In these scenarios, crowd counting plays an indispensable role in social safety and control management. Considering the specific importance of crowd counting aforementioned, more and more researchers have attempted to design various sophisticated projects to address the problem of crowd counting. Especially in the last half decades, with deep learning, CNNs or GNNs based models have been overwhelmingly dominated in various computer vision tasks, including crowd counting. Although different tasks have unique attributes, standard features such as structural features and distribution patterns exist. Fortunately, the techniques for crowd counting can be extended to some other fields with specific tools.

The various approaches for crowd counting are mainly divided into four categories: detection-based, regression-based, density estimation, and, more recently CNNs or GNNs-based density estimation approaches. Early works [200, 208, 387] on crowd counting use detection-based approaches. These approaches usually apply a person or head detector via a sliding window on an image. Recently many extraordinary object detectors such as R-CNN [119, 141, 329], YOLO [327], and SSD [235] have been presented, which may

perform dramatic detection accuracy in the sparse scenes. However, they will present unsatisfactory results when en-countered the situation of occlusion and background clutter in extremely dense crowds. To reduce the above problems, some works [52,53,155] introduce regression-based methods which directly learn the mapping from an image patch to the count. They usually first extract global features [58] (texture, gradient, edge features), or local features [340] (SIFT [249], LBP [305], HOG [83], GLCM [139]). Then some regression techniques such as linear regression [309], and Gaussian mixture regression [385] is used to learn a mapping function to the crowd counting. These methods successfully deal with the problems of occlusion and background clutter, but they always ignore spatial information. Therefore, *Lemptisky et al.* [201] first adopt a density estimation based method by learning a linear mapping between local features and corresponding density maps. To reduce the difficulty of learning a linear mapping, [314] proposes a non-linear mapping, random forest regression, which obtains satisfactory performance by introducing a crowdedness prior and using it to train two different forests. Besides, this method needs less memory to store the forest. These methods consider spatial information, but they only use traditional hand-crafted features to extract low-level information, which cannot guide the high-quality density map to estimate more accurate counting. Recently, benefiting from the powerful feature representation of CNNs or GNNs, more researchers utilise them to improve density estimation. Earlier heuristic models typically leverage basic CNNs to predict the density of the crowds [405], which obtain significant improvement compared with traditional hand-crafted features. In this thesis, we focused on the GNN based density map regression direction. We proposed a model that can utilise multi-domains of auxiliary information from auxiliary tasks with the help of graph-based information propagation and message passing mechanism.

2.3.3 2D-3D Human Face Reconstruction

Facial analysis has been exploited extensively in variety of applications, including human-computer interaction [137, 482], security [46, 172], animation [435, 436], and even health [51, 148, 321, 377]. A recent trend in this discipline is to incorporate 3D data to overcome some of the inherent limitations of the ubiquitous 2D facial analysis. Due to the 3D aspect of the face, a 2D image cannot adequately depict its geometry since it compresses one dimension. Furthermore, 3D imaging represents the facial geometry independent of posture and lighting, which are two of the problematic aspects of 2D imaging. The advantages brought by 3D facial analysis systems come at the expense of a more sophisticated imaging procedure, which can often restrict their scope. Typically, 3D facial information is usually captured via stereo-vision systems [4, 25, 26], 3D laser scanners [198] (*e.g.* NextEngine and Cyberware), or RGB-D cameras (such as Kinect). The first two methods capture high-quality facial scans but need calm surroundings and costly equipment. In contrast, RGB-D cameras are cheaper and easier to use, but the resulting scans are of limited quality [177, 462].

An attractive approach to acquiring a 3D scan of the face is to estimate its geometry from an uncalibrated 2D image [35, 130, 392]. This 3D-from-2D reconstruction alternative aims to combine the ease of obtaining 2D images with the benefit of a 3D representation of facial geometry. Even while this approach seems appealing, it is intrinsically flawed: the unique facial geometry, the pose of the head and its texture (including illumination and colour) have to be reconstructed from a single picture, resulting in an underdetermined problem. Consequently, there are ambiguities in the solution of the 3D-from-2D face reconstruction since a single 2D picture can be generated from different 3D faces, and it is hard to determine which one corresponds to the actual geometry. Recent methodological advancements have helped to achieve remarkably convincing reconstructions, allowing it

possible to use 3D-from-2D face reconstruction in a wide variety of fields [1, 49, 80]. Some methods are even able to recover local details, such as wrinkles, or to reconstruct the 3D face from images viewed under extreme conditions, such as occlusions or large head poses [35, 130]. Incorporating past information to clarify ambiguities in the solutions is crucial to the success of 3D-from-2D reconstruction approaches. Three methods for incorporating this previous knowledge have emerged in the last decade: statistical model fitting, photometric stereo, and deep learning. In the first method, previous information is represented in a 3D face model constructed from a collection of 3D facial scans and fitted to the input photos. In the second, photometric stereo techniques are used with a 3D template face or 3D facial model to estimate the facial surface normals. This technique often employs information from numerous photos, which further restricts the issue. In the third method, the 2D-to-3D mapping is accomplished using deep neural networks that, given the proper training data, can acquire the requisite priors to connect the geometry and appearance of faces.

The most widespread statistical models of 3D faces are the 3D Morphable Models (3DMM), which were introduced to the community by *Blanz and Vetter* [32]. A 3DMM consists of a shape (*i.e.*, geometry) model and, optionally, an albedo (*a.k.a* texture or colour) model, separately constructed using principal component analysis (PCA). Let M be the number of 3D faces in the training set and N the number of vertices in each mesh. Let $x = (x_1, y_1, z_1, ..., x_N, y_N, z_N) \in \mathbb{R}^{3N}$ be the shape vector of a mesh, and $c = (R_1, G_1, B_1, ..., R_N, G_N, B_N) \in [0, 1]^{3N}$ the albedo vector that contains the R (red), G (green), and B (blue) values of the RGB colour model for each of the N vertices. The idea behind the 3DMM is that, if the set of 3D faces is sufficiently large, one can express any new textured shape as a linear combination of the shapes and textures of the training 3D

faces:

$$x_{new} = \sum_{m=1}^{M} a_m x_m, \tag{2.11}$$

$$c_{new} = \sum_{m=1}^{M} b_m c_m, \tag{2.12}$$

with $a_m, b_m \in \mathbb{R}, \forall m = 1, ..., M$. Thus, it can parametrise any new face by its shape $x_{new} = (a_1, ..., a_M)^T$ and albedo $c_{new} = (b_1, ..., b_M)^T$. However, this parametrisation gets more complicated when the number of shapes in the training set M is large. PCA helps compressing the data, performing a basis transformation to an orthogonal coordinate system defined by the eigenvectors ϕ_i and ψ_i of the covariance matrices computed over the shapes and albedos in the training set. In the orthogonal basis given by PCA,

$$x_{new} = \bar{x} + \sum_{i=1}^{M-1} \alpha_i \phi_i = \bar{x} + \Phi_\alpha, \tag{2.13}$$

$$c_{new} = \bar{x} + \sum_{i=1}^{M-1} \beta_i \psi_i = \bar{c} + \Psi_\beta, \tag{2.14}$$

with $\bar{x} = \frac{1}{M} \sum_{m=1}^{M} x_m$ the mean shape, $\alpha = (\alpha_1, ..., \alpha_{M-1})^T \in \mathbb{R}^{M-1}$ the shape parameters of the model, and $\Phi = (\phi_1, ..., \phi_{M-1}) \in \mathbb{R}^{3N \times (M-1)}$ the shape basis matrix of the model; $\bar{c}, \beta, \Psi$ are analogously defined for the texture. The probability of the shape parameters $p(a)$ is given by:

$$p(\alpha) \propto exp[-\frac{1}{2} \sum_{i=1}^{M-1} (\frac{\alpha_i}{\sigma_{\alpha_i}})^2] \tag{2.15}$$

where $\sigma_{\alpha_i}^2$ are the eigenvalues of the corresponding eigenvectors ϕ_i. The probability of the albedo parameters $p(\beta)$ is defined analogously.

Finally, the shape model of the 3DMM is defined by the mean shape, $\bar{x}$, the eigenvectors

of the shape covariance matrix, Φ, and the corresponding eigenvalues, $\{\sigma_{\alpha_i}^2\}_{i=1}^{M-1}$. Similarly, the albedo model is given by $\bar{c}, \Psi$ and $\{\sigma_{\beta_i}^2\}_{i=1}^{M-1}$.

However, some of the variation modes (eigenvalues $\sigma_{\alpha_i}^2, \sigma_{\beta_i}^2$), thus they are dispensable. Keeping only the directions that represent model of the variance of the taining set can reduce the dimension of the data, which is very useful when M us large. Assuming the eigenvalues $\sigma_{xi_i}^2$ (denoting either $\sigma_{\alpha_i}^2$ or $\sigma_{\beta_i}^2$) are ordered in decending order, the $\tilde{M}$ first eigenvectors with higher eigenvalues.

Even though most extant 3D statistical face models are based on the process described above, various researchers have identified two problems with this approach. First, PCA calculates basis vectors that generically describe the input data; hence, delicate information, like wrinkles, is not captured, making it difficult to recreate face characteristics by fitting a 3DMM model. Some works [43, 107, 169, 257, 301] emphasised the significance of modelling facial deformations locally and suggested various approaches to do so. *Neumann et al.* [301] and *Ferrari et al.* [107] proposed to decompose the matrix of the training shapes by imposing sparse components. *Bruton et al.* [43] applied a wavelet transform to every training shape, obtaining a multi-scale decomposition of the surface, and computed localised multilinear models on the estimated wavelet coefficients. *Jin et al.* [169] applied non-negative matrix factorisation (NMF) since it decomposes a share into localised features. And finally, *Luthi et al.* [257] modelled shape variations using Gaussian processes, which provide a way of adding local models to global models, thus combining the information at multiple scales. The second drawback of 3DMMs was noted by [37, 168, 323], who argues that facial shape changes are not completely linear and hence cannot be adequately modelled using linear models. Using a mesh-to-mesh autoencoder, their method consists of discovering a latent space of face deformations. *Ranjan et al.* [323] and *Bouritsas* [37] modelled all the shape variations in a single latent space, as opposed to *Jiang et al.* [168],

who estimated two separated latent spaces, one corresponding to expression-related deformations. Whereas *Ranjan et al.* [323] and *Jiang et al.* [168] used spectral convolution operations, *Bouritsas* [37] proposed a spiral convolution that employs anisotropic filters, which enable a one-to-one mapping between the neighbours of a vertex and the parameters of the local filter.

To address the limitations mentioned above, we proposed a graph-based mesh reconstruction method that can directly generate the vertices locations from the 2D input images with the help of the proposed aggregated CNN and GCN.

Chapter 3

Researching Region and Boundary Correlations with Implicit Graph Representations

In this chapter, I introduce a graph based model, to address challenges of implicit structure modeling with geometry-aware graph representation.

Specifically, I built a novel graph neural network (GNN) based deep learning framework with multiple graph reasoning modules to explicitly exploit both region and boundary features in an end-to-end manner. The mechanism extracts discriminative region and boundary features, referred to as initialized region and boundary node embeddings, using a proposed Attention Enhancement Module (AEM). The weighted links between cross-domain nodes (region and boundary feature domains) in each graph are defined in a data-dependent way, which retains both global and local cross-node relationships. The iterative message aggregation and node update mechanism can enhance the interaction between each graph reasoning module's global semantic information and local spatial characteristics. Our

model, in particular, is capable of concurrently addressing region and boundary feature reasoning and aggregation at several different feature levels due to the proposed multi-level feature node embeddings in different parallel graph reasoning modules. Experiments on two challenging datasets demonstrate that our method outperforms state-of-the-art approaches for the segmentation of polyps in colonoscopy images and the optic disc and optic cup in colour fundus images. The trained models are made available at https://github.com/smallmax00/Graph_Region_Boudnary

3.1 Introduction

The precise evaluation of anatomical features in medical pictures is essential for the treatment of a broad range of medical illnesses and disorders. For instance, glaucoma is a chronic neurodegenerative condition, and a leading cause of irreversible but preventable blindness worldwide [384]. The relative size of the optic disc (OD) and optic cup (OC) in colour fundus images is often used to assess glaucomatous damage to the optic nerve head [134,213]. Similarly, colorectal polyps are positively related with colorectal cancer, the third most common cancer worldwide [364]. Segmenting polyps gives crucial diagnostic and surgical information on the location and shape of colorectal polyps. It is impracticable for doctors to manually annotate these structures since it is time-consuming, labor-intensive, and prone to human error. Automated and accurate biomedical image segmentation techniques are required to resolve this issue. To this purpose, I present a graph-based deep learning framework for segmentation problems, with the crucial innovation of aggregating information on an object's area and border. I demonstrate the framework's effectiveness for segmentation of polyps in colonoscopy images and OD & OC in colour fundus images.

Previous approaches of image segmentation based on deep learning focused on learning the intensity attributes of the input picture. Either they are region-based approaches that

do dense pixel classification or boundary-based methods that regress the position of the border. Neither approach, however, takes into account the fundamental region-boundary connection, which is essential for improving segmentation performance [72, 100]. Region characteristics, for instance, emphasise the global homogeneity of pixel-wise semantics and object-level contextual information. In contrast, boundary characteristics describe the local edge characteristics and spatial changes on both sides of the border contour. Intuitively, combining information about region and boundary features ought to improve segmentation. In addition, the subjective experience of doctors who annotate biological pictures often entails evaluating both the relevant area's specifics and the border that defines its edge. Clinicians often tour the cupped area to establish the OC border [307].

I demonstrates how to rationally combine region and boundary features using a single graph-structure model. This takes advantage of the proposed Graph Neural Network (GNN) model's long-range information propagation and cross-domain feature update capabilities. The summary pipeline of our work is depicted in Fig. 6.1, please refer to Fig. 4.2 for more details. The term 'cross-domain features' refers to the region features (containing semantic information) and boundary features (containing spatial information).

This study specifically includes information from the area and border domains of medical imaging objects of interest. Specifically, I design numerous graphs, each of which contributes to addressing the updating and reasoning of specific-level cross-domain features. There are region nodes with global semantic information and border nodes with local geographical properties in every network. Weighted linkages exchange and aggregate semantic and geographical information between nodes. Additionally, I introduce an attention enhancement module (AEM) in conjunction with two sequential attention mechanisms through the channel and the spatial inter-dependencies. The AEM is built between the multi-level backbone features and the corresponding constructed graph nodes to extract

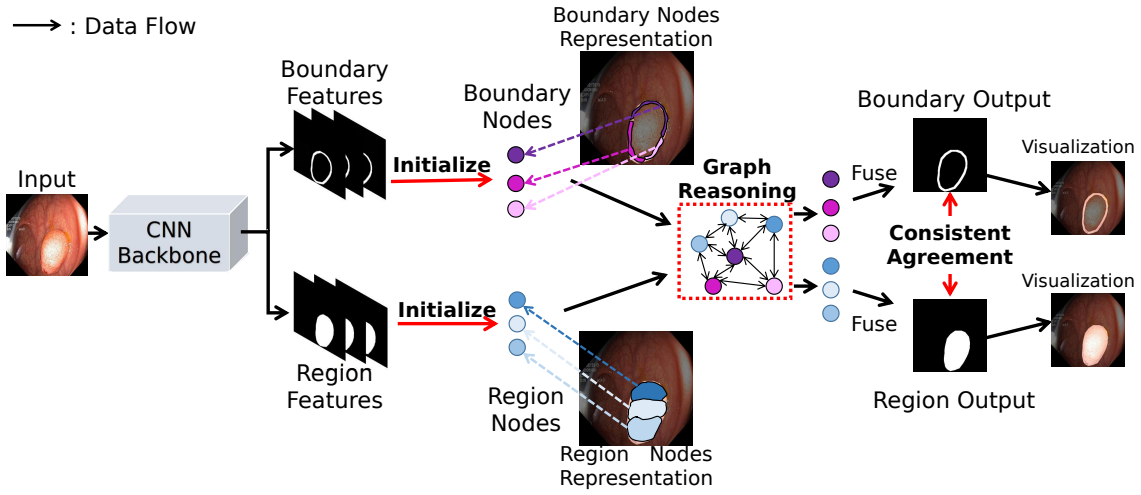


Figure 3.1: The pipeline of the proposed network, with the example of a colonoscopy polyp image as the input. The extracted region and boundary features from the *CNN* backbone are treated as the initialized graph nodes and then go through the graph-level feature aggregation and reasoning process. A requirement for consistency between the boundary and the region outputs forces the *GNN* to learn coherent features.

discriminative feature embeddings for the region and boundary nodes, respectively. To utilise the underlying coherence between the region and boundary segmentation predictions, I also generate a spatial gradient from the anticipated region mask as the resulting border probability map. To enforce boundary consistency in area mask prediction during model training, the differences between the resulting boundary probability map and the boundary ground-truth are defined as one of the loss terms, termed boundary agreement loss. Our experimental findings demonstrate that the new *GNN*-based framework significantly outperforms existing approaches.

In summary, these two works makes the following contributions:

- Despite the intuitive usage of both domains by human graders, segmentation techniques for biological pictures often ignore the underlying link between the area and border features. I present a unique trainable end-to-end segmentation model that incorporates area and boundary information as graph nodes and updates and prop-

agates cross-domain features.

- Cross-domain features are difficult to optimise simultaneously; in particular, the unavoidable prediction perturbation will hinder the simultaneous learning and updating of cross-domain features. Here, I provide a boundary agreement loss function, which ensures that the borders of the forecast area and boundary mask are consistent.
- Extensive experiments demonstrate that our proposed model outperforms the state-of-the-art approaches on two segmentation tasks. Instead of conducting experiments on a small number of datasets, I combine five different *OD & OC* segmentation datasets and five different colonoscopy polyp segmentation datasets, respectively. In terms of varying dataset sources, they may contain different annotation standards for ground truths by various clinicians. Nevertheless, our model achieves good segmentation performance, demonstrating its robustness and generalizability.

3.2 Related Works

3.2.1 Region-based Segmentation

Convolutional Neural Networks (*CNN*) have found widespread applications in medical image segmentation. Existing CNN-based methods [101, 109, 125, 164, 210, 295, 296, 337, 508] have considered segmentation as a dense pixel classification task. For example, the classic *U-net* [337] employs a skip-connection between the encoder and decoder to alleviate information loss; and it has served as a baseline model for segmentation tasks in recent years. Another classic region-based segmentation method, *U-Net++* [508], uses an aggregated mechanism to fuse multi-level features. However, it may result in excessive information flow because some low-level features are unnecessarily over-extracted while object boundaries are simultaneously under-sampled. Recently, *Gu et al.* proposed *CE-Net* [125] to

capture high-level information and preserve spatial information based on *U-Net* [337]. However, due to the limited receptive field of standard *CNN*, dense atrous convolutions were incorporated [59,470] to enlarge the receptive regions for long-range context reasoning. *M-Net* [109] represented the fundus image in polar coordinates, and achieved high accuracy in segmenting *OD & OC*. However, it needed additional processes, such as multi-scale input and side-output mechanisms with deep supervision, to achieve multi-level receptive field fusion for long-range relationship aggregation. Similarly, *Fan et al.* proposed a *Inf-Net* [101] to tackle *COVID-19* lung infection segmentation. A reverse attention module is included to work with deep supervision in terms of multiple side-outputs. The aforementioned methods have achieved promising results in segmentation tasks with the help of boosted long-range relationship reasoning abilities. However, they are not efficient since stacking local cues cannot always precisely handle long-range context relationships. Especially for pixel-level classification problems, such as segmentation, performing long-range interactions is important for reasoning in complex scenarios [59]. To address this challenge, recent self-attention [429] based methods [164,210] have demonstrated a superior ability to capture long-range relationships. For example, *Segtran* [210] proposed a squeezed attention block, which regularized the self-attention of *Transformers* [399], and an expansion attention block learned diversified representations. In this way, *Segtran* can calculate the pairwise interactions (self-attention) between all input units, combine their features and generate contextualized features. It has achieved promising results in the *OD & OC* and polyp segmentation tasks. On the other hand, in order to comprehend scenes or global contexts, these approaches must learn the object's position, boundary, and category from high-level semantic awareness and regional location information [247]. However, they tend to focus on learning image intensity features and suffer from a lack of regional position information at the pixel level [69]. This has resulted in inaccurate object boundary pre-

dictions.

3.2.2 Boundary-based Segmentation

Polygon-based boundary regression methods have drawn much recent attention. Polygon-based methods [73,278,345,424,449] regress the predefined vertex positions along the object boundaries and connect the predicted vertices to form a polygon, which is then converted into a mask. For example, *Cheng et al.* combined Active Contour Models (*ACMs*) [173] and *CNN*, to create a Deep Active Ray Network [73], which utilizes polar coordinates (*rays*) to represent active contours. Along the same lines, *Xie et al.* proposed *PolarMask* [449] to interpret the object boundary in a polar coordinate system and proposed a *CNN* to regress the length of *rays*, which implicitly estimates the object boundary. Similarly, *Meng et al.* proposed *CABNet* [278], which represents the object boundary as vertices, then explicitly estimates the vertex locations. It achieved promising results on *OD & OC* segmentation tasks. Other boundary-based methods [66, 68, 72, 175] integrate the boundary geometry constraint into the loss function or evaluation measurement. For example, *Kervadec et al.* proposed boundary loss [175] which takes the distance metric on contours' space to mitigate the difficulties of highly unbalanced foreground and background. *Cheng et al.* proposed a Boundary Intersection-over-Union (*BIoU*) [72] evaluation measurement, which quantifies boundary quality in region segmentation tasks.

These methods are applicable to segment the whole region of the objects by regressing the position of vertices along boundary contours. However, they overlook the intrinsic region-boundary relationship, which I suggest is crucial for enhancing segmentation performance.

3.2.3 Region and Boundary for Segmentation

Recent methods, such as [100, 101, 104, 299, 411, 425, 486, 489], explicitly or implicitly considered the dependency between the regions and boundaries of an object of interest in *OD & OC* or polyp segmentation. Specifically, *Zhang et al.* proposed *ET-Net* [489] for *OD & OC* segmentation, where an edge attention mechanism is proposed to explicitly emphasise the object boundary. On the other hand, *Fan et al.* [100, 101] and *Zhang et al.* [486] shared a similar boundary attention idea, where the object boundary is implicitly extracted from region predictions with a foreground erasing mechanism. In general these approaches treat segmentation as a multi-task learning problem, by using a shared backbone and two independent sub-networks to extract features of the regions and the boundaries, respectively. Then, the extracted features of regions and boundaries are directly fused with basic fusion operations such as element-wise addition or multiplication [100, 425, 486], or channel-wise concatenation [411, 489] with or without a fusion operation [104, 299].

I suggest that the correlations between region and boundary features cannot be adequately captured and exploited by two *independent* sub-networks that rely on these types of primary fusion operations. An intuitive solution would be to aggregate region and boundary features during the *whole learning process*. Unfortunately, the extracted region and boundary features are necessarily from two different domains and so contain varying semantic and spatial details. For example, region features focus on global homogeneity in pixel-wise semantics and object-level contextual information; while boundary features describe local edge characteristics and spatial variations on both sides of the boundary contours. It is well known that concurrently optimizing cross-domain features are difficult. Our experimental results also support this, and readers are directed to *Ablation Study* (Section 6.5.4) for detailed information. In contrast, our method studies the cross-domain relationship of the region and boundary features throughout the whole training process

with the help of the proposed *GNN* module. In other words, our model benefits from complementary cross-domain feature exchange and self-domain information propagation of region and boundary features along the entire training pipeline through the proposed graph structure model. Our experimental results prove that the proposed GNN reasoning module can tackle cross-domain feature optimization and achieved promising results on two segmentation tasks.

3.2.4 *GNN* in Segmentation

Graph-structure models have recently been adopted for segmentation tasks because of their natural aptitude for long-range information propagation and feature updates. *Dong et al.* [94] and *Shen et al.* [356] exploited the traditional random walk algorithm on a graph to tackle image segmentation tasks. However, the energy formulations for describing the images are complicated and higher-order energy function based methods [355, 357] may be needed to solve the problem. Recently, *Yao et al.* proposed a *GNN* network [466] to study the 3D geometrical relationship between vertices through mesh representation in an organ segmentation task. With the nature of *GNN*, long-range shape information can be updated and passed among vertices to maintain a consistency constraint. Along the same lines, *Voxel2mesh* [439] learned a deformable mesh representation through *GNN* to propagate the voxel features along the edges of the built graph model. Another paper [360] by *Shin et al.* used *GNN* to learn the global structure of the vessel's shape, which mirrored the connectivity of neighbouring vertices. Similarly, *Meng et al.* proposed *RBA-Net* [276] to regress the *OD & OC* boundaries by aggregated *CNN* and *GCN*, which learns the long-range features and directly regresses vertex coordinates in a Cartesian system.

The methods mentioned above used *GNN* to address the problem of intra-domain long-range feature propagation, as messages passing between graph nodes share similar

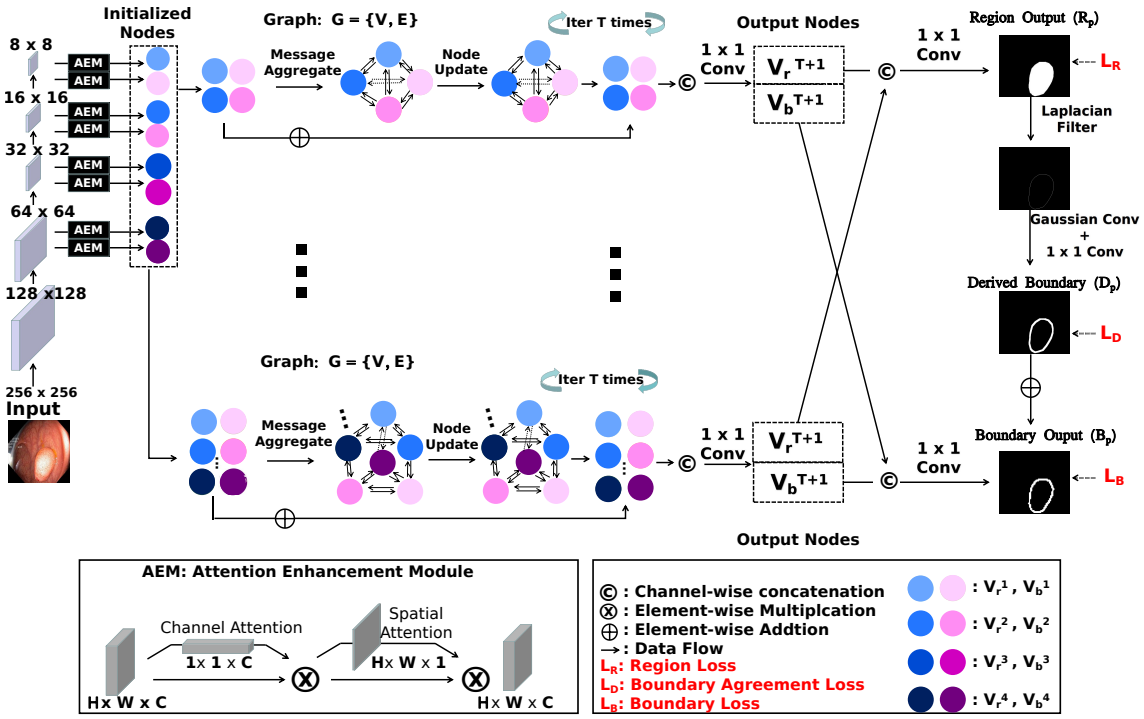


Figure 3.2: Overview of the proposed *GNN* model (best viewed in color). The initialized nodes from the *AEM* output are interpolated into the same scale (32×32) through the bi-linear interpolation layer. For simplicity, I present only two graph reasoning modules in the middle, with the top one containing two region nodes and two boundary nodes from relatively deep feature level and the bottom one containing four region nodes and four boundary nodes from both shallow and deep feature levels. In this figure, I demonstrate how to segment polyps. As for *OD & OC* segmentation, the only difference is that the output probability map has a channel size of 2.

semantic and spatial characteristics. In contrast, our method considers extracted region and boundary features as distinct graph nodes and employs *GNN* to learn their inter-domain relationship. Additionally, methods such as [276, 439, 466] represented each graph node with a predefined vertex and the corresponding coordinate under the form of mesh [439, 466] or triangle [276]. In that kind of framework, each graph node can only represent a single location. In contrast, our method represents each graph node with a set of pixels (locations) in the region area or boundary area (shown in Fig. 6.1).

3.3 Methods

Fig. 4.2 shows the model architecture of the proposed method. Given an input image, I extract the multi-level features through a backbone network. Following *PraNet* [100], I adopt the truncated *Res2Net* [114] as the backbone due to its superior ability to extract features in the segmentation task. I propose to use several *GNN* modules to reason and aggregate the extracted multi-level region and boundary features, which are elaborated as follows.

3.3.1 Attention Enhancement Module

Inspired by [440], I applied an attention enhancement module (AEM) upon each of the extracted multi-level backbone features. Specifically, the AEM is designed as a sequential operation consisting of channel attention $\mathbf{C}_{att}(\cdot)$ and spatial attention $\mathbf{S}_{att}(\cdot)$. The AEM is defined as: $F_{AEM}(f) = \mathbf{S}_{att}\left(\mathbf{C}_{att}(f)\right)$, where $\mathbf{C}_{att}(f) = f \otimes MLP(\mathbf{Pool}_c(f))$, $MLP(\cdot)$ is a multi-layer perceptron with two layers and sigmoid as the activation function; f is the input feature; $\mathbf{Pool}_c(\cdot)$ denotes the global max pooling for each feature map; $\otimes$ represents the multiplication by the dimension broadcast. In addition, $\mathbf{S}_{att}(f) = f \otimes Conv(\mathbf{Pool}_s(f))$, where $Conv(\cdot)$ is a 3×3 convolution layer with padding=1, followed by a sigmoid activation function; $\mathbf{Pool}_s(\cdot)$ denotes the global max pooling operation for each position in the feature map along the channel axis. In contrast to [440], I omitted the additional feature merging operations, such as the average pooling layer, in order to retain the most critical extracted characteristics.

As shown in Fig. 4.2, for each resolution's backbone feature map, I applied two *AEMs*, resulting in attention-enhanced region and boundary feature maps, respectively, which is referred to as the initialised nodes (region nodes $\mathbf{V}_r$ and boundary nodes $\mathbf{V}_b$). Fig. 6.1 demonstrates the boundary node and region node representations. Each node represents

a set of relative features (pixels), such as region pixels and boundary pixels. The subsequent graph reasoning module treats each region and boundary nodes independently; afterwards, the output nodes of region and boundary are fused separately, resulting in region output $\mathbf{R}_p$ and boundary output $\mathbf{B}_p$. The whole network is end-to-end trainable; the supervision gradients of the region and boundary ground truth will back-propagate to the corresponding *AEM*, respectively. Thus, the two *AEM* will excavate the discriminative feature embeddings for the region and boundary features from each resolution's backbone feature.

3.3.2 Graph Based Reasoning

Fig. 4.2 illustrates several graphs in parallel that address the cross-domain, cross-level reasoning with varying numbers of region nodes $\mathbf{V}_r$ and boundary nodes $\mathbf{V}_b$. In this manner, the deep-level semantics of a region of interest, and the shallow-level spatial characteristics of the associated boundary can be interpreted as a whole. In the *Ablation Study* section, I perform detailed studies to evaluate the effectiveness of the number of graphs and the number of node updating times in each graph.

Graph Node Initialization

In our graph-based reasoning module, I construct multiple graphs in parallel, in which various levels of the attention-enhanced features are referred to as the initialized region node embeddings $\mathbf{V}_r = \{v_{r^1}, ..., v_{r^n}\}$ and boundary node embeddings $\mathbf{V}_b = \{v_{b^1}, ..., v_{b^n}\}$. In other words, I treat the extracted region and boundary output features of the *AEM* module as the corresponding region and boundary nodes in the proposed graph. The underlying motivations are twofold: (1) As mentioned before, the region and boundary output features from *AEM* contain different levels (shallow and deep) and domains (region

and boundary) of information. In order to obtain complementary information from those features I treated them as graph nodes and used the message passing and information exchange mechanism of *GNN*. (2) In general, a GNN model propagates messages through a graph, with each node's representation conditioned on its relationships with surrounding nodes as well as its own information. Thus, through passing messages among different nodes, relevant information and relations may be gradually distilled for learning feature embeddings, where the region and boundary segmentation can be derived.

Single Graph Reasoning Module

In this section, I demonstrate the structure and components of a single graph, such as the one on the top middle in Fig. 4.2, in which there are four nodes with low-resolution (8×8 and 16×16); the one on the bottom middle has eight nodes of both low- and high-resolutions (from 8×8 to 64×64). Please note that, rather than being chosen at random, the nodes in each graph are fixed during training. Thus, each graph will address specific levels of the region and boundary feature aggregation process.

Node Embeddings. Given the initialized region nodes $\mathbf{V}_r = \{v_{r^1}, ..., v_{r^n}\}$ and boundary nodes $\mathbf{V}_b = \{v_{b^1}, ..., v_{b^n}\}$, I interpolate them to have the same size through the bi-linear interpolation layer. Then, I construct the graph $G = \{\mathbf{V}, \mathbf{E}\}$, where $\mathbf{V} = \mathbf{V}_r \cup \mathbf{V}_b$, are the combination of region and boundary nodes.

Edge Embeddings. For information propagation, nodes are linked with each other by weighted edges $\mathbf{E} = \{e_1, ..., e_{n^2-n}\}$, where the weighted edges can reflect the different correlations among various nodes. Rather than randomly initialising the edges, I define the edges in a data-dependent way. Inspired by [251, 432], for two linked nodes v_i, v_j from $\mathbf{V}$, the edge $\mathbf{e}_{i,j}$ from v_i to v_j is defined as:

$$\mathbf{e}_{i,j} = Conv(Cat(v_i - v_j, v_j)), \tag{3.1}$$

where $Cat(\cdot)$ is channel-wise concatenation, $Conv(\cdot)$ represents a 1×1 convolution layer to learn the relationships and minimise the channel size into 1. Thus, data-dependent local information $v_i - v_j$ and global information v_j are both considered in the edge $\mathbf{e}_{i,j}$. Note that, $e_{i,j}$ has the same size as v_i and v_j. In contrast, the edge $\mathbf{e}_{j,i}$ from v_j to v_i is defined as:

$$\mathbf{e}_{j,i} = Conv(Cat(v_j - v_i, v_i)). \tag{3.2}$$

In this way, the weighted edge embeddings contain the self-information of the starting node and the cross-information (cross domains or cross levels) of the connected node. Thus, both types of information can be aggregated to other connected nodes during the messaging passing process. The edge is defined as directional so as to distinguish the directional information passing and message aggregation among different nodes.

Message Aggregation & Nodes Update. In our *GNN* model, nodes connect with each other; as a result, each node aggregates the cross-level (deep and shallow) and cross-domain (region and boundary) messages from all its neighbouring nodes, then the node embeddings will be updated. At *T-th* update step, for the node v_i^{T-1} and all its neighbour nodes v_j^{T-1}, the message aggregation function $m_{j,i}^T$ from v_j^{T-1} to v_i^{T-1} is defined as:

$$m_{j,i}^T = \sum_{j}^{n-1} ReLU(e_{j,i}^{T-1}) \odot v_j^{T-1}, \tag{3.3}$$

where $\odot$ is element-wise multiplication; $ReLU(\cdot)$ as the non-linear function to convert the edge embeddings to link weight. Then I update the node embeddings with a residual connection:

$$v_i^T = (\sum_{j}^{n-1} m_{j,i}^T) + v_i^{T-1}, \tag{3.4}$$

where the last step node embeddings v_i^{T-1} is maintained for the subsequent graph reasoning

process.

After T times message aggregations and node updates, I fuse the region nodes $\mathbf{V}_r^{T+1} = \{v_{r^1}^{T+1}, ..., v_{r^n}^{T+1}\}$ and boundary nodes $\mathbf{V}_b^{T+1} = \{v_{b^1}^{T+1}, ..., v_{b^n}^{T+1}\}$ respectively through channel-wise concatenation, following by 1×1 convolution to generate the output region nodes and boundary nodes. T is 3 in our work.

Multi-level Graph Reasoning Modules

As observed by others [100, 508], the deep- and shallow- layer features from different levels complement one another, with the deep-layer features containing extensive semantic region information and the shallow-layer features retaining adequate spatial boundary information. To this end, I expand the proposed *GNN* by running several graph reasoning modules concurrently (2 in our work). Each graph includes region and boundary nodes from different shallow and deep feature levels of the backbone network. Thus, each graph reasoning module will address specific levels of aggregation and reasoning about region and boundary features. For example in Fig. 4.2, the top reasoning graph tackles the deep-level feature aggregation (8×8, 16×16), and the bottom reasoning graph tackles the shallow- and deep-level feature aggregation (8×8, 16×16, 32×32, 64×64). Finally, I fuse the output region ($\mathbf{V}_r^{T+1}$) and boundary nodes ($\mathbf{V}_b^{T+1}$) of each parallel graph respectively by channel-wise concatenation, followed by a 1×1 convolution with sigmoid activation function, then up-sample to obtain the region and boundary segmentation predictions ($\mathbf{R}_p$ and $\mathbf{B}_p$, with the same size of 256×256 as the input images). Please note, the parallel graphs are not connected during the reasoning process but have connections (fusion) on the output nodes. This is because each graph is designed to concentrate exclusively on a particular set of levels (resolutions) of nodal reasoning. I found that adding connections between graphs did not improve segmentation, but did increase training time.

3.3.3 Loss Function

The total loss function is defined as:

$$L_{total} = \mathbf{L}_R + \beta \cdot (\mathbf{L}_B + \mathbf{L}_D), \tag{3.5}$$

where **Dice Loss** [289] ($\mathbf{L}_R$) is used for the region segmentation predictions to penalize the mismatch regions against the corresponding ground truth. I defined L_R as:

$$L_R(R_p, Y_R) = 1 - \frac{2R_p GT_R + 1}{R_p + GT_R + 1}, \tag{3.6}$$

where R_p and GT_R denote the region segmentation predictions and the ground truth. Here, 1 is added to avoid divide by zero errors, such as when $R_p = GT_R = 0$. I also adopt the signed distance map loss (L_{sdm}) [175] as the boundary loss ($\mathbf{L}_B$) on boundary segmentation predictions due to the challenge of highly imbalanced foreground and background [258]. In detail, [175] used an integral approach for computing boundary variations with a signed distance transformation map, which can avoid complex local differential computations. Formally, the signed distance function (SDF) of segmentation ground truth (GT) can be defined as:

$$GT_{SDF} = \begin{cases} \underset{y \in \Delta G}{-inf} ||x - y||_2, & x \in GT_{in} \\ 0, & x \in \Delta G \\ \underset{y \in \Delta G}{inf} ||x - y||_2, & x \in GT_{out} \end{cases}$$

where $||x - y||_2$ represent the Euclidean distance between pixel x and y. Besides, GT_{out}, GT_{in} and ΔG, denote the outside, inside and boundary of the object, respectively. Given the signed distance maps of ground truth (GT_{SDF}) and the sigmoid outputs of the model

$Pred_\theta$ (θ is the parameters), the signed distance map loss (L_{sdm}) is represented as:

$$L_{sdm}(Pred_\theta, GT_{SDF}) = Pred_\theta \odot GT_{SDF}, \tag{3.7}$$

where $\odot$ denotes Hadamard product. In this way, I can represent the boundary loss L_B in this work as:

$$L_B = L_{sdm}(B_p, GT_B), \tag{3.8}$$

where GT_B represents the signed distance map of the boundary segmentation ground truth. β is empirically set as 0.5 to balance the losses between Dice loss, region and boundary predictions.

Boundary Agreement Loss ($\mathbf{L}_D$). Firstly, I derive the spatial gradient from the predicted region mask (R_p), as the derived boundary probability map (D_p). In detail, I empirically adopt the *Laplacian* filter as a 3×3 kernel $[[1,1,1],[1,-8,1],[1,1,1]]$ convolution layer to compute the spatial gradient in an end-to-end manner. The *Laplacian* filter is the direct result of a finite-difference approximation of the spatial derivative [108], highlighting the rapid intensity change regions. However, this will lead to thin and coarse derived boundaries, which results in extremely unlabeled classes (Shown in Fig. 4.2). To address this issue, I then empirically applied an approximated 3×3 *Gaussian* kernel convolution layer (*sigma* equals to 3 for two directions), followed by a 1×1 convolution layer to increase the boundary width and address the unbalanced issues [72]. The derived boundary probability map (D_p) is defined as:

$$D_p = Conv_{1\times1}\Big(Gaussian_{3\times3}\big(Laplacian_{3\times3}(R_p)\big)\Big). \tag{3.9}$$

Furthermore, the signed distance map loss [175] is applied to it against the boundary ground truth due to address the challenge of unbalanced classes.

The boundary agreement loss (L_D) is defined as:

$$L_D = L_{sdm}(D_p, GT_B). \tag{3.10}$$

With boundary agreement loss, region segmentation can benefit from additional boundary constraints, resulting in more reliable region segmentation predictions with more accurate boundary details. The boundary ground truth was generated by applying the same *Laplacian* filter and *Gaussian* kernel convolution to the corresponding segmentation ground truth mask. I then converted it into a binary map with threshold 0 as the final ground truth.

Furthermore, I empirically found that incorporating the derived boundary (D_p) into the boundary output (B_p) can enhance both the region and boundary segmentation performance. Thus, to augment the segmentation accuracy, I fuse the derived boundary probability map $\boldsymbol{D}_p$ with the boundary segmentation map $\boldsymbol{B}_p$ in terms of element-wise addition. The resulting concatenated feature map is then fed into a 1×1 convolution layer with a sigmoid activation function to produce the final boundary segmentation prediction. In this way, the boundary segmentation prediction $\boldsymbol{B}_p$ can benefit from the feature supplement provided by the derived boundary maps $\boldsymbol{D}_p$.

3.4 Experiments

3.4.1 Datasets

I evaluate our approach with two distinct yet challenging medical image segmentation tasks: segmentation of *OD & OC* from retinal images, segmentation of polyps from colonoscopy images. Accurate segmentation of the *OC* in colour fundus images is often difficult because of poor contrast between the cup and the surrounding rim [307]. The boundary between a

Table 3.1: Quantitative segmentation results of *OD & OC* and polyps on respective testing datasets. The performance is reported as *Dice* (%) and *B-Acc* (%) and *BIoU* (%). 95% confidence intervals are presented in the brackets, respectively. I compare our model with previous state-of-the-art methods by running their open-source code. Notably, I sampled 120 vertices for *PolarMask* [449], *CABNet* [278] and *RBA-Net* [276] to construct a smooth boundary.

Tasks / Methods	OC			OD			Polyps		
	Dice (%)↑	*B-Acc (%)*↑	*BIoU(%)*↑	*Dice (%)*↑	*B-Acc (%)*↑	*BIoU(%)*↑	*Dice (%)*↑	*B-Acc (%)*↑	*BIoU(%)*↑
U-Net [337]	85.3 (82.1, 86.8)	87.1 (85.9, 88.8)	80.1 (77.6, 82.4)	95.0 (93.1, 97.1)	97.0 (95.3, 98.6)	86.2 (84.1, 88.3)	66.7 (63.6, 68.1)	73.7 (72.1, 75.1)	60.0 (57.6, 62.2)
U-Net++ [508]	86.0 (83.8, 88.5)	87.6 (85.3, 89.1)	81.4 (79.5, 83.8)	95.0 (93.9, 96.1)	97.9 (97.0, 98.5)	88.0 (86.4, 89.8)	65.6 (63.1, 67.7)	72.6 (70.1, 74.4)	58.8 (55.6, 61.3)
M-Net [109]	86.9 (85.0, 88.0)	89.7 (88.3, 90.9)	82.9 (79.5, 84.7)	96.8 (95.5, 97.6)	96.7 (95.9, 97.9)	88.1 (87.0, 89.3)	-	-	-
PolarMask [449]	87.2 (85.3, 89.1)	90.9 (88.7, 91.6)	83.2 (81.0, 85.1)	96.5 (95.8, 97.2)	97.8 (96.9, 98.5)	87.0 (86.0, 88.3)	69.3 (67.2, 71.4)	83.6 (81.2, 85.7)	60.3 (58.4, 61.9)
PraNet [100]	-	-	-	-	-	-	74.0 (72.6, 75.7)	85.6 (84.1, 86.9)	66.0 (63.3, 68.9)
Psi-Net [299]	85.7 (83.0, 88.2)	87.1 (85.5, 89.0)	82.1 (80.3, 84.0)	95.8 (94.5, 97.1)	97.7 (96.5, 98.4)	87.9 (85.4, 89.2)	63.8 (59.7, 65.9)	75.5 (73.1, 77.2)	57.1 (55.7, 58.6)
RBA-Net [276]	87.8 (85.2, 89.7)	89.5 (87.1, 91.6)	83.8 (81.6, 85.9)	96.1 (95.5, 96.7)	97.5 (96.4, 98.1)	88.9 (88.0, 89.2)	73.5 (71.2, 75.6)	85.1 (83.0, 87.3)	66.2 (64.8, 67.9)
ACSNet [486]	-	-	-	-	-	-	70.1 (67.8, 72.3)	82.6 (80.8, 84.4)	63.3 (60.1, 65.7)
CABNet [278]	87.1 (84.9, 88.8)	88.8 (87.1, 90.2)	83.0 (81.1, 85.4)	95.5 (94.6, 96.7)	96.4 (95.5, 97.2)	88.2 (87.1, 89.6)	73.0 (70.7, 75.4)	84.2 (82.0, 86.3)	65.5 (63.2, 67.7)
Segtran [210]	88.8 (86.5, 90.3)	91.0 (88.6, 93.2)	83.9 (81.3, 85.8)	97.3 (96.1, 98.2)	97.5 (96.6, 98.8)	90.0 (89.1, 91.2)	75.3 (73.5, 77.1)	86.5 (84.4, 88.3)	67.9 (65.5, 69.2)
Ours	**89.4** (87.6, 90.8)	**91.7** (91.1, 92.5)	**85.1** (83.3, 86.8)	**97.7** (97.0, 98.7)	**98.1** (97.8, 98.5)	**91.1** (90.2, 92.0)	**75.7** (73.1, 77.6)	**87.0** (86.1, 88.3)	**69.3** (67.9, 70.5)

polyp and its surrounding mucosa is typically blurred in colonoscopy images and lacks the intense contrast required for segmentation approaches [163].

Fundus images of OD and OC: I pooled 2068 images from five datasets (Refuge [307], Drishti-GS [370], ORIGA [490], RIGA [6], RIM-ONE [111]). 613 fundus images were randomly selected as the test dataset, leaving the other 1455 images for training and validation. Following [276], I located the disc center from each image and then cropped a subimage of 256 × 256 pixels centered on the disc for the subsequent analysis.

Colonoscopy polyp images: I retrieved 2085 colonoscopy images from five datasets (ETIS [364], CVC-ClinicDB [28], CVC-ColonDB [379], EndoScene-CVC300 [400], and Kvasir [163]). I used the same data split settings as [100], namely 1450 colonoscopy images

from Kvasir [163] and CVC-ClinicDB [28] comprised the training and validation datasets. The remaining 635 colonoscopy images from [364, 379, 400] were used for testing. All of the images are uniformly resized to 256×256.

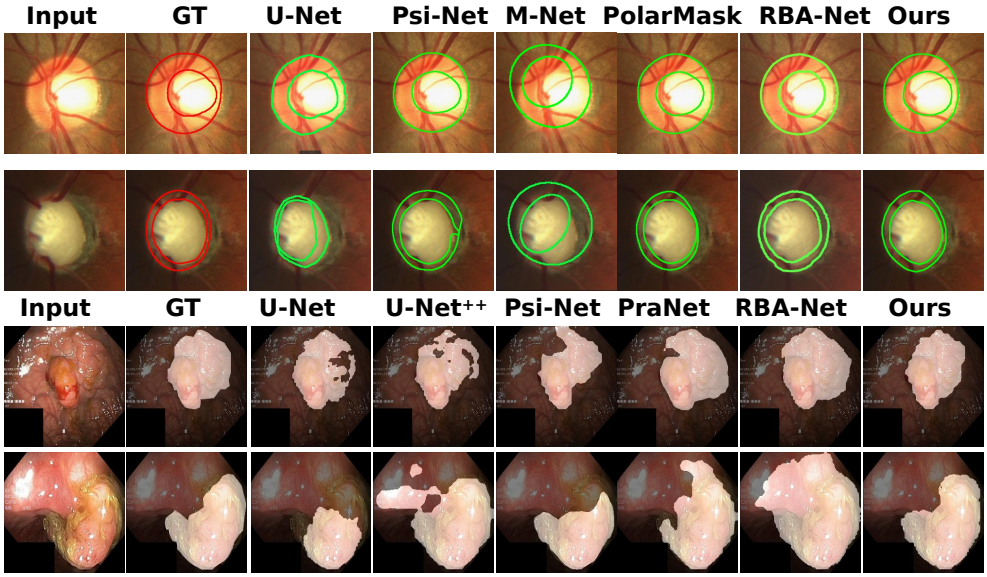


Figure 3.3: Qualitative results of *OD & OC* segmentation and colonoscopy polyp segmentation. I compare our model with *U-Net* [337], *U-Net++* [508], *M-Net* [109], *PolarMask* [449], *PraNet* [100], *Psi-Net* [299], *RBA-Net* [276]. Our method can produce more accurate segmentation results when compared with ground truth (*GT*). Note that I plot the boundary (spatial gradient through *Laplacian* filter) of the region mask on the input image to better visualise the *OD & OC* segmentation comparison. Along the same lines, I highlight the region in the input image for colonoscopy polyp segmentation comparison.

3.4.2 Experimental Setting and Evaluation Metrics

To augment the dataset, I randomly rotated and horizontally flipped the training dataset with a probability of 0.3. The rotation ranges from -30 to 30 degree. I use stochastic gradient descent with a momentum of 0.9 to optimize the overall parameters. I trained the model around 300 epochs for all the experiments, with a learning rate of 1e-2 and a decay rate of 0.5 every 100 epochs. The batch size was set as 48. The network was

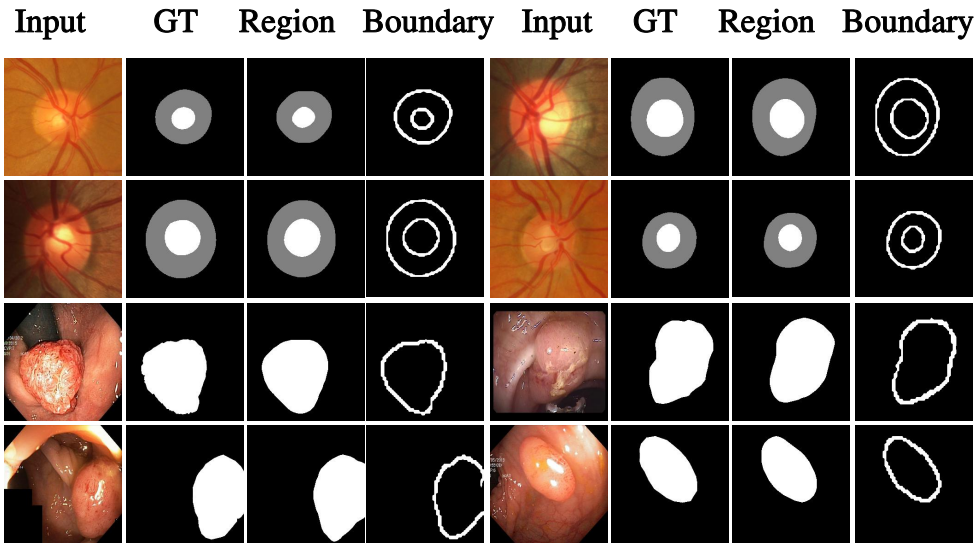


Figure 3.4: Figure shows the binary mask comparison between our model's prediction and the ground truth. Our model produces consistent region **(Region)** and boundary **(Boundary)** predictions compared with the ground truth **(*GT*)**.

trained end-to-end; all the training processes were performed on a server with 4 TESLA V100, and all the test experiments were conducted on a local workstation with *Intel(R) Xeon(R) W-2104 CPU* and *Geforce RTX 2080Ti GPU* with 11GB memory. Five-fold cross-validation was used for fair comparison and hyper-parameters tuning in all settings. I randomly selected 10% of the training dataset for internal validation.

I report Dice similarity score (*Dice*) and balanced accuracy (*B-Acc*) as the region segmentation accuracy metrics; and Boundary Intersection-over-Union (*BIoU*) [72] as the boundary segmentation metric. 95% confidence intervals were generated by using 2000 sample bootstrapping. As for *BIoU* [72], compared with other boundary-based evaluation metrics such as *Trimap IoU* [59, 185] or *Boundary F1-measure* [81, 311], *BIoU* is more sensitive to show boundary errors on small objects (*e.g.* polyps) [72]. *BIoU* is defined as:

$$BIoU = \frac{|(B_p \cap Y_B) \cap (R_p \cap Y_R)|}{|(B_p \cap Y_B) \cup (R_p \cap Y_R)|}, \tag{3.11}$$

Table 3.2: Ablation study on different feature fusion methods. The performance is reported as *Dice* (%), *BIoU* (%), on the two segmentation test datasets.

Methods \ Tasks	OC		OD		Polyps	
	Dice (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑
w/o Fusion	86.6	79.1	94.7	86.7	71.2	64.6
w/ Addition	87.0	81.7	96.0	86.6	70.9	63.0
w/ Concatenation	85.7	80.1	94.8	87.5	71.1	65.3
w/ Non-local [429]	87.2	83.4	95.2	89.6	74.9	69.1
w/ *GloRe* [70]	88.1	84.3	96.1	89.9	73.7	67.5
Ours	**89.4**	**85.1**	**97.7**	**91.1**	**75.7**	**69.3**

where Y_B and Y_R are the boundary segmentation ground truth and the region segmentation ground truth, respectively; R_p and B_p are the region and boundary predictions.

3.4.3 Performance Comparison and Analysis

In this section, I show qualitative (Fig. 3.3, Fig. 3.4) and quantitative (TABLE 5.3) results of the *OD & OC* and polyp segmentation tasks. The best result in each category is highlighted in bold.

***OD & OC* Segmentation** Fig. 3.3 and 3.4 show qualitative results. TABLE 5.3 provides the quantitative results of *Ours* and other methods. I obtain an average 89.4% and 97.7% *Dice* on *OC* and *OD* segmentation, respectively, outperforming approaches based on region segmentation such as *U-Net++* [508] and *M-Net* [109] by an average of 3.4% and 1.9% respectively; outperforming polygon-based boundary regression approaches such as *Polar-Mask* [449] by 1.9%; outperforming boundary-region based methods such as *Psi-Net* [299] by 3.2%; and outperforming *GNN* based segmentation methods such as *RBA-Net* [276], *CABNet* [278] by 1.8% and 2.5%. Note that *PraNet* [100] and *ACSNet* [486] are specially designed for binary segmentation of colorectal polyps with respect to the implicit region-

boundary reverse attention module. I cannot extend it to *OD & OC* segmentation directly since this is a multi-segmentation task. On the other hand, training two models, one for *OD* segmentation and another for *OC* segmentation, would be unfair to the other models under comparison. As a result, this model was not tested on the *OD & OC* segmentation tasks.

Polyp Segmentation TABLE. 5.3 and Fig. 3.3, Fig. 3.4 show the quantitative and qualitative results. Our model achieves 75.7% *Dice*, which outperforms the cutting-edge *ACSNet* [486] and *PraNet* [100] by 8.0% and 2.2% respectively. As for boundary segmentation accuracy, our model achieves 69.3% *BIoU*, which is 5.0% better than *PraNet* [100] and 8.0% better than *ACSNet* [486]. Our model size ($\sim$ 38.69 million parameters) is larger than *PraNet* [100] ($\sim$ 30.49 million parameters) when our framework has 2 graph reasoning modules (shown in TABLE 3.4). However, our model can gain more accurate segmentation performance (74.3% *Dice*; 68.1% *BIoU*) with a comparable model size ($\sim$ 30.57 million parameters) with *PraNet* [100] when the number of graph reasoning modules is 1 (N = 1 in TABLE 3.4). *Segtran* [210] is a very recent region-based approach for polyp segmentation. It benefits from the long-range feature reasoning ability of *Transformer* [399], and achieves comparable performance with *ours*. However, it has a larger model size (93.0 million parameters) than *ours* (38.69 million parameters), and due to the complexity of the model structure it has a relatively lower inference speed (8.7 *fps*) compared with *ours* (21.6 *fps*) on our local machine).

3.5 Discussion and Conclusion

3.5.1 Ablation Study

I conducted detailed ablation studies, and all the results demonstrate our model's effectiveness. As an illustration, the ablation results for different feature fusion methods, network components, attributes of the graph reason modules, and loss functions are shown in TABLE 3.2, TABLE 3.3, TABLE 3.4, and TABLE 6.10.

Feature Fusion. In this section, I evaluated the effectiveness of the proposed *GNN* reasoning module. Firstly, I replaced the *GNN* module with two feed-forward *CNN* blocks for the region and boundary features, respectively, to minimise the model size gap and retain a comparable number of parameters (e.g., $\sim$ 38.69 million for our model). In each *CNN* block, I built several standard convolution layers with kernel size 3×3, padding 1, followed by a Batch Normalization layer. Then, the boundary and region features are fused in three ways (similar to previous methods [100, 104, 299, 411, 425, 489]), including element-wise addition [100, 425], channel-wise concatenation [411, 489] or without fusion operation [104,299]. Finally, two 1×1 convolution layers were added to generate the region and boundary predictions. Additionally, I adopted two more potent fusion mechanisms to show our proposed *GNN* reasoning module's superiority. In detail, I applied the Non-local module [429], and *GloRe* module [70] respectively, where the Non-local module exploits a self-attention mechanism [399] and *GloRe* utilizes graph convolution [182] to tackle the long-range relations among features. TABLE 3.2 shows that our model with the *GNN* reasoning module achieves much more accurate and reliable results than simple fusion operations and outperforms the Non-local and *GloRe* modules by 2.2% and 2.0% in terms of *Dice* (%); and 1.4% and 1.7% in terms of *BIoU* (%) on two segmentation tasks respectively.

Network Components. This section presents the results of our ablation study on net-

Table 3.3: Ablation study on different model structure components. The performance is reported as *Dice* (%), *BIoU* (%), on the two segmentation test datasets. The best results are highlighted in bold.

Methods \ Tasks	OC		OD		Polyps	
	Dice (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑
w/o *AEM*	87.1	83.6	96.7	89.2	74.0	68.0
w/o *Gaussian*	88.3	84.7	97.1	90.2	74.6	68.1
w/o Boundary nodes	86.3	82.0	94.8	88.7	72.9	66.0
w/o Region nodes	83.6	80.2	91.2	87.5	64.1	57.9
Ours	**89.4**	**85.1**	**97.7**	**91.1**	**75.7**	**69.3**

Table 3.4: Ablation study on the attributes of the graph reason modules. The segmentation performance is reported as *Dice* (%), *BIoU* (%); the inference speed is reported as frame per second (*fps*) on the two testing datasets. Additionally, I present the model size in millions of parameters. The best result in each category is highlighted in bold.

Methods \ Tasks	*OD & OC*			Polyps			*Model Size*
	Inference (fps) ↑	*Dice* (%)↑	*BIoU* (%)↑	*Inference (fps)*↑	*Dice* (%)↑	*BIoU* (%)↑	*(# of parameters in millions)*↓
$N = 1,\ T = 3$	∼21.6	92.1	86.6	∼29.3	74.3	68.1	∼30.57
$\boldsymbol{N = 2,\ T = 3}$	∼21.6	93.6	88.1	∼29.3	75.7	69.3	∼38.69
$N = 3,\ T = 3$	∼21.6	91.8	86.1	∼29.3	72.1	66.0	∼46.56
$N = 2,\ T = 1$	∼38.1	92.0	87.4	∼44.0	74.8	68.3	∼38.69
$\boldsymbol{N = 2,\ T = 3}$	∼21.6	93.6	88.1	∼29.3	75.7	69.3	∼38.69
$N = 2,\ T = 5$	∼3.7	91.9	87.3	∼13.8	73.4	68.1	∼38.69

work structure components. I evaluated the effectiveness of the attention enhancement module (*AEM*), *Gaussian* kernel convolution layer, boundary nodes, and region nodes, respectively. I did this by removing each of those components in turn while retaining the rest of the structure. Notably, I overlooked the model size difference for the ablation study of the *AEM* and the *Gaussian* kernel convolution layer because there is no significant difference in the number of model parameters. To retain a comparable model size for the boundary nodes and region nodes ablation studies, I added feed-forward *CNN* blocks (same

Table 3.5: Ablation study on the loss function. The performance is reported as *Dice* (%), *BIoU* (%) on two segmentation test datasets. The best result in each category is highlighted in bold.

Methods \ Tasks	OC		OD		Polyps	
	Dice (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑
w/ Dice Loss	87.0	83.2	95.2	89.0	73.3	67.0
w/o Agreement Loss	88.1	84.1	96.2	90.0	74.2	67.9
Ours	**89.4**	**85.1**	**97.7**	**91.1**	**75.7**	**69.3**

as the one in the Feature Fusion ablation study) after the *GNN* reasoning module. (1). The *AEM* is designed to extract the discriminating features for boundary and region nodes through the back-propagation mechanism in the proposed end-to-end trainable network. TABLE 3.3 demonstrates that our model (*Ours*) improves average 2.1% *Dice* and 2.0% *BIoU*, respectively, using *AEM* upon two segmentation test datasets.

(2). The *Gaussian* kernel convolution layer (*Gaussian*) is critical to increasing the boundary width in the generation of boundary ground truth and the derived boundary prediction (D_p). As discussed previously, I use it to increase the boundary width of the boundary output (B_p) and of the boundary ground truth. Our model (*Ours*) gains 1.1% *Dice* and 1.6% *BIoU* improvement upon two segmentation tasks.

(3). I performed extensive experiments to evaluate the significance of boundary nodes and region nodes by removing every element associated with the boundary nodes, including the corresponding *AEM*, V_b, D_p, B_p, L_D, L_B, *etc.*. In this way, the network is devoid of boundary information supervision and produces only region prediction. Furthermore, the proposed *GNN* module can only serve as a cross-level (shallow and deep) feature refinement module for the region segmentation task. It shows that our model (*Ours*) gains 3.5% *Dice* and 3.1% *BIoU* improvement from boundary information supervision on two segmentation

tasks. On the other hand, I remove region information related elements in the network such as the corresponding *AEM*, V_r, D_p, R_p, L_D, L_R, *etc.*, and construct a boundary segmentation network. TABLE 3.3 shows that the model cannot achieve comparatively promising segmentation results due to the lack of supervision over region details. This further demonstrates the importance of boundary and region information in biomedical image segmentation tasks.

Attributes of the Graph Reason Modules. In this section, I present the results of the ablation study on the attributes of the graph reason modules. Here I evaluated the effectiveness of the number of graph reasoning modules (N) and the number of update times (T) in each graph reasoning module. TABLE 3.4 shows that our model achieves the best performance on two segmentation test datasets with two graph reasoning modules ($N = 2$), and each module updates three times ($T = 3$). In detail, the two graph reasoning model tackles (8×8, 16×16) and (8×8, 16×16, 32×32, 64×64) levels' features, respectively.

Furthermore, the number of graph reasoning modules (N) impacts the model size; the number of update times in each graph can influence inference time. To present a comprehensive analysis, I show the inference time and model size with different attributes of the graph reason module in TABLE 3.4. As shown, with $N = 2$ and $T = 1$, our model can run at a real-time speed of ~ 38.1 *fps* and ~ 44.0 *fps* for a 256×256 input of fundus image and colonoscopy image, respectively.

Loss Function. In general, the losses employed in this work serve a variety of purposes. Dice loss [289] ($\mathbf{L}_R$) is a commonly used region-based loss for segmentation task. While Dice loss outperforms other losses (i.e. Cross-Entropy loss) in addressing the unbalanced issues [175], I discover that by using Dice loss for boundary segmentation, the predicted boundary segmentation masks appear to be incomplete, leading to almost black masks

(most zero pixel values) due to the unbalanced foreground and background. I addressed this challenge by applying boundary loss [175] ($\mathbf{L}_B$) to the boundary segmentation predictions (B_p). Boundary agreement loss ($\mathbf{L}_D$) adopts [175] as well. However, it is applied on the derived boundary (D_p), which aims for the consistent boundary upon the region predictions (R_p) and boundary predictions (B_p). L_D brings two essential advantages. Firstly, since D_p and B_p are under the supervision of the same boundary ground truth, L_D can be considered as the consistency loss between the D_p and B_p; at the same time, it can force the model to learn consistent boundary features for region nodes V_r and boundary nodes V_b. Secondly, the L_D serves as a boundary focus on the R_p with additional boundary ground truth supervision. This aids the model to produce more precise boundary predictions.

To analyse the effectiveness of the L_B and L_D, I applied Dice loss [289] to L_B (w/ Dice Loss), which is inevitably vulnerable to unbalanced foreground and background. TABLE 6.10 shows that our model improves by 2.9% *Dice* and 2.7% *BIoU* with boundary loss [175] on two segmentation tasks. Additionally, I excluded boundary agreement loss (L_D) while maintaining the remaining components to verify its importance (w/o Boundary Agreement Loss). As shown, L_D can deliver a 1.7% *Dice* improvement in region segmentation and 1.5% *BIoU* improvement in boundary segmentation.

3.5.2 Clinical Evaluation and 'Failure' Analysis

Clinical Evaluation. As well as assessing computer vision evaluation metrics, I also evaluated the clinical output of our method. The vertical Cup to Disc Ratio (*vCDR*) is an important indicator for screening and diagnosis of glaucoma. The *vCDR* value is calculated by the ratio of vertical cup diameter to vertical disc diameter. A larger *vCDR* indicates a higher possibility of glaucoma and vice versa. Following previous methods [109, 276], I provided the Mean Absolute Error of *vCDR* (δ_{vCDR}) between the predictions and the

ground truth. Our method (*Ours*) achieved 0.056 δ_{vCDR} on the *OC & OD* segmentation test set, which outperformed classic methods *U-Net* [337] (0.089 δ_{vCDR}) and *U-Net++* [508] (0.077 δ_{vCDR}) by 37.1 and 27.3% respectively, outperformed cutting-edge methods *M-Net* [109] (0.064 δ_{vCDR}), *RBA-Net* [276] (0.062 δ_{vCDR}), *Segtran* [210] (0.060 δ_{vCDR}) and *CABNet* [278] (0.067 δ_{vCDR}) by 12.5%, 9.7%, 6.7% and 16.4%. *Ours* provides more accurate *vCDR* estimation than these other methods, and this is consistent with superior segmentation.

'Failure' Analysis. I studied the reasons for poor segmentation by our method, and found that in some cases this could be attributed to imprecise ground truth in public *OD & OC* segmentation datasets. In detail, for each retinal image in the *OD & OC* test dataset, I considered segmentation to have 'failed' when the *Dice* (%) of *OC* segmentation was below 80.0% or *OD* segmentation was below 90.0%. According to these criteria, segmentation failed on 28 out of 613 test images. I made a montage of each case, comprising the original image, our segmentation, and ground truth. I present some of the failed segmentations by using our model (*Ours*) and the ground truth (*GT*) in Fig. 3.5. The ophthalmologist (IJCM) reviewed these 28 montages in a masked manner and indicated which of the two segmentations was more accurate for *OC* and *OD*, respectively. A McNemar-Bowker test [270] confirmed that *Our* segmentation was regarded as clinically accurate significantly more often than the *GT* (p=0.029 for *OC* and p=0.001 for *OD*). Further subjective clinical review of some *GT* image sets suggested variable *GT* accuracy. This highlights the robustness of our model, but also points to important limitations in the ground truth manual annotations. The quality of manual annotations is of utmost importance for developing and validating segmentation models as well as translating automation tools into clinical practice. I advise investigators to apply extra caution when using public datasets. Quality assurance of manual annotations of public datasets is a strategic vulnerability in the field

and requires further work.

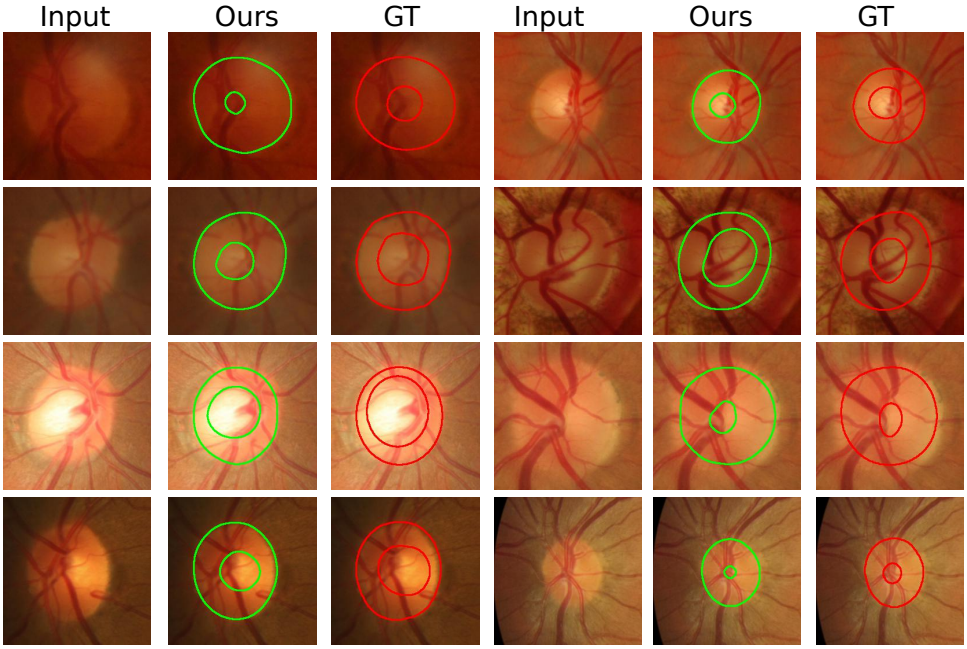


Figure 3.5: A comparison of our segmentation (green) and the ground truth (red) in some 'failed' cases. The ground truth has inaccurate OC boundaries for most of the cases. According to an ophthalmologist (IJCM), our model generally produces more precise boundaries than the ground truth.

3.5.3 Limitation and Future Work

Limitations. Our method achieves promising results for segmenting *OC & OD* and colonoscopy polyps. However, it may not work as well for highly complex objects, such as curvilinear structures like retinal vessels [66, 260, 499]. The primary reason for this is that retinal vessels' region and boundary areas can be challenging to distinguish due to their complex topology and tortuosity. In particular, the derived boundary map (D_p) I propose may have a significant overlap with the region map (R_p) in these situations. Thus, an inevitable perturbation will be included in the information propagation and

message passing process between the region and boundary nodes, harming the segmentation performance.

Future Work. Our method can be extended to tackle video-based segmentation tasks, especially for polyp segmentation. In brief, video-based polyp segmentation methods require high accuracy and speed at the same time. In addition, polyps are of varying size, and their appearance depends on the movement of the camera past the lesion. Thus, dynamic and rapid updates to the receptive field of the network are essential. An extension from our proposed multi-level graph reasoning modules, where each graph is responsible for tackling a specific level of the receptive field, a dynamic attention module (similar to [164]) could be applied on the fusion of different graphs. In this way, our model could automatically adopt the weight contributions between different graphs for inference predictions. As for the inference speed required by video-based tasks, a trade-off between accuracy and speed can be achieved by a different number of graphs and iteration numbers for message passing. Besides this, our proposed model could also be extended to tackle 3D image-based segmentation tasks. In 3D settings, I can regard the boundary as a surface mesh (vertices) and the region as voxels. Thus, the proposed boundary nodes in our method could represent the extracted surface mesh (vertices) features, and the region nodes could represent the extracted voxel-wise features. In this case information exchange and message passing between the surface and volume of 3D objects could be achieved with the same network, simply by redefining the identity of the nodes.

3.5.4 Conclusion

I propose a novel graph-based aggregation module that takes advantage of intuitive associations between the region and boundary features in biomedical images, in order to produce more accurate segmentation. Our experiments have demonstrated that the proposed model

can effectively aggregate and explain the semantic region features and spatial boundary features for segmentation of polyps from colonoscopy images, and the optic disc & optic cup from retinal images. I believe the proposed *GNN* model can also tackle other cross-domain feature reasoning challenges, such as regions, boundaries, and landmark reasoning segmentation tasks.

Chapter 4

Researching Regional and Marginal Consistency with Implicit Graph Representations

In this chapter, I research the geometry-aware graph representation in the task of biomedical image analysis, specifically on the task of glaucoma assessment of optic disc and cup segmentation with colmy fundus images. In detail, glaucoma is a progressive eye disease that results in permanent vision loss, and the vertical cup to disc ratio ($vCDR$) in colmy fundus images is essential in glaucoma screening and assessment. Previous fully supervised convolution neural networks segment the optic disc (OD) and optic cup (OC) from color fundus images and then calculate the $vCDR$ offline. However, they rely on a large set of labeled masks for training, which is expensive and time-consuming to acquire. To address this, I propose a weakly and semi-supervised graph-based network that investigates geometric associations and domain knowledge between segmentation probability maps (PM), modified signed distance function representations ($mSDF$), and boundary re-

gion of interest characteristics (*B-ROI*) in three aspects. Firstly, I propose a novel Dual Adaptive Graph Convolutional Network (*DAGCN*) to reason the long-range features of the *PM* and the *mSDF* *w.r.t.* the regional uniformity. Secondly, I propose a dual consistency regularization-based semi-supervised learning paradigm. The regional consistency between the *PM* and the *mSDF*, and the marginal consistency between the derived *B-ROI* from each of them boost the proposed model's performance due to the inherent geometric associations. Thirdly, I exploit the task-specific domain knowledge via the oval shapes of *OD* & *OC*, where a differentiable *vCDR* estimating layer is proposed. Furthermore, without additional annotations, the supervision on *vCDR* serves as weakly-supervisions for segmentation tasks. Experiments on six large-scale datasets demonstrate my model's superior performance on *OD & OC* segmentation and *vCDR* estimation. The implementation code has been made available [1].

Glaucomatous damage to the optic nerve head can be assessed on colmy fundus images, by measuring the relative size of the optic disc (*OD*) and the optic cup (*OC*) in the vertical direction of the image [307]. Traditionally, a widely adopted method is to calculate the vertical cup to disc ratio (*vCDR*) [109]. Few of the current methods directly regresses the *vCDR* values from fundus images [497]. However, it has lead to the difficulty and uninterpretability in learning [307]. A common pipeline is to segment *OD* and *OC* regions respectively, after which the *vCDR* is calculated as the ratio between the vertical cup diameter and vertical disc diameter. Consequently, accurate segmentation of *OD & OC* is critical for the *vCDR* measurement, in turn for the glaucoma assessment. Recently, numerous deep learning-based segmentation models [109,276,278,280,286,307,445] have been proposed, significantly improving the *OD & OC* segmentation accuracy. However, most of them use a fully supervised paradigm, where a large number of manual delineation labels

[1] https://github.com/smallmax00/Dual_Adaptive_Graph_Reasoning

by clinicians or trained experts are required as the ground truth prior to training the model. The manual annotations are also hugely subjective, time-consuming, laborious, and costly. Solving this problem depends on automated and precise segmentation algorithms that can exploit a large number of unlabeled images without the need for manual delineations. To this end, I proposed a newly designed weakly/semi-supervised learning mechanism that is integrated with my proposed Dual Adaptive Graph Convolutional Network ($DAGCN$). With the critical novelty of exploiting the geometric associations and domain knowledge, I have demonstrated the framework's effectiveness for the segmentation of OD & OC and also glaucoma assessment *w.r.t.* $vCDR$ estimation in colmy fundus images.

The previous segmentation methods concentrated on learning the intensity features of the input images; they would normally rely on a single task such as dense probability map classification, boundary localization, or signed distance function regression. Despite human graders' instinctive use of both image intensity features and spatial relationships between object's boundary and region, they ignore the inherent geometric association between these learned representations, which are critical for improving segmentation performance [72, 286]. To be more precise, segmentation probability map (PM) features emphasize on the global homogeneity of pixel-level semantics and contextual information at the object level. The local boundary characteristics, such as boundary region of interest (B-ROI), describes the spatial variations on both sides of the boundary contour. The signed distance function (SDF) representations emphasize on the global geometry-aware signed distance *w.r.t.* the object contours. Notably, in this work, I propose a modified signed distance function ($mSDF$) that has similar attributes to the SDF but indicates more coherent signals at the semantic level akin to PM. More specifically, the sign label is reversed from the SDF to the proposed $mSDF$ (*e.g.* +, -) for the inner and outer regions of objects in order to make the learned $mSDF$ features need to be coherent with the PM features

for the construction of the dual graph adjacency matrix. Intuitively, the geometric associations between them appears to complement one another during model learning, such as regional and marginal consistency via spatial area and boundary uniformity, thereby improving segmentation performance. To accomplish this, I propose a semi-supervised learning paradigm to construct dual consistency regularizations on both object's region and boundary via the three aforementioned tasks. Additionally, I investigated the method to accompany the feature complementing rationally between *PM* segmentation and *mSDF* regression tasks at semantic and spatial levels. For example, the proposed novel *DAGCN* leverages the advantage of the graph-based model's long-range information propagation and cross-domain feature update capabilities. Specifically, I adaptively constructed the dual graph via initializing the adjacency matrix in a data-dependent way. The estimated vertex embeddings of *mSDF* and *PM* contributed to the dual adjacency matrices adaptively according to the geometric associations between them. I implemented two matrices to quantify the distance and relationship among different vertices so as to achieve adaptive graph construction and reasoning. On the other hand, previous *OD & OC* segmentation-based glaucoma assessment methods have chased high segmentation accuracy but have overlooked the fact that the ultimate goal of such a learning pipeline is to estimate the *vCDR* in order to aid in glaucoma assessment. As a result, the underlying weak supervision label of *vCDR* in *OD & OC* segmentation task is understudied. The previous methods adopted an offline post-processing step to calculate the *vCDR* given the estimated diameters of the *OD & OC*. On the contrary, I have exploited the domain-specific knowledge between the boundary and region in terms of the perimeter and area of an oval shape of *OD & OC*, where a new differentiable *vCDR* estimating layer is proposed for the end-to-end training. Thus, my model does not only avoid any offline post-process to generate *vCDR* but also gains more weakly-supervised guidance without further annotations. Such

a novel design ensured that the proposed model learns the well-defined goals and gains more supervision from the ground truth on both the regions and boundaries of objects. The overview pipeline of my work is depicted in Fig. 6.1, please refer to Fig. 4.2 for more details. In summary, this work makes the following contributions:

- I proposed a dual adaptive graph convolutional network (*DAGCN*) to reason the cross-domain segmentation probability maps and modified signed distance function representations. The information propagation and message exchange *w.r.t.* geometric associations and semantic context were exploited to learn a comprehensive graph representation and adaptive structure.
- I proposed a dual consistency-based paradigm on region and boundary geometric associations in a semi-supervised manner. The enforced consistency on regional and marginal features leads the learned model to a generalizable characteristic learning via leveraging a large amount of unlabeled data.
- For the first time, I exploited the task-specific domain knowledge in terms of perimeter and area of the oval-shaped *OD & OC*, and proposed to estimate the *vCDR* in a differentiable way. Thus, without any further laborious annotations, the supervision on *vCDR* serves as weakly-supervised guidance on the accurate *OD & OC* region and boundary segmentation.

4.1 Related Works

4.1.1 Pixel-wise Medical Image Segmentation

Convolution Neural Network (*CNN*) has found widespread use in the segmentation of medical images. Existing CNN-based methods [109,125,337] have considered segmentation as a dense pixel classification task. For example, the classic *U-net* [337] employs a skip-

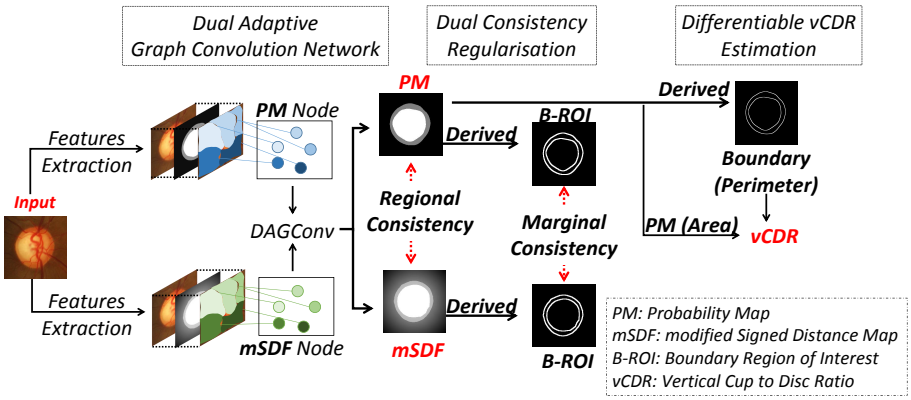


Figure 4.1: Overview of the proposed network, where three major contributions, *DAGCN*, dual consistency regularization and differential *vCDR* estimation, are shown.

connection between the encoder and decoder to minimize information loss. In recent years, it has been used as a baseline model for medical image segmentation tasks. Recently, *Gu et al.* [125] proposed to capture high-level information while preserving spatial information on *OD & OC* segmentation task. However, due to the limited receptive field of standard *CNN*, dense atrous convolutions were incorporated [470] to enlarge the receptive regions for long-range context reasoning. Similarly, *M-Net* [109] requires multi-scale input and side-output mechanisms with deep supervision, to achieve multi-level receptive field fusion for aggregating long-range relationships. With the assistance of the enhanced long-range reasoning abilities, the aforementioned methods achieved promising results in the *OD & OC* segmentation task. They are however inefficient as the stacking of local cues as it does not always accurately represent long-range context relationships [286]. On the contrary, I benefit from the long-range information aggregating ability of the graph-based models to address this issue.

4.1.2 Geometry-aware Medical Image Segmentation

It is well established that boundary knowledge is essential in acquiring geometric features in segmentation tasks. When it comes to medical image segmentation, the boundary accuracy is often more critical than that of the regional pixel-wise coverage [276, 280]. Recent methods, such as [255, 276, 278, 280, 286, 457], have explicitly or implicitly taken into account the geometry dependency between the regions and boundaries of an object of interest in *OD & OC*. Specifically, *Meng et al.* proposed an aggregated hybrid network [286] to jointly learn the relationship between region and boundary of *OD & OC*, conducting an accurate boundary localization. On the other hand, *Luo et al.* [255] and *Xue et al.* [457] adopted *SDF* to represent the target mask in segmentation tasks as it enables the network to learn a distance-aware representation $w.r.t$ the object boundary, emphasizing the spatial perception of the input images. Similarly, I proposed to learn a *mSDF* regression task in this work to exploit the geometry-aware feature learning. Also, it is integrated into the proposed dual consistency semi-supervised paradigm at the task level, leading to a coherent semantic and spatial information integration with *PM* segmentation task in the proposed graph-based model.

Other boundary-based methods [72, 445] integrate the region and boundary geometry constraint into the loss function or evaluation measurement. For example, *Cheng et al.* proposed a Boundary Intersection-over-Union ($BIoU$) [72] evaluation measurement, which quantifies boundary quality in segmentation tasks. *Wu, et al.* [445] proposed an oval shape constraint-based loss function to regularize the contmy shape of the predicted *OD & OC* during learning. Similarly, I exploited the boundary and region relationship in terms of perimeter and area of oval shape to estimate the *vCDR* in a differentiable way. The underlying geometry association of the oval shape of *OD & OC* was researched and specially designed in this work.

4.1.3 Weakly and Semi-supervised Medical Image Segmentation

By learning directly from a small set of labeled data and a large set of unlabeled data, the semi-supervised learning frameworks [202,255,256] achieved high-quality segmentation results. Numerous semi-supervised methods [211, 472] have recently been developed that incorporate unlabeled data through unsupervised consistency regularization. In general, there are majorly two different types of unsupervised consistency regularizations, *i.e.* a data-level of perturbations [211, 382, 472] and a feature-level of perturbations [202, 256]. However, on the other hand, the consistency regularization of task-level in semi-supervised learning has rarely been explored, until very recently in different computer vision tasks, such as crowd counting [284], 3D object detection [250], and 3D medical image segmentation [255]. To be more precise, various levels of information from different task branches can complement one another during training, whereas divergent focuses can lead to inherent prediction perturbation [475]. For example, [284], [255] and [250] all shared a similar idea that the dual task's outputs can be aligned into the same presentation space, and then an unsupervised loss is applied to regularize the consistency. In this work, I have also demonstrated a dual-task level of geometric consistency on the *OD & OC* segmentation. Apart from that, I have integrated the boundary quality into the task-level of consistency regularization.

On the other hand, weakly supervised methods [193, 197, 241, 322] segmented images using image-level of labels [197], bounding boxes [322], points [193], scribbles [241] rather than pixel-by-pixel annotation, which alleviated the burden of annotation. They all focused on the data-driven learning-based way of general coarse labels . For example, given the image-level labels, *Wu et al.* [197] proposed an attention mechanism on the top of the class activation maps [502] to improve 3D brain lesion localization. The estimated lesion regions and normal tissues were then used to train the 3D brain lesion segmentation network.

Differently, for the first time, I integrated the task-specific domain knowledge into the proposed weakly supervised paradigm, where the oval shape of the *OD & OC* is exploited in the segmentation task. As a result, my model could estimate the *vCDR* end-to-end on the basis of *OD & OC* segmentation. At the same time, the information gained from *vCDR* ground truth could weakly-supervise the segmentation process for the both region and boundary of *OD & OC*.

4.1.4 Graph Reasoning in Segmentation

In the recent years, the graph-based models [70, 223, 280, 286, 483] have gained popularity for the segmentation tasks due to their inherent ability to propagate information over long distances and update feature information. *Meng et al.* proposed *RBA-Net* [276] and *CABNet* [278] to regress the *OD & OC* boundaries by aggregated *CNN* and Graph Convolutional Network (*GCN*), which learns the long-range features and directly regresses vertex coordinates in a Cartesian system. The methods described above made use of a Graph Neural Network (*GNN*) to address the challenge of intra-domain long-range feature propagation because messages passing between graph nodes have semantic and spatial characteristics that are similar to one another. Contrary to this, my method treats extracted pixel-level *PM* features and geometry-aware *mSDF* representations as distinct graph nodes and employed *GNN* to learn their inter-domain relationship. In particular, the geometric associations between them were exploited. Additionally, methods such as [70, 223, 276, 278, 483] used *Laplacian* smoothing-based graph convolution [182], provide specific benefits in the sense of global long-range information reasoning. They estimated the initial graph structure from a data-independent *Laplacian* matrix defined by randomly initialized adjacency matrix [70, 483] or hand-crafted adjacency matrix [182, 223, 276, 278]. However, one may enable a model to learn a specific long-range context pattern [215, 280], which is less re-

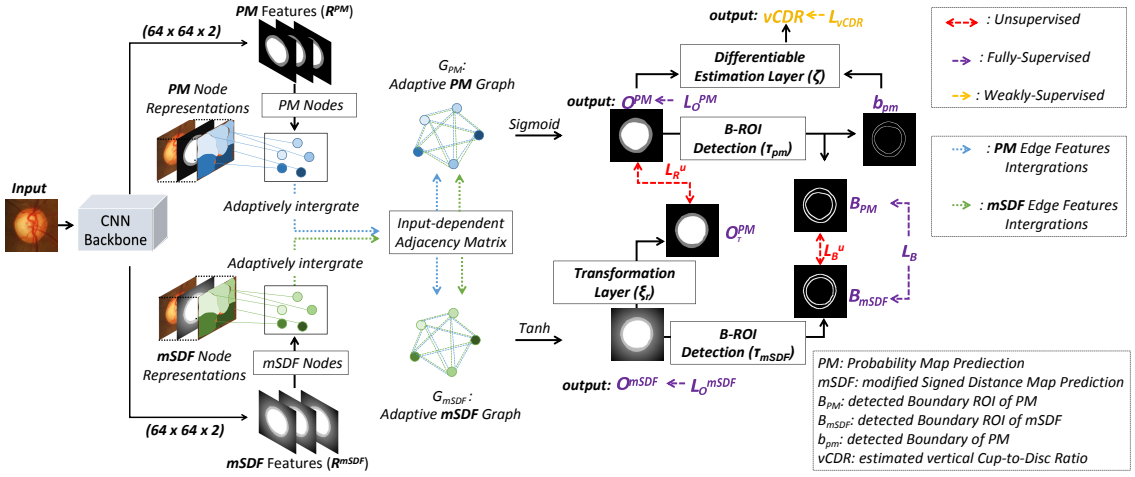


Figure 4.2: Overview of the proposed *DAGCN* model (best viewed in color). O^{PM} and O^{mSDF} both have two channels to represent the output of *OC* and *OD* and I overlapped them for better visualization. L_O^{PM}, L_O^{mSDF}, L_B are the supervised *PM*, *mSDF* and *B-ROI* loss functions; L_{vCDR} is the weakly-supervised *vCDR* loss for *OD* & *OC* segmentation; L_{R^u} and L_{B^u} are the unsupervised region and *B-ROI* consistency losses.

lated to the input features, and thus I considered them as a data-independent non-adaptive graph convolution. Differently, as seen in previous works that the graph structure could be estimated with the similarity matrix from the input data [215], I estimated the initial adjacency matrix in a data-dependent way. The constructed dual graph in this work had two distinct structures, which were adaptively learned from the input features of *PM* and *mSDF* features. Hence, my model was capable of adaptively learning an input-related long-range context pattern, which improved the model segmentation performance; please read *Ablation Study* (Section 6.5.4) for more details.

4.2 Methods

4.2.1 Dual Adaptive Graph Convolutional Network

Graph Node Initialization

A backbone network was used to extract the multi-level features. The deep- and shallow-layer features from different levels complemented one another. For example, the deep-layer features contained extensive semantic region information, while the shallow-layer features retained sufficient spatial boundary information. Thus, for initializing the dual graph vertices, I used the feature aggregation module that is similar to [280] on relative deep-level and low-level features. Specifically, the backbone feature maps of 16 × 16, 32 × 32, and 64 × 64 were aggregated with 1 × 1, 3 × 3 convolutions and bilinear up-sampling operations. Reader are referred to Feature Aggregation Module (*FAM*) in [280] for more details. As a result, following the feature aggregation module, the output feature maps for *PM* (R_{pm}) and *mSDF* (R_{mSDF}) have the same sizes of *64 × 64 × 2*. I then referred them to as the initialised *PM* node embeddings and *mSDF* node embeddings, respectively.

Classic Graph Convolution

I first revisited the classic graph convolution and their graph construction process *w.r.t* the adjacency matrix. Given a graph *G = (V, E)*, normalised *Laplacian* matrix is defined as $L = I - D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$, where I is the identity matrix, A is the adjacency matrix, and D is a diagonal matrix that represents each vertex's degree in V, such that $D_{ii} = \sum_j A_{i,j}$. The *Laplacian* of the graph is a positive semi-definite symmetric matrix, so L can be diagonalized by the Fourier basis $U \in \mathbb{R}^{N\times N}$, such that $L = U\Lambda U^T$. Thus, the spectral graph convolution of i and j can be defined as $i * j = U((U^T i) \odot (U^T j))$ in the Fourier space. The columns of U are the orthogonal eigenvectors $U = [u_1, ..., u_n]$, and

$\Lambda = diag([\lambda_1, ..., \lambda_n]) \in \mathbb{R}^{N \times N}$ is a diagonal matrix with eigenvalues that are not negative. Due to the fact that U is not a sparse matrix, this operation is computationally inefficient. To solve this, it was proposed that the convolution operation on a graph can be defined by formulating spectral filtering [85] with a kernel g_θ using a recursive Chebyshev polynomial in Fourier space. The filter g_θ is parameterized in terms of an order K Chebyshev polynomial expansion, such that $g_\theta(L) = \sum_k \theta_k T_k(\hat{L})$, where $\theta \in \mathbb{R}^K$ is a vector of Chebyshev coefficients, and $\hat{L} = 2L/\lambda_{max} - I_N$ represents the rescaled *Laplacian*. $T_k \in \mathbb{R}^{N \times N}$ is the Chebyshev polynomial of order K. In [182], *Kipf et al.* further simplified the graph convolution as $g_\theta = \theta(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}})$, where $\hat{A} = A + I$, $\hat{D}_{ii} = \sum_j \hat{A}_{ij}$, and θ is the only Chebyshev coefficient left. The corresponding graph *Laplacian* adjacency matrix $\hat{A}$ is hand-crafted, which leads the model to learn a specific long-range context pattern rather than the input-related one [215]. As a result, I refered to the classic graph convolution as data-independent non-adaptive graph convolution.

Dual Adaptive Graph Convolution

This section adopts the similar graph structure *w.r.t* adjacency matrix from my previous works [280]. I extended it into a dual adaptive graph, perfectly fitting the proposed semi-supervised paradigm with dual consistency regularization. Given the initialized *PM* nodes $R_{pm} \in \mathbb{R}^{N \times C}$ and *mSDF* nodes $R_{mSDF} \in \mathbb{R}^{N \times C}$, I constructed the input-dependent adaptive adjacency matrix for the dual adaptive graph (G_{pm} and G_{mSDF}), where C is the channel size; $N = H \times W$ is the number of spatial locations of input feature, which is referred to as the number of vertices.

I illustrate G_{pm} as an example and elaborate the graph construction process as below. Firstly, I implemented two matrices ($\tilde{\Lambda}^c$ and $\tilde{\Lambda}^s$) to perform channel-wise attention on the dot-product distance between input vertex embeddings and to quantify spatially weighted

relations between different vertices, respectively. For example, $\tilde{\Lambda}^c(R_{pm}) \in \mathbb{R}^{C\times C}$ is the matrix containing channel-specific information about the dot-product distance of the input vertex embeddings.; $\tilde{\Lambda}^s(R_{pm}) \in \mathbb{R}^{N\times N}$ is a spatially weighted matrix that quantifies the relationships between different vertices.

$$\tilde{\Lambda}^c(R_{pm}) = \Big(MLP\big(Pool_c(R_{pm})\big)\Big)^{\mathrm{T}} \cdot \Big(MLP\big(Pool_c(R_{pm})\big)\Big), \tag{4.1}$$

where $Pool_c(\cdot)$ denotes the global max pooling for each vertex embedding; $MLP(\cdot)$ is a multi-layer perceptron with one hidden layer. On the other hand,

$$\tilde{\Lambda}^s(R_{pm}) = \Big(Conv\big(Pool_s(R_{pm})\big)\Big) \cdot \Big(Conv\big(Pool_s(R_{pm})\big)\Big)^{\mathrm{T}}, \tag{4.2}$$

where $Pool_s(\cdot)$ represents the global max pooling for each position in the vertex embedding along the channel axis; $Conv(\cdot)$ is a 1×1 convolution layer. In this way, the data-dependent adaptive adjacency matrix $\bar{A}$ is given by spatial and channel attention-enhanced input vertex embeddings. I initialized the input-dependent adaptive adjacency matrix $\bar{A}$ as:

$$\begin{aligned}\bar{A} = \psi(R_{pm}, W_\psi) \cdot \tilde{\Lambda}^c(R_{pm}) \cdot \psi(R_{pm}, W_\psi)^T + \\ \phi(R_{pm}, W_\phi) \cdot \phi(R_{pm}, W_\phi)^T \odot \tilde{\Lambda}^s(R_{pm}),\end{aligned} \tag{4.3}$$

where $\cdot$ represents matrix product; $\odot$ denotes Hadamard product; $\psi(R_{pm}, W_\psi) \in \mathbb{R}^{N\times C}$ and $\phi(R_{pm}, W_\phi) \in \mathbb{R}^{N\times C}$ are both linear embeddings (1×1 convolution); W_ψ and W_ϕ are learnable parameters. Secondly, I exploited the geometric association between *PM* and *mSDF* through integrating *mSDF* into the built *Laplacian* matrix $\tilde{L}$, which allowed us to adaptively built the graph according to their own constraints. Specifically, I fuse it into the spatial-wise weighted matrix $\tilde{\Lambda}^s(R_{pm})$. The geometry-aware spatial weighted matrix

$\tilde{\Lambda}_g^s(R_{pm}, R_{mSDF})$ is given as follows:

$$\begin{aligned}\tilde{\Lambda}_g^s(R_{pm}, B_{mSDF}) = Conv\Big(Pool_s(R_{pm})\Big)\cdot \\ \Big(Conv\Big(Pool_s(R_{pm}+R_{mSDF})\Big)\Big)^T\end{aligned} \tag{4.4}$$

where $Conv(\cdot)$ is a 1×1 convolution layer. In this way, the semantic features of the object's foreground were emphasized by geometry-aware features of *mSDF*. As this is the case, the proposed adaptive graph convolution could take the spatial characteristics into account when reasoning the correlations between different regions. Then, the geometry-aware input-dependent adjacency matrix $\tilde{A}$ will be given as:

$$\begin{aligned}\tilde{A} = \psi(R_{pm}, W_\psi)\cdot\tilde{\Lambda}^c(R_{pm})\cdot\psi(R_{pm}, W_\psi)^T + \\ \zeta(R_{pm}, W_\zeta)\cdot\zeta(R_{pm}, W_\zeta)^T \odot \tilde{\Lambda}_g^s(R_{pm}, R_{mSDF}),\end{aligned} \tag{4.5}$$

where $\zeta(R_s, W_\zeta) \in \mathbb{R}^{N\times C}$ is 1×1 convolution; W_ζ is learnable parameter. With the constructed $\tilde{A}$, the normalized *Laplacian* matrix is given as $\tilde{L} = I - \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$, where I is the identity matrix, $\tilde{D}$ is a diagonal matrix that represents the degree of each vertex, such that $\tilde{D}_{ii} = \sum_j \tilde{A}_{i,j}$. I calculated degree matrix $\tilde{D}$ with the same way that is used in [215, 280], to override the computation overhead. Given computed $\tilde{L}$, with R_{PM} as the input vertex embeddings, I formulate the single-layer $DAGConv$ as :

$$Y = \sigma(\tilde{L}\cdot R_{pm}\cdot W_G) + R_{pm}, \tag{4.6}$$

where $W_G \in \mathbb{R}^{C\times C}$ denotes the trainable weights of the $DAGConv$; σ is the ReLu activation function; Y is the output vertex features. Moreover, I add a residual connection to reserve the features of input vertices.

Please note that the graph construction and convolution process of G_{mSDF} is similar

to G_{pm}, where the only difference is to replace R_{PM} to R_{mSDF} or reverse the position of R_{PM} and R_{mSDF}, from Eq. 4.1 to Eq. 5.12. In that case, the semantic features of *PM* is adaptively integrated into the geometry-aware *mSDF* during the graph construction of G_{mSDF}. As a result, the proposed *DAGCN* consists of two adaptive graphs (G_{pm} and G_{mSDF}), to reason the pixel-wise *PM* features and geometry-aware *mSDF* representations respectively and concurrently, with the benefits of their underlying geometric associations.

After the *DAGConv* (Eq. 5.12) in graph G_{pm} and graph G_{mSDF}, I apply bilinear upsampling layers to scale the feature map in dual graph to the same size as input image. Then the *Sigmoid* and *Tanh* activation function were used to generate the *PM* output (O^{PM}) and *mSDF* output (O^{mSDF}) respectively. I then applied *Dice* loss (L_O^{PM}) and *MSE* loss (L_O^{mSDF}) on O^{PM} and O^{mSDF} respectively for all of the labeled input data, to supervise the dual regional predictions.

4.2.2 Dual Consistency Regularization of Semi-supervised Manner

Modified Signed Distance Function (*mSDF*)

Given O^{PM} and O^{mSDF}, I explored the geometric association between them and build the unsupervised dual consistency regularization losses via two differentiable transformation layers (ξ_r and τ). As mentioned above, various levels of information from different task branches can complement one another during training, whereas divergent focuses can lead to inherent prediction perturbation. The dual consistency regularization imposed the regional and marginal consistency in the task level in a semi-supervised manner. Given a

target object (OD or OC), the mSDF is defined as:

$$mSDF(x) = \begin{cases} 1, & x \in B_{in} \\ 0, & x \in \Delta B \\ -\underset{y \in \Delta B}{inf} ||x - y||_2, & x \in B_{out} \end{cases} \tag{4.7}$$

where $||x - y||_2$ represent the Euclidean distance between pixel x and y. Besides, B_{out}, B_{in} and ΔB denote the outside, inside, and boundary of the object, respectively. In other words, the absolute value of $mSDF(x)$ represented the distance between the point and the nearest point on the object's boundary, whereas the sign indicates whether the point is inside or outside the object. The differences between standard SDM and my proposed $mSDF$ are twofold. Firstly, the $mSDF$ has a reversed sign label against SDF because the learned $mSDF$ features are used to build adjacency matrix along with PM features to learn a dual adaptive graph ($DAGCN$), it needs to have the similar feature space to the PM features before activation function (*e.g.* $R^{mSDF}(x) \rightarrow +\infty$, if $x \in B_{in}$). Secondly, I set the distance value of the inside region of $mSDF$ to 1, for the ease of building regional consistency (Eq. (8)) between PM and $mSDF$. However, the proposed $mSDF$ still has the similar attribute as the standard SDF to learn distance-aware spatial features. In this way, dual tasks can acquire the coherent semantic features, meanwhile the $mSDF$ regression task benefits from the distance-aware spatial information supervision.

Regional Consistency

As for region-wise consistency, similar to [255, 284, 457], I proposed a transformation layer to convert the O^{mSDF} to O^{PM} in a differentiable way. To be precise, the region-wise

transformation layer ξ_r is defined as:

$$\xi_r(z) = 2 * Sigmoid(K \cdot ReLu(z)) - 1, \tag{4.8}$$

where z denotes the $mSDF$ value at pixel x; K is a very large value; *Sigmoid* and *ReLu* are the non-linear activation functions. The larger K value indicates a closer approximation, and it is adopted as 5000 in this work. With Eq. 4.8, I could obtain the transformed segmentation maps O_T^{PM}, for example, $O_T^{PM} = \xi_r(O^{mSDF})$. For all of the unlabeled input, I applied a *Dice* loss (L_{R^u}) between O^{PM} and O_T^{PM} to enforce the unsupervised regional consistency regularization.

Marginal Consistency

I derived the spatial gradient of O^{PM} and O^{mSDF} as the estimated contours concerning the boundary-wise consistency. Previous studies [72, 286] have proven that such narrow contours with a width of one pixel are challenging to optimize due to the highly unbalanced foreground and background, resulting in weakened consistency regularizations. Rather than focusing exclusively on the thin contmy locations, I considered the *ROI* within a certain distance (boundary width) of the corresponding estimated contours. A simple yet efficient *B-ROI* detection layer (τ) is proposed for O^{PM} and O^{mSDF}. For example, τ_{PM} and τ_{mSDF} are defined as :

$$\tau_{PM} = O^{PM} + Maxpooling2D(-O^{PM}), \tag{4.9}$$

$$\tau_{mSDF} = \xi_r(O^{mSDF}) + Maxpooling2D\big(-\xi_r(O^{mSDF})\big), \tag{4.10}$$

The *Maxpooling2D* operation conducts the same feature map size as its input. It is worth noting that the output width of τ can be determined by varying the kernel size, stride,

and padding value of the Maxpooling2D operation. I empirically set the output boundary width of τ_{PM} and τ_{mSDF} to 4 pixels in this work. After τ_{PM} and τ_{mSDF}, I referred to such *B-ROI* of O^{PM} and O^{mSDF} as B_{PM} and B_{mSDF}, respectively. Ideally, B_{PM} and B_{mSDF} should be close enough to one another. Thus, a *Dice* loss (L_{B^u}) between B_{PM} and B_{mSDF} was applied to enforce the unsupervised marginal consistency regularization of unlabeled data. Meanwhile, I apply a *Dice* loss (L_B) on both B_{PM} and B_{mSDF} to supervise the dual boundary predictions of labeled data.

4.2.3 Differentiable *vCDR* estimation of Weakly Supervised Manner

Because the shapes of *OD* & *OC* are oval-like [307], previous methods resort to offline post-process the segmentation predictions with ellipse fitting to improve the segmentation accuracy [109], or to calculate *vCDR* using the approximated diameters of the *OD* & *OC* in the long axis [276, 278, 286]. However, they only use *vCDR* as an evaluation tool for glaucoma assessment but overlook the underlying supervision value of it in *OD* & *OC* segmentation task. Additionally, in the real world setting of clinical ophthalmology and ophthalmic image reading centres, clinicians and graders prefer to calculate the *vCDR* value with manually measured diameters of the *OD & OC* on the long axis, rather than to delineate the contmy of *OD & OC* then calculating the *vCDR*, to save time. This results in a large number of labeled data with *vCDR* scalars; however, they have not been exploited in the computer vision community yet. For example, one of the datasets I used in this work (*UKBB*) contains 117,832 images with *vCDR* ground truth labeled. To address this issue, I took advantage of the specific domain knowledge between the boundary and region in terms of the perimeter and area of an oval-like shape to approximate the *vCDR* in a differentiable way.

To be precise, the *vCDR* is defined as the ratio of dividing the measured diameters of

the cup by disc in the long axis. While such ratio can also be estimated given the size of perimeter and the area of *OD* and *OC*. According to the *Euler's Method* [246], the area (A_o) and perimeter (P_o) of the oval shape are defined as:

$$A_o = \pi \cdot a \cdot b, \tag{4.11}$$

$$P_o = \pi \cdot \sqrt{2(a^2 + b^2)}. \tag{4.12}$$

where a and b denote the semi-axis of the long and short axis of oval shape, respectively. I approximated A_o with the summed pixel value of O^{PM}, which can be regarded as the area of oval shape in pixel level. Furthermore, I derived the spatial gradient of O^{PM} via the *B-ROI* detection layer (τ_{PM}), to detect the boundary (b_{pm}) with width = 1. Then the summed pixel values of b_{pm} was approximately regarded as P_o. With Eq. 4.11 and Eq. 4.12, I could approximate a with A_o and P_o, such as:

$$a = \sqrt{\frac{(P_o)^2 + \sqrt{(4\pi A_o + (P_o)^2) \cdot |((P_o)^2) - 4\pi A_o)|}}{4\pi^2}}), \tag{4.13}$$

where $|\cdot|$ is used to prevent sqrt from returning a negative value during the initial learning period. Given Eq. 4.13, I could calculate the *OD* long semi-axis (a^{OD}) and the *OC* long semi-axis (a^{OC}) with the respective P_o and A_o. Then, a *vCDR* estimation layer ζ was defined as:

$$\zeta(vCDR) = \frac{a^{OC} + e^{-6}}{a^{OD} + e^{-6}}, \tag{4.14}$$

where, e^{-6} is added to avoid dividing by zero errors. Given the prediction of *vCDR*, I apply a *MSE* loss (L_{vCDR}) between the prediction and ground truth to fully supervise the *vCDR* estimation and weakly-supervise the *OD* & *OC* segmentation.

Table 4.1: Quantitative segmentation results of *OD & OC* and glaucoma assessment on *SEG* testing datasets. The performance is reported as *Dice* (%), *BIoU* (%), *MAE*, and *Corr*. 95% confidence intervals are presented in brackets, respectively. I compare my model with previous state-of-the-art fully-supervised methods by running their codes in the public domain. The implementation of the compared state-of-the-art semi-supervised works is mainly based on an open-source codebase [254]. *Ours (Semi)* achieves statistically significant improvements consistently over other compared semi-supervised methods; please refer to TABLE. 4.2 for more details. Up and down arrows represent proportional and inversely proportional metric value and performance correlations.

Methods	SEG (OC)		SEG (OD)		*SEG (vCDR)*		*UKBB (vCDR)*	
	Dice (%)↑	*BIoU(%)*↑	*Dice (%)*↑	*BIoU(%)*↑	*MAE* ↓	*Corr* ↑	*MAE* ↓	*Corr* ↑
U-Net [337]	85.3 (82.1, 86.8)	80.1 (77.6, 82.4)	95.0 (93.1, 97.1)	86.2 (84.1, 88.3)	0.089 (0.079, 0.095)	0.685 (0.643, 0.713)	0.150 (0.140, 0.158)	0.301 (0.275, 0.329)
M-Net [109]	86.9 (85.0, 88.0)	82.9 (79.5, 84.7)	96.8 (95.5, 97.6)	88.1 (87.0, 89.3)	0.064 (0.051, 0.073)	0.707 (0.668, 0.741)	0.128 (0.119, 0.140)	0.365 (0.337, 0.390)
GRBNet [286]	89.4 (87.6, 90.8)	85.1 (83.3, 86.8)	97.7 (97.0, 98.7)	91.1 (90.2, 92.0)	0.056 (0.043, 0.067)	0.750 (0.739, 0.764)	0.118 (0.094, 0.134)	0.398 (0.371, 0.415)
RBA-Net [276]	87.8 (85.2, 89.7)	83.8 (81.6, 85.9)	96.1 (95.5, 96.7)	88.9 (88.0, 89.2)	0.062 (0.051, 0.073)	0.713 (0.690, 0.734)	0.126 (0.109, 0.142)	0.369 (0.350, 0.373)
MT [382]	84.1 (81.8, 85.7)	78.2 (77.0, 79.6)	94.3 (94.0, 94.7)	86.5 (85.0, 87.3)	0.091 (0.080, 0.099)	0.683 (0.641, 0.701)	0.145 (0.139, 0.150)	0.307 (0.276, 0.340)
UAMT [472]	85.3 (82.8, 86.9)	80.2 (79.0, 81.7)	95.2 (94.7, 95.6)	86.4 (85.1, 87.7)	0.075 (0.063, 0.081)	0.692 (0.642, 0.723)	0.134 (0.127, 0.139)	0.339 (0.301, 0.361)
URPC [256]	86.1 (83.1, 87.2)	81.2 (79.6, 82.0)	96.0 (95.4, 96.3)	87.3 (85.0, 87.9)	0.067 (0.059, 0.073)	0.701 (0.659, 0.742)	0.126 (0.121, 0.135)	0.361 (0.337, 0.382)
DTCNet [255]	86.1 (83.0, 87.4)	81.1 (79.5, 82.8)	96.1 (95.3, 96.4)	87.0 (85.2, 87.8)	0.065 (0.060, 0.072)	0.703 (0.661, 0.739)	0.126 (0.120, 0.137)	0.364 (0.339, 0.389)
UDCNet [202]	86.2 (83.3, 87.1)	81.4 (79.6, 83.0)	96.2 (95.7, 96.5)	87.1 (85.6, 87.9)	0.067 (0.059, 0.071)	0.714 (0.663, 0.742)	0.127 (0.119, 0.135)	0.389 (0.365, 0.412)
SASSNet [211]	85.8 (82.1, 87.3)	80.6 (78.2, 82.9)	95.7 (94.1, 96.5)	86.5 (85.4, 87.6)	0.070 (0.061, 0.079)	0.695 (0.633, 0.741)	0.139 (0.118, 0.153)	0.340 (0.313, 0.368)
Ours (Semi-100%)	**90.3** (89.6, 90.8)	**87.6** (83.6, 90.8)	**98.4** (98.4, 98.5)	**93.3** (92.1, 94.9)	**0.037** (0.035, 0.041)	**0.894** (0.863, 0.918)	**0.075** (0.073, 0.078)	**0.558** (0.514, 0.583)
Ours (Semi)	**88.2** (87.5, 88.9)	**84.1** (81.0, 87.6)	**97.6** (97.5, 97.8)	**89.9** (88.8, 90.7)	**0.047** (0.044, 0.051)	**0.848** (0.809, 0.879)	**0.097** (0.094, 0.099)	**0.463** (0.447, 0.480)

4.3 Experiments

4.3.1 Datasets

SEG dataset: following the previous methods [280, 286], I pooled 2,068 images from five public available datasets (Refuge [307], Drishti-GS [370], ORIGA [490], RIGA [6], RIM-ONE [111]). These five datasets provided the fundus images and the ground truth masks, then I generated the corresponding ground truth of O^{mSDF}, B_{PM}, B_{mSDF} and $vCDR$ with Eq. 4.7, 4.9, 4.10 and 4.14. Following the previous methods [280, 286], 613 fundus images were randomly selected as the test dataset, leaving the other 1,315 images for training and 140 images for validation.

UKBB dataset: The UK Biobank [2] is a large-scale population-based biomedical database and research resource that contains detailed health information on half a million participants from the United Kingdom. Retinal colmy photographs were acquired in a subset of participants that were scanned using the TOPCON 3D OCT 1000 Mk2 camera (Topcon Inc, Japan). The color fundus photographs have been graded for various eye diseases by NetwORC UK, a network of three UK Ophthalmic Reading Centers (Moorfields, Queen University of Belfast, and Liverpool) to support further scientific research on this invaluable dataset. First and foremost, the accredited graders evaluated the image quality to determine whether it was sufficient for measuring the $vCDR$. Then $vCDR$ was calculated by dividing the measured diameter of the cup by the measured diameter of the disc in the long-axis or vertical direction. There were 117,832 fundus images with $vCDR$ scalars are available, of which 38,421 were randomly selected as the weakly/semi-supervised training dataset, and the rest 79,411 are used as the test datasets.

[2] https://www.ukbiobank.ac.uk/

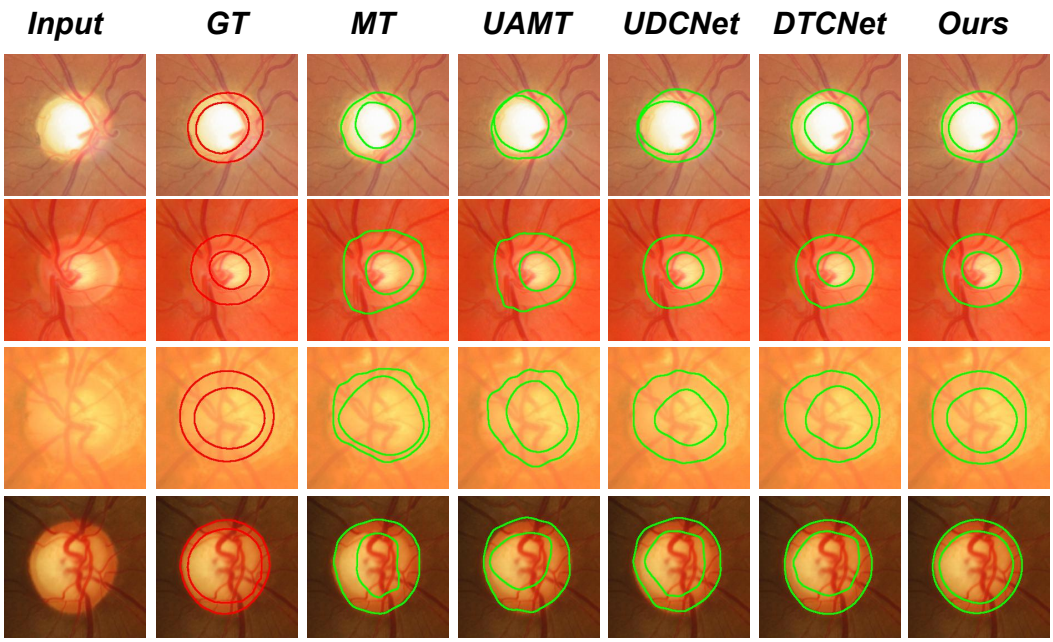


Figure 4.3: Qualitative results of *OD & OC* segmentation in the *SEG* test dataset. I compare my model with *MT* [382], *UAMT* [472], *UDCNet* [202] and *DTCNet* [255]. my method can produce more accurate segmentation results than the other methods when compared with the ground truth (*GT*). The boundaries were superimposed on the input image for better visualization of the segmentations.

Table 4.2: Paired t-test results between *Ours (Semi)* and the compared semi-supervised methods. I presented the *p*-value of the mean *Dice* of *OD & OC* segmentation on *Seg* test dataset; the mean *MAE* of *vCDR* estimation on *UKBB* test dataset; the mean *AUROC* of glaucoma diagnosis on *ORIGA*, *RIM-ONE*, *Refuge* test datasets; the mean *Dice* of *polyps* segmentation on colonoscopy polyps test dataset. Because my model achieves consistently better performance than the other compared semi-supervised methods on the fmy tasks, the *p*-value demonstrates that *Ours (Semi)* achieves statistically significant improvements consistently over other compared semi-supervised methods.

Tasks:	*Ours (Semi)* vs others	*MT* [382]	*UAMT* [472]	*URPC* [256]	*DTCNet* [255]	*UDCNet* [202]	*SASSNet* [211]
OD & OC	*p*-value (on *Dice*)	0.014	0.021	0.039	0.041	0.033	0.019
vCDR	*p*-value (on *MAE*)	0.018	0.029	0.040	0.044	0.036	0.020
Diagnosis	*p*-value (on *AUROC*)	0.009	0.021	0.033	0.037	0.029	0.011
Colonoscopy polyps	*p*-value (on *Dice*)	0.010	0.028	0.041	0.024	0.031	0.013

4.3.2 Experimental Settings and Evaluation Metrics

I cropped the image of 256 × 256 pixels in the same way of [276, 280, 286]. To avoid over-fitting, I adopted an on-the-fly data augmentation strategy. Specifically, I randomly flipped the training dataset with a probability of 0.5. I used stochastic gradient descent with a momentum of 0.9 to optimize the overall parameters. I trained the model for 10,000 iterations for all the experiments, with a learning rate of 1e-2 and a step decay rate of 0.999 every 100 iterations. The batch size was set as 56, consisting of 28 labeled and 28 unlabeled images. A backbone network [114] is used for ours and all the compared methods. The network was trained end-to-end; all the training processes were performed on a server with fmy *GEFORCE RTX 3090 24GiB GPUs*, and all the test experiments were conducted on a workstation with *Intel(R) Xeon(R) W-2104 CPU* and *Geforce RTX 2080Ti GPU* with 11GB memory. I used the output of the *PM* as the segmentation result. A fixed threshold of 0.5 is employed to obtain a binary mask from the probability map. Given the previously discussed loss function terms, I defined the overall loss function as:

$$Loss = L_O^{PM} + L_O^{mSDF} + L_B + \beta * (L_{R^u} + L_{B^u} + L_{vCDR}) \tag{4.15}$$

where β is adopted from [307] as the time-dependent Gaussian ramp-up weighting coefficient to trade-off between the supervised loss, unsupervised loss and weakly-supervised loss. This avoids the network getting stuck in a degenerated solution during the initial training period because no meaningful prediction of the unlabeled data, as well as *vCDR*, are obtained.

I reported Dice similarity score (*Dice*) as the region segmentation accuracy metrics; Boundary Intersection-over-Union (*BIoU*) [72] as the boundary segmentation metrics; and Mean Absolute Error (*MAE*) in pixel level, Pearson's correlation coefficients [297] (*Corr*),

Bland-Altman analysis [31] as the *vCDR* estimation metric. 95% confidence intervals were generated by using 2,000 sample bootstrapping. Note that the Pearson's correlation coefficients [297] are used to measure the linear association. Paired t-test was used to assess statistical significance of the differences between my model and the compared methods. A p-value of < 0.05 was deemed as statistically significant.

4.3.3 Performance Comparison and Analysis

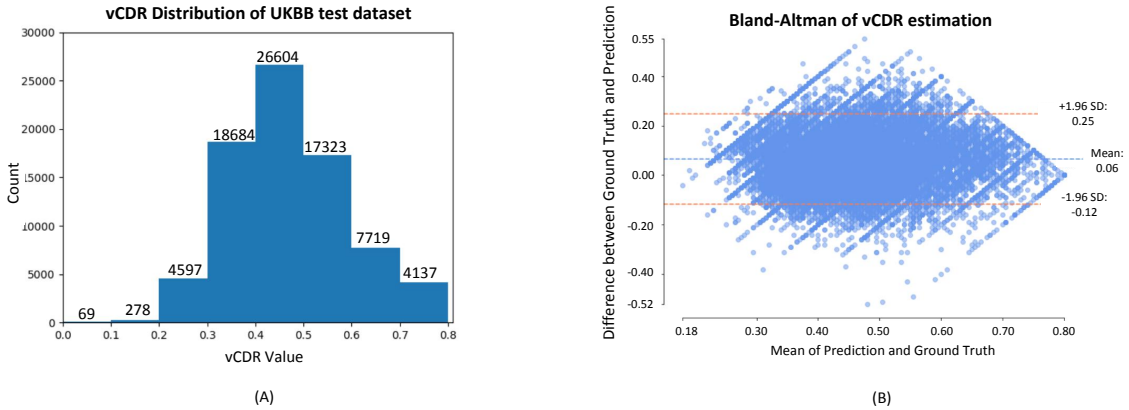
In this section, I demonstrate the qualitative (Fig. 4.3) and quantitative (TABLE. 4.1) results of the *OD & OC* segmentation and glaucoma assessment tasks. Specifically, in TABLE. 4.1, I have presented the results of fully-supervised methods on the upper half part, and the rest are semi-supervised methods. All the fully-supervised methods were trained with 100% of the labeled *SEG* training dataset, and all the semi-supervised methods were trained with 5 % of *SEG* training dataset and 100 % of *UKBB* training dataset. In order to conduct complementary experiments, I trained my model with 100 % *SEG* and 100 % *UKBB* training data to fully utilise the available labeled and unlabeled data (*Ours (Semi-100%)*).

***OD & OC* Segmentation** Fig. 4.3 illustrates qualitative comparison with other semi-supervised methods on *SEG* test dataset. TABLE. 4.1 shows the quantitative performance of *Ours* and other methods under fully-supervised and semi-supervised manner, respectively. More experimental results for the data utilization efficiency can be found in Section 6.5.4.

With only 5 % labeled segmentation training data, *Ours (Semi)* obtains an average 92.9 % *Dice* on *OC* and *OD* segmentation, outperforms data-level consistency regularization based methods *MT* [382], *UAMT* [472] by 4.2 % and 2.9 %, outperforms feature-level regularization based methods *URPC* [256] and *UDCNet* [202] by 2.0 % and 1.9 %, and

Table 4.3: Number of parameters and *FLOPs* on a 256 × 256 input image.

	M-net [109]	*RBA-Net* [276]	*GRBNet* [286]	*MT* [382]	*UAMT* [472]	*URPC* [256]	*DTCNet* [255]	*SASSNet* [211]	***Ours (Semi)***
Params (*M*)	27.7	34.3	24.7	26.3	26.3	27.2	26.7	29.5	28.6
FLOPs (*G*)	15.5	130.3	7.1	5.5	5.5	7.3	5.5	10.3	9.1



(A) (B)

Figure 4.4: (A): The *vCDR* distribution histogram of the *UKBB* test dataset. In total, there are 79,411 testing images with corresponding *vCDR* ground truth ranging from 0 to 0.8. (B): Bland-Altman plot of *vCDR* estimation for *Ours (Semi)* in *UKBB* test dataset. The x-axis and y-axis represents the mean and difference between ground truth and predicted *vCDR* value, respectively. The mean offsets and the limits of agreement, as well as the 95 % confidence interval on the mean values are shown.

outperforms adversarial regularization based method *SASSNet* [211] by 2.3 %. Paired t-tests on average *Dice* of *OD & OC* segmentation between *Ours (Semi)* and other semi-supervised methods were conducted to evaluate the statistical significance in the difference. The proposed method achieves statistically significant improvements consistently over other compared semi-supervised methods. Readers are referred to Table. 4.2 for the details. Distinctively, with sufficient labeled and unlabeled data, *Ours (Semi-100%)* achieved the best performance of averaged 94.4 % *Dice* on *OD & OC* segmentation, outperforming previous fully-supervised cutting-edge methods, such as *M-Net*, *RBA-Net* and *GRBNet* [286] by 2.7 %, 2.6% and 0.9 %.

Clinical Evaluation: vCDR Assessment TABLE. 4.1 illustrates the *vCDR* evaluation

results on *SEG* and *UKBB* test dataset respectively. The *UKBB (vCDR)* has 79,411 images, which is much larger than *SEG (vCDR)* (619 images). The performance on *UKBB (vCDR)* could reflect a more realistic situation in the real-world *w.r.t.* data distribution. Specifically, with only 5 % labeled *SEG* training data, *Ours (semi)* achieved the best performance of 0.097 *MAE* and 0.463 *Corr*, which outperforms *DTCNet* [255] by 23.0 % and 53.3 %. Paired t-tests on the *MAEs* of *vCDR* estimation between *Ours (Semi)* and other semi-supervised methods in Table. 4.2 were conducted to evaluate the statistical significance in the difference. Please note that, I utilised 38421 images of *UKBB* training dataset with the corresponding *vCDR* ground truth for weakly-supervised *OD & OC* segmentation and fully supervised *vCDR* estimation. On the other hand, with 100 % labeled *SEG* training dataset, *Ours (Semi-100%)* achieved much better performance with 0.075 *MAE* and 0.558 *Corr*, which is 22.7 % and 20.5 % better than *Ours (Semi)*. Additionally, the direct *vCDR* regression-based method [497] with all *UKBB* train data achieved 0.074 *MAE* but only 0.240 *Corr* on the *UKBB* test data. As the distribution of glaucoma patients and health participants are unbalanced, thus such regression model tends to predict closer to the majority of the distribution. The distribution of *vCDR* ground truth in *UKBB* test dataset is shown in Fig. 4.4 (A) for a better understanding of the data and my model's performance. In total, there were 79,411 test images with corresponding *vCDR* ground truth ranging from 0 to 0.8. It illustrated that the majority of *vCDR* ground truth distribution fell between 0.3 and 0.7. On the other hand, in order to evaluate mean biases and 95 % limit of agreements of estimated *vCDR*, a Bland-Altman plot [31] for *UKBB* test dataset was conducted and shown in Fig. 4.4 (B). The mean value of the offsets was 0.06, and the 95 % confidence interval was 0.18, which indicated a close agreement and minimal bias between the ground truth and my predictions. The bias occurs mainly for a value within the range of 0.3 to 0.7 in the majority of data distribution. However, my model performs

well when *vCDR* is small or big (*e.g.* less than 0.3 or larger than 0.7), where little bias cases are observed.

4.3.4 Computational Efficiency

Tab. 6.4 demonstrates the number of parameters (M) and floating-point operations (*FLOPs*) of the compared models. *Ours (Semi)* and other compared models adopted the same backbone network, thus showing similar model size (*Params*). While, *RBA-Net* [276] has the largest model size and *FLOPs* because it contains several iterative feature aggregation modules, which requires more computations. *Ours (Semi)* contains $28.6M$ parameters and $9.1G$ *FLOPs*, which is comparable to other compared models.

4.4 Discussion and Conclusion

4.4.1 Ablation Study

I conducted detailed ablation studies with 5 % *SEG* training data and 100 % *UKBB* training data, and all the results demonstrated my model's effectiveness. As an illustration, the ablation results for different graph reasoning modules, weakly/semi-supervisions, and the efficiency analysis of data utilization are shown in TABLE. 4.4, TABLE. 4.5 and Fig. 4.5.

Graph Reasoning In this section, I assessed the efficacy of the proposed *DAGCN*. Notably, I maintained the same dual graph structure while experimenting with various graph construction methods (via adjacency matrix) and graph convolutions. To begin, I use the classic graph convolution [182] to reason about the relationships between the *PM* and the *mSDF*, respectively. Then, I investigated input-dependent graph convolutions in terms of channel attention (*w/ Channel*) and spatial attention (*w/ Spatial*) mechanisms, both

Table 4.4: Ablation study on graph convolutions. The performance is reported as *Dice* (%), *BIoU* (%), *MAE* and *Corr* on two test datasets. The best results are highlighted in bold.

Methods	SEG (OC)		SEG (OD)		UKBB (vCDR)	
	Dice (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑	*MAE* ↓	*Corr* ↑
Classic GCN [182]	85.9	80.4	95.7	85.9	0.149	0.323
w/ Channel	86.8	82.8	95.8	86.8	0.121	0.349
w/ Spatial	87.1	83.0	96.0	87.1	0.109	0.407
w/ Both	87.6	83.4	96.6	87.8	0.108	0.411
w/ *SGR* [223]	87.2	83.6	96.5	87.7	0.105	0.430
w/ *DualGCN* [483]	87.5	83.7	96.6	88.1	0.104	0.427
w/ *GloRe* [70]	87.4	83.6	96.7	88.4	0.106	0.429
Ours (Semi)	**88.2**	**84.1**	**97.6**	**89.9**	**0.097**	**0.463**

separately and concurrently (*w/ Both*). Additionally, I adopt three more powerful graph reasoning modules to demonstrate the superiority of my proposed *DAGCN*. In particular, I use the *SGR* [223], *DualGCN* [483], and *GloRe* module [70] respectively. In detail, the *SR* module exploits knowledge graph mechanism; *DualGCN* investigates the coordinate space and feature space graph convolution; and *GloRe* leverage projection and re-projection mechanism to reason the semantics between different regions. Note that the methods mentioned above belong to single graph reasoning; thus, I have built two separate graphs for *PM* segmentation and *mSDF* regression individually, where there was no associations or geometric associations between the dual graph. TABLE. 4.4 shows that my model achieved more accurate and reliable results than [182] and outperformed the *SGR* [223], *DualGCN* [483], and *GloRe* [70] by 1.1 %, 0.9 % and 0.9 % mean *Dice* on the *SEG* test datasets.

Weakly/Semi-supervision I performed experiments to evaluate the effectiveness of the proposed dual consistency regularization paradigm in semi-supervised learning and the

Table 4.5: Ablation study on weakly/semi-supervisions. The performance is reported as *Dice* (%), *BIoU* (%), *MAE* and *Corr* on two test datasets. The best results are highlighted in bold.

Methods	SEG (OC)		SEG (OD)		UKBB (vCDR)	
	Dice (%)↑	*BIoU* (%)↑	*Dice* (%)↑	*BIoU* (%)↑	*MAE* ↓	*Corr* ↑
w/o L_{R^u}	86.1	80.9	96.3	86.9	0.146	0.326
w/o L_{B^u}	86.5	81.7	96.5	87.4	0.131	0.345
w/ Both	86.8	82.6	96.8	88.4	0.123	0.348
w/ L_{vCDR}	87.1	82.9	96.7	88.8	0.108	0.415
w/ L_{B^u} +L_{vCDR}	87.3	83.3	96.9	88.9	0.106	0.434
w/ L_{R^u} +L_{vCDR}	87.4	83.2	97.1	89.1	0.102	0.443
Ours (Label-only)	80.5	70.7	91.6	75.8	0.628	0.118
Ours (Semi)	**88.2**	**84.1**	**97.6**	**89.9**	**0.097**	**0.463**

proposed differentiable *vCDR* estimation module in a weakly-supervised manner. The results are shown in TABLE. 4.5. Specifically, I evaluated the region-wise consistency loss, the boundary-wise consistency loss, and the *vCDR* estimation loss, respectively. I have represented my model that is trained with only 5 % *SEG* training data as *Ours (Label-only)*. Firstly, I have retained the same model structure and eliminate the *vCDR* estimation loss to focus on the dual consistency regularization losses (*w/ Both*). Following that, I have removed the region-wise unsupervised loss (*w/o* L_{R^u}), boundary-wise unsupervised loss (*w/o* L_{B^u}) respectively. Secondly, I removed both of the consistency losses and only applied the weakly-supervised *vCDR* estimation loss (*w/* L_{vCDR}). Then I added the other two unsupervised consistency losses individually (*w/* L_{B^u} +L_{vCDR} and *w/* L_{R^u} +L_{vCDR}) to see if the performance were boosted. TABLE. 4.5 demonstrates that the proposed unsupervised dual consistency losses and weakly supervised loss could improve the model by 6.6 % and 6.5 % mean *Dice* for segmentation. Particularly, the boundary-wise unsupervised loss can increase the model by 6.2 % *BIoU*, which leads to a better boundary segmentation

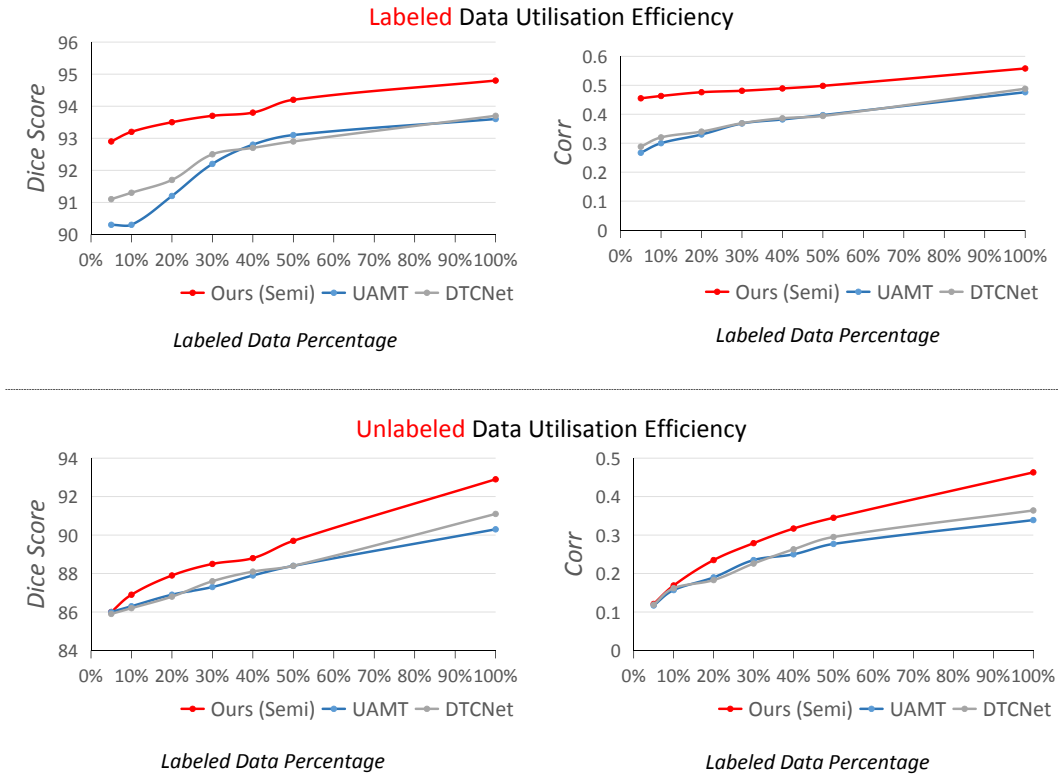


Figure 4.5: The mean *OD* & *OC* segmentation performance of my semi-supervised approach with different ratio of labeled data. The performance is reported with *Dice* and *Corr*.

quality. The weakly supervised loss can bring a large improvement of 82.8 % *MAE* of *vCDR* estimation, which is the ultimate goal for *OD* & *OC* segmentation task *w.r.t* clinic application.

Data Utilization Efficiency In this section, I show more ablation study results on the data utilization efficiency. In detail, I have examined the performance of cutting-edge semi-supervised methods *UAMT* [472], *DTCNet* [255] and *Ours (Semi)* with different ratio of labeled and unlabeled images. I evaluated the segmentation performance on the *SEG* test dataset with *Dice* and the *vCDR* estimation performance on the *UKBB* test dataset with

Table 4.6: Quantitative comparisons between the Ground Truth *vCDR* values (*GT vCDR*), *Ours (semi)*, *Ours (Semi-100 %)* and other cutting-edge semi-supervised methods for the glaucoma classification performance on ORIGA [490], RIM-ONE [111], and Refuge [307] test dataset. The performance is reported as *Precision (%)*, *Specificity (%)*, *Sensitivity (%)*, *AUROC (%)*. 95 % confidence intervals are presented in the brackets.

Methods	***ORIGA*** [490]				***RIM-ONE*** [111]				***Refuge*** [307]			
	Precision(%)	*Specificity(%)*	*Sensitivity(%)*	*AUROC(%)*	*Precision(%)*	*Specificity(%)*	*Sensitivity(%)*	*AUROC(%)*	*Precision(%)*	*Specificity(%)*	*Sensitivity(%)*	*AUROC(%)*
MT [382]	27.8 (21.1, 34.9)	25.5 (18.6, 32.9)	**95.7** (88.9, 100.0)	76.3 (68.7, 83.2)	28.0 (11.5, 46.7)	55.0 (39.5, 70.0)	**87.5** (64.0, 100.0)	88.8 (73.4, 99.3)	38.2 (22.6, 55.6)	81.3 (73.8, 88.1)	**86.7** (66.7,100.0)	95.0 (88.7, 99.4)
UAMT [472]	38.9 (29.4, 18.9)	62.1 (54.4, 69.9)	80.4 (68.1, 91.7)	76.7 (69.0, 83.6)	50.0 (16.7, 83.3)	87.5 (76.3, 97.4)	62.5 (25.0, 100.0)	83.8 (66.7, 96.8)	84.6 (62.5, 100.0)	98.2 (95.4, 100.0)	73.3 (50.0, 93.8)	95.4 (88.6, 99.6)
URPC [256]	44.7 (34.1, 55.4)	69.3 (62.0, 76.6)	82.6 (70.8, 92.7)	80.2 (72.9, 87.1)	41.7 (12.5, 71.4)	82.5 (70.0, 93.9)	62.5 (25.0, 100.0)	84.1 (65.3, 97.6)	90.9 (69.2, 100.0)	**99.1** (97.2, 100.0)	66.7 (40.0, 90.0)	94.7 (85.6, 99.6)
DTCNet [255]	37.4 (27.7, 47.1)	59.5 (51.6, 67.3)	80.4 (68.2, 91.1)	76.2 (67.7, 84.1)	44.4 (11.1, 80.0)	87.5 (76.5, 97.4)	50.0 (14.3, 85.7)	86.6 (72.5, 97.2)	83.3 (58.3, 100.0)	98.2 (95.4, 100.0)	66.7 (40.0, 90.0)	93.7 (83.2, 99.6)
UDCNet [202]	45.3 (34.3, 57.0)	73.2 (65.6, 80.0)	73.9 (60.0, 86.5)	80.2 (72.8, 87.1)	50.0 (14.3, 87.5)	90.0 (80.4,85.7)	50.0 (14.3, 85.7)	87.8 (73.9, 98.4)	83.3 (58.5, 100.0)	98.2 (95.7, 100.0)	60.0 (35.0, 85.7)	93.6 (82.7, 99.6)
SASSNet [211]	38.5 (28.6, 48.7)	63.4 (55.7, 71.2)	76.1 (63.5, 88.4)	77.7 (70.0, 84.9)	40.0 (10.0, 72.7)	85.0 (73.7, 95.0)	50.0 (14.3, 85.7)	85.0 (69.8, 97.3)	76.9 (50.0, 100.0)	97.3 (93.8, 100.0)	66.7 (40.0, 90.0)	94.8 (87.1, 99.3)
GT vCDR	45.5 (35.6, 55.8)	68.6 (60.9, 76.0)	87.0 (76.7, 96.0)	82.9 (76.1, 88.9)	63.6 (33.3, 90.9)	90.0 (80.0, 97.6)	**87.5** (60.0, 100.0)	91.3 (75.4, 100.0)	81.8 (44.4, 87.5)	98.2 (95.5, 100.0)	60.0 (33.3 84.6)	90.9 (81.0, 98.3)
Ours (Semi)	45.7 (35.6, 55.7)	67.3 (59.7, 75.3)	91.3 (88.2, 98.0)	85.5 (78.9, 91.3)	42.9 (16.7, 71.4)	80.0 (66.7, 92.3)	75.0 (40.0, 100.0)	90.6 (78.6, 99.1)	**91.7** (71.4, 100.0)	**99.1** (97.2, 100.0)	73.3 (50.0, 93.8)	**95.7** (90.2, 99.3)
Ours (Semi-100%)	**54.2** (42.5, 65.4)	**78.4** (71.1, 85.1)	84.8 (72.7, 94.4)	**86.5** (80.3, 91.9)	**66.6** (33.3, 100.0)	**92.5** (84.2, 100.0)	75.0 (40.0, 100.0)	**91.6** (78.4, 99.7)	90.0 (66.7, 100.0)	**99.1** (97.2, 100.0)	60.0 (35.0, 85.7)	**95.7** (90.0, 99.7)

Corr, respectively. As for the labeled images, I vary the ratio of labeled segmentation images from 5 % to 100 % (out of 1315 *SEG* training data) while fixing the number of unlabeled images to be 38421 (100 % *UKBB* training data). The performance are shown in the top of Fig. 4.5 for the averaged *OD* & *OC* segmentation performance and *vCDR* estimation, respectively. It shows *Ours (Semi)* achieves consistent superior performance over the *UAMT* [472], *DTCNet* [255] on both tasks under different labeled data utilizations. Primarily when less labeled data is used, *Ours (Semi)* suppressed the other two methods by a large margin. On the other hand, for unlabeled images, I varied the ratio of unlabeled segmentation images from 5 % to 100 % (out of 38421 *UKBB* training data) while fixing the number of labeled images to be 73 (5 % *SEG* training data). The performance are shown in the bottom of Fig. 4.5 for the averaged *OD* & *OC* segmentation performance and *vCDR* estimation, respectively. It shows *Ours (Semi)* achieved consistent superior performance over the *UAMT* [472], *DTCNet* [255] on both tasks under different unlabeled data utilizations, which indicated that my method effectively utilized the unlabeled data. When more unlabeled data is used, *Ours (Semi)* significantly outperformed the other two

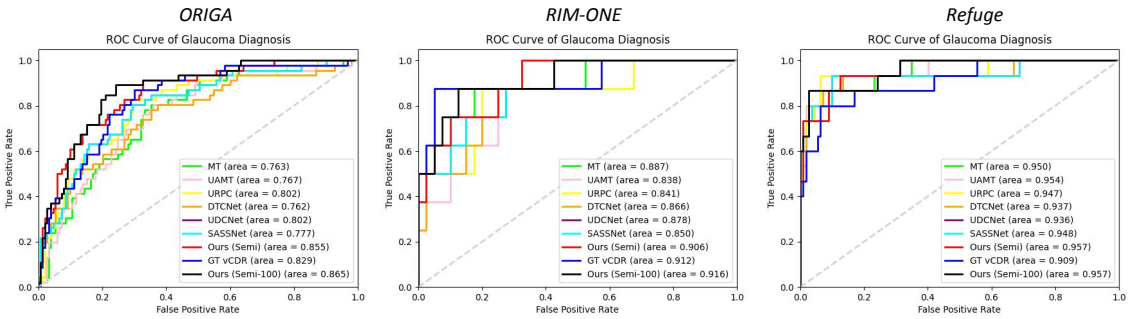


Figure 4.6: ROC curves showing the glaucoma classification performance using the Ground Truth *vCDR* values (*GT vCDR*), *Ours (Semi)*, *Ours (Semi-100 %)* and other cutting-edge semi-supervised methods on ORIGA [490], RIM-ONE [111], and Refuge [307] test dataset, respectively.

methods by a large margin.

Table 4.7: Quantitative segmentation results of polyps on the test dataset. The performance is reported as *Dice* (%) and *BIoU* (%). 95% confidence intervals are presented in brackets, respectively.

Methods	Fully-Supervised				Semi-Supervised						
	ACSNet [486]	*BI-GCN* [280]	*PraNet* [100]	*GRBNet* [286]	*MT* [382]	*UAMT* [472]	*URPC* [256]	*DTCNet* [255]	*UDCNet* [202]	*SASSNet* [211]	***Ours (Semi)***
Dice (%)	70.1 (67.8, 72.3)	73.2 (70.7, 75.8)	74.0 (72.6, 75.7)	75.7 (73.1, 77.6)	69.6 (67.2, 72.0)	70.9 (68.4, 73.3)	72.6 (70.1, 74.9)	73.1 (70.6, 75.5)	71.5 (69.0, 74.1)	71.6 (69.0, 74.0)	**74.7** (72.1, 77.0)
BIoU (%)	65.2 (62.5, 67.7)	67.5 (65.9, 70.7)	66.0 (63.3, 68.9)	69.3 (67.9, 70.5)	64.3 (61.0, 67.9)	65.4 (61.7, 67.7)	66.8 (62.3, 69.7)	66.8 (63.4, 70.0)	65.9 (62.1, 69.8)	65.4 (62.0, 68.7)	**68.7** (64.2, 71.1)

4.4.2 Glaucoma Diagnosis

In order to understand the relevance of the glaucoma diagnosis and the *vCDR* value, I conducted a classification evaluation based on the given glaucoma and healthy participant labels. Among the datasets used in this work, the glaucoma classification labels are available in RIM-ONE [111], Refuge [307], and ORIGA [490] datasets. Their corresponding test datasets with glaucoma and healthy participant labels are used in this section. In detail, there were 40 healthy participants and 8 glaucoma patients in the RIM-ONE test dataset; 112 healthy participants and 15 glaucoma patients in the Refuge test dataset; 153

healthy participants and 46 glaucoma patients in the ORIGA test dataset. I compared the aforementioned semi-supervised methods (in TABLE. 4.1), to evaluate the *vCDR* assessment performance in glaucoma diagnosis. *Precision*, *Specificity*, *Sensitivity* and Area Under the Receiver Operating Characteristic (*AUROC*) were used as the classification metrics. Specifically, a *vCDR* value larger than 0.6 is considered at risk for glaucoma, because the optic nerve damage from increased eye pressure reflected by an increase in the *vCDR* value [5, 8, 157]. TABLE. 5.3 shows the quantitative comparison between ours and previous cutting-edge semi-supervised methods on the three test datasets, respectively. *GT vCDR* represents the glaucoma diagnosis performance using the ground truth *vCDR* values of three test datasets. Specifically, *Ours (Semi-100%)* achieved consistently better *Precision* and *Specificity* than *GT vCDR* and other compared semi-supervised methods on the three test datasets. Fig. 4.6 demonstrates the *ROC Curve* comparison, where *Ours (Semi-100%)* obtains 86.5 %, 91.6% and 95.7% *AUROC* scores on ORIGA, RIM-ONE and Refuge test datasets respectively, which is consistently better than *GT vCDR*. Paired t-test results on *AUROC* of glaucoma diagnosis between *Ours (Semi)* and other semi-supervised methods were conducted using bootstrapping [335] and are shown in TABLE. 4.2, which indicates that my method achieved statistically significant improvements over other semi-supervised methods. Notably, paired T-test between *Ours (Semi-100%)* and *GT vCDR* also suggests that my method outperforms the *vCDR* ground truth with a statistically significant difference in performance ($P < 0.05$). The potential reason for not so perfect *GT vCDR* diagnosis performance and my better *AUROC* performance could be twofold. Firstly, glaucoma patients usually have a higher *vCDR* compared to normal people; however, there is a significant overlap in *vCDR* between healthy individuals and glaucoma patients [140]. Thus, only relying on *vCDR* value cannot guarantee an accurate glaucoma diagnosis. Instead, it can be used as a strong clinical indicator for suspected

disc in clinical practice [8,157]. Secondly, some of the *OD & OC* ground truth annotations in the aforementioned datasets may not be accurate, thus leading to an inaccurate *vCDR* value, which has also been noted previously [286].

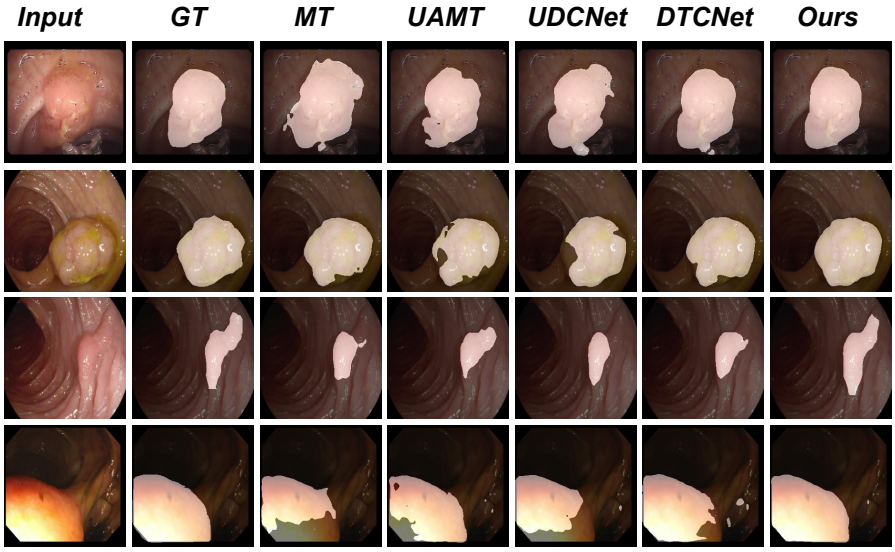


Figure 4.7: Qualitative results of colonoscopy polyps segmentation in the polyps segmentation test dataset. I compare my model with *MT* [382], *UAMT* [472], *UDCNet* [202] and *DTCNet* [255]. my method can produce more accurate segmentation results when compared with the ground truth (*GT*).

4.4.3 Generalizability of Dual Consistency regularization

In order to verify the generalizability of my proposed dual consistency regularization mechanism in semi-supervised learning, I conducted external experiments on a large-scale colonoscopy polyps segmentation benchmark [100] that has been validated by previous methods [280,286,486]. The polyp shapes in the dataset are irregular and complex, which compose a more challenging task than the *OD & OC* segmentation. The dataset contains 2,247 colonoscopy images from five datasets (ETIS [364], CVC-ClinicDB [28], CVC-ColonDB [379], EndoScene-CVC300 [400], and Kvasir [163]). All the images were resized

to 256 × 256 pixels. As suggested by [100] in fully-supervised data setting, *i.e.*, 1,450 images from Kvasir [163] and CVC-ClinicDB [28] were selected as the training datasets and the other 635 images from ETIS [364], CVC-ColonDB [379], EndoScene-CVC300 [400] are pooled together for independent testing (**unseen data**). By doing this, the training and test data are from different data sources so as to evaluate the model's generalization capability. Note that 10 % of training datasets were randomly selected as internal validation. For the semi-supervised data setting in this section, I used 50 % of the training data as the labeled data and the rest 50 % training data was used as the unlabeled data. As for the framework structure, the differentiable *vCDR* estimation module was specially designed for the *OD & OC* segmentation tasks with prior knowledge of ellipse shape objects. Thus, I removed it and remained the rest of the structure (*Ours (Semi)*) as my framework in this section. The quantitative results are shown in TABLE. 4.7, where *Ours (Semi)* achieved 74.7 % *Dice* with only 50 % labeled training data, which is comparable to the previous fully-supervised cutting-edge methods *PraNet* [100] and *GRBNet* [286]. On the other hand, my model outperformed other state-of-the-art semi-supervised methods *UAMT* [472], *URPC* [256], and *UDCNet* [202] by 5.4 %, 2.9 %, and 4.5 % in *Dice*, respectively. Paired T-test on *Dice* of segmentation between *Ours (Semi)* and other semi-supervised methods suggests a statistically significant difference in performance ($P < 0.05$). I have shown the qualitative results comparison in the Fig. 4.7, where *Ours* could generate a more accurate polyps segmentation performance compared to other semi-supervised methods. This demonstrated that my proposed dual consistency regularization mechanisms could generalise to more complex objects with irregular shapes.

4.4.4 Limitations

Regarding to the *vCDR* estimation performance in *UKBB* test dataset, *Ours (Semi)* and *Ours (Semi-100 %)* could gain 0.463 and 0.558 *Corr*, respectively. Although my method outperformed the compared cutting-edge semi-supervised methods in TABLE. 4.1, the performance is moderate if applied to real-world clinical applications. The main reason for moderate *Corr* performance could be threefold. Firstly, the limited number of labeled segmentation masks for training would undoubtedly affect the model's performance. my model could achieve better *Corr* if given more labeled data. For example, in TABLE. 1, *Ours (Semi-100 %)* outperforms *Ours (Semi)* by 20.5 % of *Corr* on *UKBB* test dataset. Secondly, the underlying low-quality input images also lead to limited performance. I considered *vCDR* estimation to be 'failed' if it fell outside 95 % confidence interval of the Bland-Altman plot in Fig. 4.4 (B). According to these criteria, I showed some of the 'failed' predictions in Fig. 4.8. It illustrates that my model could not accurately segment the *OC* and *OD* if the image quality was relatively low, such as incomplete *OD & OC* region, blurred area, extremely low-contrast, *etc.*. Thirdly, an extremely unbalanced data distribution could contribute to a moderate *Corr* performance. As the *vCDR* distribution and Bland-Altman plot shown above, the majority of *vCDR* falls between 0.3 to 0.7, where the bias mainly occurs. The glaucoma diagnosis evaluation presented in Section 4.4.2 further demonstrates that my method could achieve satisfying diagnosis performance, even when compared to the *vCDR* ground truth.

On the other hand, the designed dual consistency regularization mechanism can be widely applied to other semi-supervised medical image segmentation tasks such as ultrasound fetal head segmentation, *etc.* However, it may not work for highly complex objects, such as curvilinear structures like blood vessels [125], whose region and boundary areas can be challenging to distinguish due to their composite topology and tortuosity. Thus,

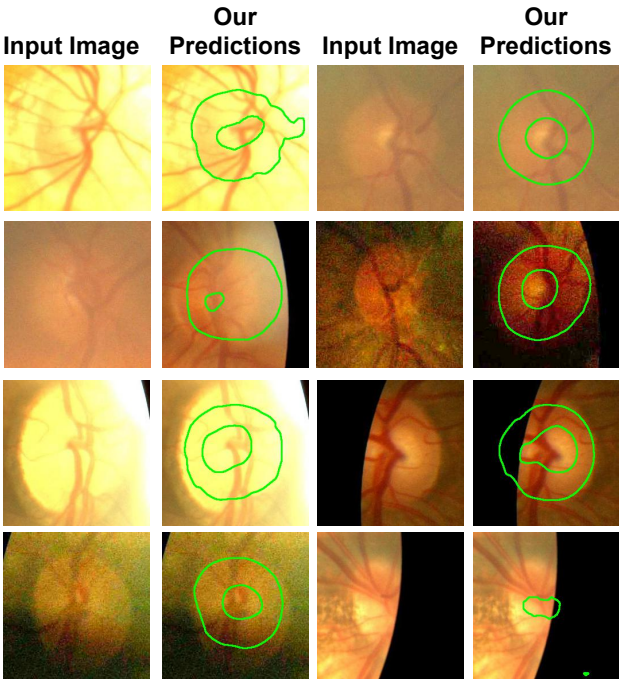


Figure 4.8: Examples of the input image and my model's predictions (*Ours (Semi)*) in some challenging cases. The proposed model failed to segment the *OD & OC* if the image quality is considerably poor, such as incomplete *OD & OC* region, blurred area, extremely low-contrast, *etc.*.

an inevitable perturbation will be introduced in the marginal and regional consistency regularization, thus impacting the semi-supervised segmentation performance.

4.4.5 Conclusion

I have proposed a novel graph-based weakly/semi-supervised segmentation framework. The geometric associations between the pixel-wise probability map features, modified signed distance function representations and object boundary characteristics are exploited in the proposed dual graph model, semi-supervised consistency regularizations, and weakly-supervised guidance. Remarkably, the proposed differential *vCDR* estimation module boosts the proposed model with a significant improvement in glaucoma assessment. Apart

from the performance, It has facilitated my model to leverage an extensive data set with no segmentation but only *vCDR* labels. Such data and labels commonly exist in real-world clinical circumstances (*UK-Biobank*); however, they are usually understudied. my experiments have demonstrated that the proposed model can effectively leverage semantic region features and spatial boundary features for segmentation of optic disc & optic cup and *vCDR* estimation for glaucoma assessment from retinal images. I believe my proposed method can be easily extended to explore geometric associations between more feature representations, such as regions, surfaces, boundaries, and landmarks in different medical image segmentation tasks.

Chapter 5

Researching Cross-Granularity Information Fusion with Implicit Graph Representations

I research the implicit graph representation learning with cross-granularity in this section, especially on the task of COVID-19 diagnosis. In detail, coronavirus disease (COVID-19) has caused a worldwide pandemic, putting millions of people's health and lives in jeopardy. Detecting infected patients early on chest computed tomography (CT) is critical in combating COVID-19. Harnessing uncertainty-aware consensus-assisted multiple instance learning (*UC-MIL*), we propose to diagnose COVID-19 using a new bilateral adaptive graph-based (*BA-GCN*) model that can use both 2D and 3D discriminative information in 3D CT volumes with arbitrary number of slices. Given the importance of lung segmentation for this task, we have created the largest manual annotation dataset so far with 7,768 slices from COVID-19 patients, and have used it to train a 2D segmentation model to segment the lungs from individual slices and mask the lungs as the regions of interest

for the subsequent analyses. We then used the *UC-MIL* model to estimate the uncertainty of each prediction and the consensus between multiple predictions on each CT slice to automatically select a fixed number of CT slices with reliable predictions for the subsequent model reasoning. Finally, we adaptively constructed a *BA-GCN* with vertices from different granularity levels (2D and 3D) to aggregate multi-level features for the final diagnosis with the benefits of the graph convolution network's superiority to tackle cross-granularity relationships. Experimental results on three largest COVID-19 CT datasets demonstrated that our model can produce reliable and accurate COVID-19 predictions using CT volumes with any number of slices, which outperforms existing approaches in terms of learning and generalisation ability. To promote reproducible research, we have made the datasets, including the manual annotations and cleaned CT dataset, as well as the implementation code, available at https://doi.org/10.5281/zenodo.6361963.

5.1 Introduction

Coronavirus disease (*COVID-19*) is a highly contagious respiratory infection caused by the new coronavirus SARS-CoV2. The most frequent symptoms of infection in the majority of infected people are fever, dry cough, and malaise ([414]). Some of these patients quickly deteriorate, developing acute respiratory distress syndrome, septic shock, multiple organ failure, and even death, among other complications ([62, 152, 209]). Nearly 600 million people have been infected worldwide so far, and over 6 million lost their lives, *COVID-19* still spreads across the world. Timely and accurate *COVID-19* diagnosis is critical for estimating the need for intensive care unit admission, oxygen therapy, prompt treatment, and so on. Despite the large number of deep learning models that have been proposed so far for the diagnosis of COVID-19 using computed tomography (CT) and *X-ray*, none of them is clinically usable due to methodological flaws and/or underlying biases [334]. There

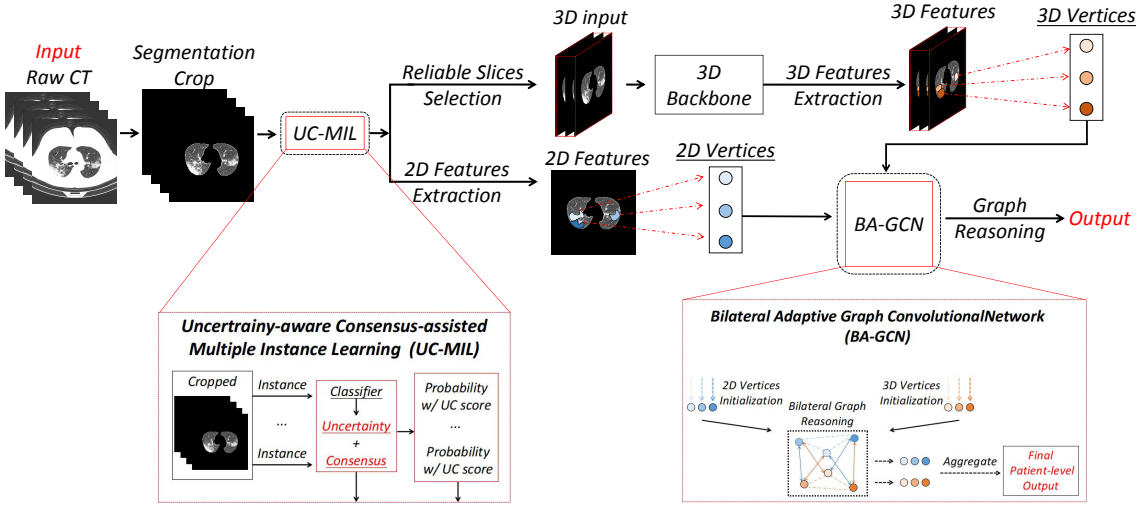


Figure 5.1: Overview of the proposed diagnosis framework. Our framework first segments and crops automatically the lung regions from the input raw 3D CT volume. Then, I automatically select trustworthy slices and the corresponding 2D features via the proposed *UC-MIL*. Afterwards, a graph-based reasoning model *BA-GCN* is proposed to aggregate and fuse the information (vertices) at 2D and 3D levels simultaneously, which contributes to the final diagnosis.

is an unmet need of accurate and robust diagnosis models for COVID-19.

Given existing *COVID-19* related datasets, such as computed tomography (CT), *X-ray*, etc., previous deep learning based diagnosis methods ([21, 122, 135, 145, 207, 430]) focus on the identification of three classes: novel coronavirus pneumonia (*NCP*), normal controls (*Normal*), and common pneumonia (*CP*) at either 2-dimensional (2D) or 3-dimensional (3D) level depending on the types of data they have used. Specifically, CT plays an important role in diagnosing and quantifying *COVID-19* and other pneumonia ([54, 145, 428, 451, 454, 460, 465, 467, 511]). The appearances on CT of infective pneumonia can give clues to its aetiology, as certain consolidation patterns are associated with specific pneumonia. Fig. 5.2 demonstrates axial CT slices comparison between various patterns of pneumonia.

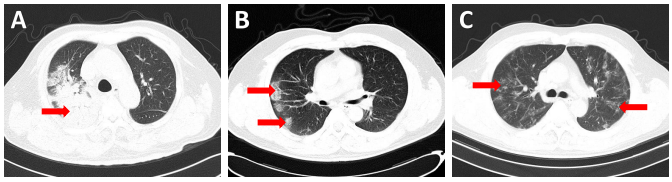


Figure 5.2: Axial CT slices demonstrate various patterns (red arrows emphasised) of pneumonia. *A*: consolidation in the posterior right upper lobe and superior right lower lobe showing typical air bronchograms and a segmental/lobar distribution in an individual with bacterial pneumonia. *B*: multifocal patches of airspace change in the posterior right upper lobe in an individual with viral pneumonia. *C*: bilateral multifocal ground glass changes in the upper lobes with some smaller reticulonodular opacities, in an individual with COVID-19 pneumonia.

The *CP* group consists of different disease types, normally including community-acquired bacterial pneumonia and viral pneumonia. In detail, community-acquired bacterial pneumonia is described as showing focal segmental or lobar opacities, but may also show ground glass attenuation or centrilobular nodules ([402]). Viral pneumonia is often described as multifocal, patchy or ground glass consolidation with influenza specifically demonstrating bilateral reticulonodular opacities ([179, 187]). COVID-19 is associated with ground glass opacities (*GGO*) and areas of consolidation that are often bilateral and peripheral ([136, 206, 358]). However, given the overlap in radiological appearances between etiological agents, with a few exceptions no reliable diagnosis of bacterial or viral origin can be made from CT ([328]), and attempts to differentiate definitively between COVID-19 and other viral pneumonia by imaging have met previously with similarly limited success ([20,180]). Additionally, the underlying correlations among CT slices are essential in *NCP* diagnosis or infection detection tasks ([124]) but have not been considered enough in existing methods. Thus, we specially design and evaluate the proposed model on *COVID-19* CT dataset in this work. The proposed framework is also readily applied to other medical applications where 3D data such as CT or MRI are used. Fig.5.3 shows an overview of

our proposed methods, where we propose a novel diagnosis framework in an attempt to address four critical difficulties that were rarely discussed or unsolved by earlier CT-based *COVID-19* approaches. The four critical issues are discussed and elaborated as follows.

Firstly, lung segmentation is an essential step prior to performing the COVID-19 classification task, however, it has received little attention in previous methods. Due to a lack of ground truth masks, previous methods ([122, 446]) segmented the lungs with pre-trained models on non-COVID datasets, while others ([416, 430]) adopted un-/ weakly-supervised schemes. However, due to the noticeable domain gap and complex appearances of CT images specific to COVID-19 (*e.g.* severe cases with massive *GGO*), the major issue is poor segmentation performance, which will compromise the subsequent *NCP* classification task. As a result, these methods need to manually clean a large number of wrong segmentations, which increase the labour cost and inconvenience for use. Here we manually annotated 7,768 slices from public COVID-19 datasets and trained a segmentation model under a fully-supervised learning mechanism. Our segmentation model can achieve more accurate results than pre-trained models or previous un-/ weakly-supervised methods; please refer to Fig.5.7 for the qualitative comparison between our model's and others' segmentations. We also prove that, without the lung segmentation, the subsequent diagnosis model may only learn a specific format pattern of different classes rather than the actual radiographic diagnosis characteristics (*i.e.* *GGO* for *NCP*). This may be due to specific CT scanner models, protocol standards, data sources of different classes, *etc.*. The potential dataset issues related to the lung segmentation process are further discussed in Section 5.7.1.

Secondly, selecting a fixed number of slices from each CT volume is often compulsory as the size of the inputs have to be the same for specific models ([103, 145, 218, 428]). Manual selection of CT slices is labour-intensive and time-consuming, which is incompatible with the goal of using *AI* models. Automatic selection following pre-defined slice sampling

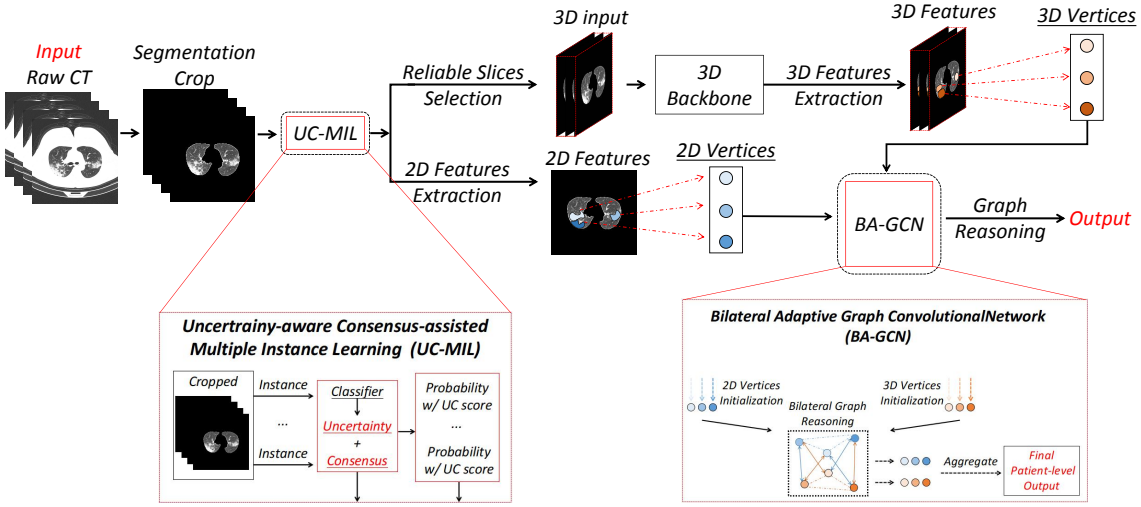


Figure 5.3: Overview of the proposed diagnosis framework. Our framework first segments and crops automatically the lung regions from the input raw 3D CT volume. Then, we automatically select trustworthy slices and the corresponding 2D features via the proposed *UC-MIL*. Afterwards, a graph-based reasoning model *BA-GCN* is proposed to aggregate and fuse the information (vertices) at 2D and 3D levels simultaneously, which contributes to the final diagnosis.

rules, on the other hand, may result in a hand-crafted optimisation process. Additionally, possibly infected slices being missed may construct a noisy dataset with intrinsic uncertainty. In this work, we propose automatically selecting reliable CT slices according to the model's probability prediction on 2D slices. A specially designed Uncertainty-aware Consensus-assisted Multiple Instance Learning *UC-MIL* model is proposed to achieve such a goal. Our *UC-MIL* can extract 2D level features for each CT slice and automatically select trustworthy slices.

Thirdly, several methods ([47, 265, 350]) have attempted to quantify the uncertainty in the *COVID-19* classification task but rarely exploited it during the training process. In other words, they only treated uncertainty as a quantification tool after the model had been trained, which overlooked the potential benefits of uncertainty during the model training

process. In general, there are two types of uncertainty [174]: epistemic uncertainty, which corresponds to the uncertainty in the model parameters and can be addressed when sufficient data is available; and aleatoric uncertainty, which corresponds to the inevitable noisy perturbations presented in the data. Publicly available CT datasets (*e.g.* [481]) contain inevitable inherent uncertainties and constraints ([145]), such as multiple domains data sources, duplicated or noisy slices, damaged data, disordered and incomplete slices, *etc.*. Alleviating the aleatoric uncertainty and exploiting it during the supervision is significant for the *COVID-19* classification tasks. In this work, we propose a *UC-MIL* to extract 2D level features and select reliable CT slices. Specifically, an uncertainty and consensus estimation module is proposed to assist the supervision process of the multiple instance learning (*MIL*) models. The underlying motivations are threefold: (1) As discussed before, the inherent uncertainty in the CT dataset may perturb the model learning process. (2) In some *NCP* cases, there might be only few slices with COVID-19 features. Under the assumptions of class-imbalanced slices distribution in a CT volume, classic *MIL* might result in the classification decision boundary closer to the uncertain (rare) slices ([202]). (3) Owing to the weakly supervised learning nature of *MIL*, the model is prone to overfitting to noisy and uncertain slices ([176]), as all the slices from a COVID-19 positive CT volume will have the same positive labels. Nonetheless, because many slices may still look normal, this label assignment may mistakenly introduce label noise and uncertainty.

Fourthly, previous COVID-19 related deep learning methods only rely on the extracted features from either 2D or 3D level, for example, 2D CNN models on 2D *X-ray* images ([290, 371, 501]) or 3D CNN models for CT volumes ([145, 428, 511]). Differently, we propose to aggregate and reason features from 2D and 3D levels concurrently during the model learning process. Specifically, we adopt the pre-trained 2D CNN model of the proposed *UC-MIL* as the initialised 2D level feature extractor. We also adopt a 3D CNN

as the backbone network to extract the 3D level features. Please refer to Section 5.3.3 for further details. With the 2D and 3D features extracted from the input CT slices, we propose a *BA-GCN* to aggregate the 2D and 3D information. Previous graph-based methods ([129, 251, 285]) have proven the superiority of graph-based models on tackling cross-granularity relationships. In this work, we regard 2D and 3D features as the bilateral vertices in a graph. A Graph Convolution Network (*GCN*) based model is proposed to aggregate information and exchange messages between cross-granularity vertices (2D and 3D). Note that the graph structure and edge relationships between vertices are adaptively learnt during the reasoning process, according to the 2D and 3D level features, respectively. In this way, the proposed *BA-GCN* can adaptively fuse and reason the bilateral relationships between 2D and 3D vertices. Specifically, in this work, the message exchange and information aggregation within 2D/3D vertices can be considered as 'inner-granularity', and between 2D/3D vertices can be considered as 'cross-granularity'. Our experiments prove that such an adaptively learnt graph can better tackle the cross-granularity relationships and achieves superior classification performance than previous *GCN* reasoning based methods. Please refer to Section 5.6.2 for more details.

In summary, this work makes the following contributions:

- This work proves that lung segmentation is an essential and necessary pre-processing step for the COVID-19 classification task on the public CT datasets. We establish the largest lung region mask dataset, with precise manual annotations of lung boundaries on the public COVID-19 CT dataset. Because of its significance we will make them publicly available to promote related research in the community.
- We propose an Uncertainty-aware Consensus-assisted Multiple Instance Learning (*UC-MIL*) model for 2D level feature extraction and automatic selection of reliable CT slices simultaneously. This avoids handcrafted data preparation and also allows

the framework to work on CT volumes with an arbitrary number of slices. It also alleviates the effects of inherent noise in public datasets on the learning and the potential uncertainty from the weakly-supervised learning mechanism of *MIL*.

- We propose a Bilateral Adaptive Graph Convolution Network (*BA-GCN*) to aggregate information and exchange messages between bilateral cross-granularity vertices (2D and 3D levels). An adaptively learned graph structure and edge relationships are built during the graph learning process to fuse and reason the relationships between 2D and 3D vertices. This helps our proposed method consider features at both levels when making inference, thus improving the classification performance.
- Extensive experiments show that our framework comprising UC-MIL and BA-GCN outperforms existing related approaches in terms of learning ability on the three largest publicly available COVID-19 CT datasets. In respect of varying dataset sources, we evaluate the generalisation ability of the proposed model on one of them as the external dataset, demonstrating its superior robustness and generalisability to the previous methods.

5.2 Related Works

In this section, we review previous *COVID-19* related works *w.r.t.* 2D and 3D level, respectively in several aspects, such as classification, infection segmentation, severity assessment, *etc.*. Additionally, as lung segmentation is an essential pre-process for the diagnosis, we review and compare previous works with such pre-process in a separate section. Apart from that, *GCN* related works in biomedical image tasks (segmentation, classification, *etc.*) have also been discussed.

5.2.1 COVID-19 Diagnosis at 2D Level

It is known that tackling the *NCP* diagnosis problem with 2D *X-ray* or 2D ultrasound images can achieve promising results in many tasks, such as severity assessment ([363,456]), infection localisation ([264,338,401,434]) and diagnosis ([18,38,102,126,190,290,304,361, 371,501]). However, compared with CT images, *X-ray* cannot indicate the significant appearance characteristics of *NCP*, such as *GGO)*, multi-focal patchy consolidation and bilateral patchy shadows ([481]). On the other hand, CT images are 3D volumes, which contain correlated spatial information among slices, essential for *NCP* diagnosis and infection localisation tasks. However, some previous methods ([101,113,149,230,395,415,433,447,504]) overlooked the 3D spatial information and developed 2D deep Learning model for the aforementioned tasks only on selected CT slices. This is mainly due to limited 3D data at the early pandemic stage, and various slice numbers of CT scans from different patients. Thus, it is difficult to develop models that can directly take CT volumes with a random number of slices as input. A potential solution adopted by previous methods ([21,122,207,317]) is to extract 2D features independently for each slice, then combining all slices' feature maps via pooling operations. Although all the slices are considered, features are still extracted independently, and correlations between slices are not utilised. Other than that, hand-crafted selection of a fixed number of slices is commonly used for most CT based *COVID-19* methods. We will discuss these methods in the following section (Section 5.2.2).

5.2.2 COVID-19 Diagnosis at 3D Level

Information at 3D level is essential for the tasks related with *COVID-19*. Most deep learning based models used 3D CT volume as the input, such as classification ([145, 381, 428]), segmentation ([451, 460, 465, 467]), disease progression ([54, 454, 511]), *etc.*.

However, all of them need a pre-process to select a fixed number of slices as the input of these models. For example, [103] selected 64 slices per CT volume as the model's input. Similarly, [145,381] utilised different slices sampling rules, including random sampling and symmetrical sampling, to select a fixed number of slices. Then a neural architecture search (*NAS*) technique was proposed to search 3D models for the *NCP* diagnosis. Along the same line, [428] used an equal interval sampling rule to select slices. A joint segmentation and classification model was proposed to indicate 3D lesion regions and *NCP* diagnosis simultaneously. [218] proposed to extract the features of *COVID-19* positive and negative samples as the pretext task, then a downstream model was developed to tackle the *NCP* classification. However, the pre-selection step was not discussed, where a fixed size of $256 \times 192 \times 56$ voxels were cropped from CT volume as the input. [308] proposed a size-balanced slice sampling mechanism to train the model in terms of repeating *NCP* data with small infections and *CP* data with large infections in each mini-batch. A pre-selection process of different patients *w.r.t.* different infection regions (small or large) need to be manually done as well. Excessive manual pre-processes made the whole framework labour-extensive and unsuitable for real-world applications.

Despite the cutting-edge performance of the models mentioned above, manual selection of a fixed number of slices is an underrated and rarely discussed issue in the task of *COVID-19* with CT. For example, manual selection of CT slices is labour-intensive and time-consuming, which violate the intention of developing *AI* models. Automatic selection under manually designed slice sampling rules may lead to a handcrafted optimisation process and cause missing potential infected slices, which results in noise data and unsubtle predictions. Furthermore, more hyper-parameters, such as the interval value, are introduced into the developed model, which will impair models generalisability. Differently, we propose a *UC-MIL* framework to work as an automated trustworthy slice selection module, according to

the estimated uncertainty and consensus score during the inference. Thus, our framework can automatically select corresponding slices, eliminating the labour-extensive pre-selection process and meeting real-world applications' needs. In other words, our model can work with a raw CT volume with an arbitrary number of slices instead of pre-selected stacked CT slices.

5.2.3 Multiple Instance Learning

Multiple Instance Learning (*MIL*) based methods ([77, 135, 144, 221]) play a significant role to address the aforementioned challenges. In detail, a whole CT volume of a patient is considered as a bag of slices (instances) that can be *COVID-19* positive or negative. Then a patient-level label is given to train the model under the weakly-supervised learning mechanism. Most of the aforementioned *MIL* based methods are inspired from ([158]), where an attention mechanism is proposed to learn a scoring system among different instances for the patient-level inference. For example, [221] proposed an attention-based *MIL* framework for the task of *NCP* severity assessment, where the instance-level attention module assigns attention scores to different instances automatically during inference. Along the same line, [77] and [135] both exploited the instance-level attention mechanisms in the task of *NCP* diagnosis. In contrast, we propose to research the uncertainty and interpretability learning of the *MIL* model. A scoring system among different slices is achieved by the uncertain value of each instance's probability predicted by our *UC-MIL* model. On the other hand, previous *MIL* based methods only rely on the extracted features of 2D instance levels. The attention module can only be seen as a weighting system among the embeddings of bags; the underlying correlations between instances are still understudied. Nevertheless, the correlations are essential in *NCP* diagnosis or infection detection tasks ([124]). In our proposed framework, the *UC-MIL* works for feature extraction and reliable

slices selection in the first stage. Moreover, we developed a 3D volume-based *BA-GCN* model in the second stage to simultaneously exploit the 2D pixel-level features and 3D slice-level correlations for a better diagnosis performance.

5.2.4 Segmentation before Classification

To mitigate the influence of non-lung region in CT slices, a standard pipeline will be to segment the lung region as a prerequisite before the *NCP* diagnosis ([416, 430, 446, 491]). For example, [446] and [122], segmented the lung regions using a pre-trained U-net on other disease (non-COVID) dataset, such as *NSCLC* ([183]) and *LUNA16* ([383]), then directly applied it to the COVID-19 CT datasets, (*e.g.* *CC-CCII* ([481]) or *MosMed* ([294])). However, *NSCLC* and *LUNA16* are CT datasets containing epithelial lung cancers, which differ noticeably from *CC-CCII* ([481]) and *MosMed* ([294]). The domain gaps between these datasets will cause poor segmentation performance of the pre-trained model, which in turn compromises the *NCP* diagnosis performance. Differently, [430] utilised an unsupervised method ([224]) to extract the connected component activation regions, which are regarded as the lung regions. However, the segmentation performance is relatively poor. It is due mainly to the distinct appearance of *NCP* CT slices from other normal ones, such as *GGO*, multi-focal patchy consolidation and patchy bilateral shadows. Thus, they had to manually clean a large number of failure cases. On the other hand, [416] followed [417], used primitive thresholding and connected-component labelling algorithms to obtain a binary lung mask that indicates the coarse lung regions. Then, a sub-image was cropped to contain lung regions covered by the convex hull of the lung masks. They treated the rough mask as the ground truth to train a model to segment the lungs, which led to inevitable noisy training data because of the inaccurate lung regions.

In summary, the aforementioned methods either adopted a pre-trained model or un-/

weakly-supervised methods to segment the lung region due to the lack of ground truth. The primary issue is poor segmentation performance, which perturbs the following *NCP* diagnosis task. Again, some methods need to clean the wrong segmentations manually, which increases the labour cost and reduces repeatability. On the contrary, we trained our segmentation module with the manually annotated lung masks under a fully-supervised learning mechanism; our segmentation model can achieve highly accurate results. We will make this manual annotation dataset publicly available. For more details about the dataset, readers are referred to Section 5.4.2.

5.2.5 Uncertainty-Assisted *COVID-19* Diagnosis

In recent years, the uncertainty and interpretability of deep learning models have been explored in several different computer vision tasks, such as scene understanding ([284,480]) and medical image analysis ([165,253,427,472]), *etc.*. Quantifying the uncertainty is crucial for *COVID-19* classification task since publicly available CT datasets contain inherent constraints, such as multiple domains of data sources, limited dataset size, *etc.*. [350] proposed a transfer learning-based framework with the help of quantified uncertainty to address the *COVID-19* diagnosis problem. They estimated the epistemic uncertainty with an ensemble learning scheme ([191]). Differently, [265] developed a deep uncertainty-aware classifier using a probabilistic generalisation of the non-parametric *KNN* approach. The proposed probabilistic neighbourhood component analysis method maps samples to latent probability distributions and then minimises a form of nearest-neighbour loss to develop classifiers. Then they estimated the uncertainty in terms of a threshold of the fraction of correctly classified examples. On the other hand, [47] researched the underlying capability of unlabeled data to improve the reliability of uncertainty. They estimated the uncertainty with the *Monte Carlo Dropout* ([194]) methods under the *MixMatch* ([29]) semi-supervised

learning scheme.

Although these aforementioned methods studied the uncertainty in the diagnosis of *COVID-19* cases, the estimated uncertainty is only used as a quantification tool at the inference stage, which overlooked the potential benefits of uncertainty during the model training. Instead, we exploit the value of uncertainty throughout the training process. Specifically, an uncertainty-aware consensus-assisted training mechanism is proposed to help the model produce more reliable predictions. Please read Section 5.3.2 for more details.

5.2.6 Graph-based Diagnosis and Reasoning

Graph-based reasoning algorithms have been studied in the recent years. Benefits from Graph Neural Network (*GNN*)'s superior ability of information propagation and message exchange, it achieved promising results in segmentation ([154,276,278,281,285,483], classification ([56,70,138,223,303,331]) and reconstruction ([64,186,439,466,498]) tasks in the field of biomedical images analysis. Graph based techniques ([18,90,227]) have been used to tackle COVID-19 related tasks as well. For example, [18] proposed a graph diffusion model that reinforces the natural relation among tiny labelled sets and vast unlabeled data in a semi-supervised learning scheme. Specifically, the graph is built on initial embeddings of the network, where each node represents an image, to produce pseudo labels, which is used for the semi-supervised *NCP* classification task. Moreover, [190] combined *CNN* and *GCN* to learn the relation-aware representation from the *NCP X-ray* images. Along the same line, [90] proposed a hypergraph model for the diagnosis of *NCP*. In detail, various types of features (*e.g.* regional features and radiographic features) are extracted from CT images for each case (CT volume). Then, the relationship among different cases was formulated by a hypergraph structure. Again, each case represented a vertex (node) in the

hypergraph. Similarly, [227] proposed a distance-aware pooling procedure along with the *GCN* to aggregate the slice level feature into the patient level gradually. The CT scan is converted to a densely connected graph, where each slice represents a vertex (node) in the graph. The problem becomes a graph classification task, and each graph represents a different patient (CT volume).

The aforementioned methods shared a similar idea: each instance (single slice or whole CT volume) was represented as a vertex in the proposed graph. A subsequent graph reasoning mechanism then propagates the vertex and edge information among instance levels. However, there are some fundamental limitations: (1) the instance level features are reasoned individually. For example, work by [227] focused only on slice level feature reasoning by a graph; the same situation happened in the works of [18] and [90] on patient levels (whole CT volume or *X-ray*) as well. This setting limits the graph-based model's capability to tackle cross-granularity or cross-feature information propagation. In other words, the *GCN* mentioned above only serves to build a long relationship between instances. However, such functionality can also be achieved by pure *CNN* based methods, according to the recent development of Non-local methods ([429]) or Transformer-based methods ([95]). (2) For *GCN* based methods ([190, 227]), they adopted *Laplacian* smoothing-based graph convolution ([182]), which provided specific benefits in the sense of global long-range information reasoning. However, they estimated the initial graph structure from a data-independent *Laplacian* matrix. Such matrix is defined by a handcrafted or randomly initialised adjacency matrix ([281]), which leads a model to learn a specific long-range context pattern ([215]). Differently, our graph-based model considered features from both 2D and 3D level to propagate the cross-granularity information. Also, as seen in previous works, the graph structure can be estimated with the similarity matrix from the input data [217]. We estimate the initial adjacency matrix in an input-dependent way.

Specifically, a reasoning mechanism is achieved by propagating information and passing messages among inner-granularity and cross-granularity vertices (2D and 3D). Additionally, the structure of our *BA-GCN* is adaptively built during the graph reasoning according to the information of 2D and 3D levels. Thus, the graph representations can be adaptively learnt in an input-dependent way instead of the pre-defined hand-craft one from the previous methods. Please read Section 5.3.3 for more details. Notably, a recent work ([500]) built the adjacency matrix based on the instance features in a bag under *MIL* paradigm, which can also be regarded as input-dependent. However, they handcrafted the adjacency matrix weights, and the major difference between *Ours* and theirs are threefold: (1) *Zhao et al.* ([500]) built a binary adjacency matrix with edge weight values of 0 or 1 to indicate whether the vertices are connected or not. However, the similarity among vertices is overlooked. Differently, *Ours* exploited the relationship among vertices' own correlation and can indicate the similarity of different vertices with normalised edge weights between 0 and 1. (2) *Zhao et al.* ([500]) introduced a hyper-parameter (γ) to determine if two vertices are connected or not, according to their Euclidean distance. Conversely, *Ours* does not introduce any hyper-parameter and only relies on the vertices' own correlations. (3) *Ours* constructs a fully-connected graph with every vertex connecting to one other, while *Zhao et al.* ([500]) did not because of the potential edge weight of 0.

5.3 Methods

Fig.5.4 shows the proposed method's pipeline. It contains three sub-tasks: (1) lung region segmentation, (2) reliable CT slices selection and *COVID-19* classification on 2D levels (*UC-MIL*), (3) *COVID-19* classification at both 2D and 3D levels (*BA-GCN*). Given an input CT volume with an arbitrary number of slices, we first segmented the lung regions for each slice, then fed the segmented CT volume into a *UC-MIL* model to learn and extract

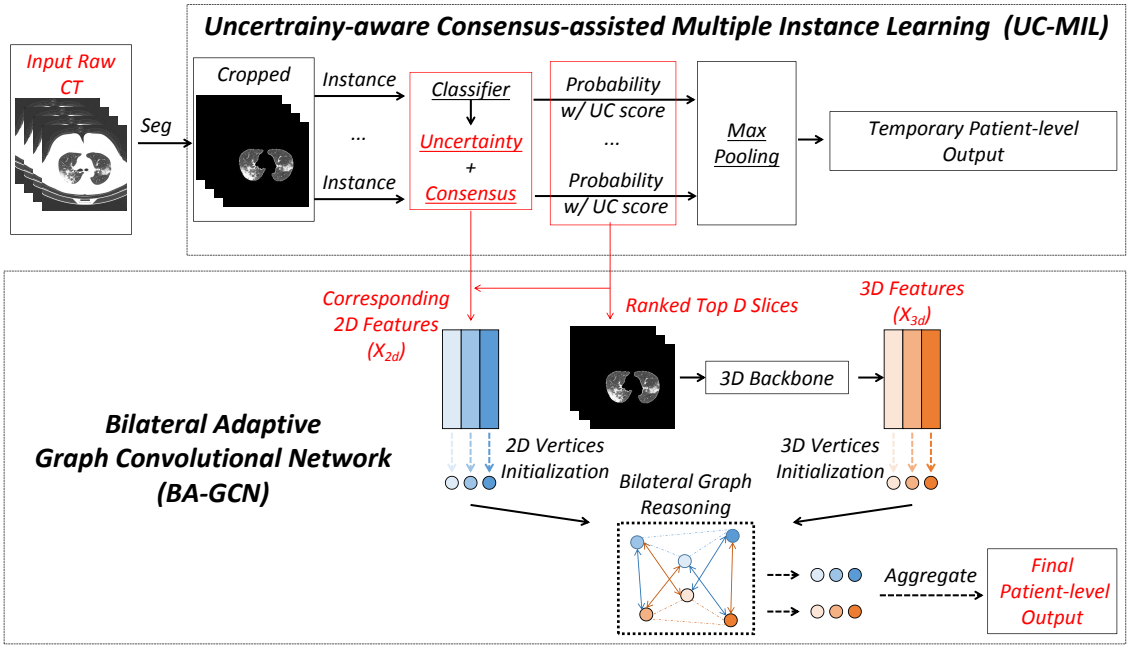


Figure 5.4: Illustration of the proposed method's pipeline. In addition to the lung segmentation and region cropping, the two stage diagnosis mechanism *w.r.t.* *UC-MIL* and *BA-GCN* is shown on the top and bottom, respectively. *Seg* represents the lung region segmentation; *UC score* denotes the estimated uncertainty and consensus scores. Notably, the non-lung regions were masked out from the raw CT data by using our lung segmentation model before input into the *UC-MIL*. The 2D/3D level of vertices are initialised by the feature maps at 2D/3D level, which are extracted from *UC-MIL* and *MF-Net* backbone, respectively.

the relevant features at 2D slice level under a weakly supervised learning mechanism. After that, we selected D slices from each CT volume according to the predicted probability of *UC-MIL* model. The D slices and the corresponding 2D features extracted from *UC-MIL* are regarded as the input for the proposed *BA-GCN*. The *BA-GCN* learns the features on the 3D volume level (D slices) and also propagate the information from 2D level features among different vertices in the bilateral graph. Notably, the hyper-parameter D is empirically set as 16 in this work. The details of each task and the developed models are elaborated as follows.

5.3.1 Lung Segmentation

Because our intention was primarily the task of *COVID-19* classification, here we only utilised classic methods, such as *UNet* ([337]), *UNet++* ([509]), and other cutting-edge methods, such as *PraNet* ([100]), *RBA-Net* ([276]), *CABNet* ([278]), *GRB-GCN* ([285]), and *BI-GConv* ([281]). We trained those models with the annotated slices at the 2D level, and applied the trained model on the rest unannotated images then cropped the lung regions. After that, the CT volume containing lungs only is ready for the following *COVID-19* diagnosis task. Please note that lung segmentation process is essential and necessary in the task of *COVID-19* classification primarily due to the dataset issue. Please refer to Section 5.7.1 for further details.

5.3.2 UC-MIL for Diagnosis on 2D Level

To develop a comprehensive *COVID-19* classification model, we built a *UC-MIL* model to learn the diagnosis features at 2D level. In the MIL paradigm ([9, 91, 268]), unlabeled instances belong to labeled bags of instances. The goal is to predict the label of a new bag or the label of each instance. We will elaborate the mechanisms of the proposed *UC-MIL* in the following subsections.

Multiple Instance Learning

We denote a patient's CT volume as a *bag* and the slices herein as *instances*, following the standard *MIL* formulation. We associate the bag label with the corresponding instances. In other words, all instances from the same bag have the same label and are considered discriminatory. Nonetheless, this assignment may inadvertently add label noise in positive bags due to the possibility of a certain number of slices being negative. Thus, exploiting the discriminative training samples is essential under this circumstance. Here, 'discriminative'

represents that the true hidden label of the instance is the same as the true label of the bag.

Let $X = \{X_1, X_2, ..., X_N\}$ as the dataset containing N bags. Each bag $X_i = \{x_{i,1}, x_{i,2}, ..., x_{i,N_i}\}$ consists of N_i instances, where $X_{i,j} = \{x_{i,j}, y_i\}$, $x_{i,j}$ is the *j-th* instance, y_i denotes its associated label in the *i-th* bag. Please note, N_i may differ due to different number of slices in different CT volumes. The label Y_i of bag X_i is given by:

$$Y_i = \begin{cases} 0, & iff \sum_i y_i = 0 \\ 1, & otherwise. \end{cases} \tag{5.1}$$

Generally, a *MIL* based prediction model contains an appropriate transformation f and a *permutation-invariant* transformation g ([158, 202, 431]). Thus, the *MIL's* prediction for bag X_i is defined as:

$$P(X_i) = g\big(f(x_{i,1}), f(x_{i,2}), ..., f(x_{i,N_i})\big). \tag{5.2}$$

With respect to the choice of f and g, there are generally two types: 1.) Instance-based approach. f is an instance classifier that assigns a score to each instance, and g is a pooling operator (*e.g.* max pooling) that fuses the instance scores to obtain a bag score. Specifically, a 2D *CNN* was trained to predict the class probability of each instance. A few instances with higher responses were selected and performed back-propagation during training. An iteration process was used with a new set of discriminative instances until convergence. 2.) Embedding-based approach. f is an instance-level feature extractor that maps each instance to an embedding; g is an aggregation operator that produces a bag embedding from the instance embeddings and outputs a bag score based on the bag embedding. The embedding-based method generates a bag score from a bag embedding supervised by the bag label. The discriminative and non-discriminative instances'

embeddings contribute differently to the overall bag prediction ([431]). However, it is typically more challenging to identify the discriminative instances that activate the classifier, compared with instance-based approaches ([244]).

Uncertainty-aware Consensus-assisted Multiple Instance Learning

All the previous *MIL* based *COVID-19* diagnosis methods ([77, 135, 221]) are embedding-based methods, which are adapted from [158]. In this work, we take another direction and propose an instance-based *UC-MIL* method. Our experimental results prove that the proposed method outperforms previous instance-based and embedding-based methods on two different evaluation settings (learning ability and generalisation ability). Additionally, we conducted extensive ablation studies to determine the backbone network of the proposed *MIL* method. More experimental details are referred to Section 5.6.2.

Previous instance-based *MIL* methods ([48,150,312,315,318,444,479]) achieved promising results on different medical image classification tasks, such as whole slide image classification, optical coherence tomography image classification, *etc.*. However, two significant challenges remain for these works. Firstly, the distribution of instances in the positive bags may be extremely imbalanced when only a tiny proportion of instances are positive, and models are prone to misclassify those positive instances as negative, especially when a simple aggregation operator, such as max-pooling, is used. This is because, under the assumptions of *MIL* and imbalanced instances in a bag, max-pooling might result in the classification decision boundary closer to the uncertain (rare) instances ([202]). Secondly, as discussed above, all the instances from the same bag have the same label and are considered discriminatory. Nonetheless, this assignment may inevitably add label noise into positive bags due to the possibility of a certain amount of slices being negative. Due to such a weakly supervised learning mechanism, the model is prone to overfitting to noisy

and uncertain instances, resulting in poor generalisability in real-world clinical practice. Additionally, instances with high uncertainty have a disproportionate presence in the classification space, making it difficult to generalise learnt limits to new test examples. ([176]).

To solve this problem, we specifically design an uncertainty estimation module and a consensus achievement module into the standard instance-based *MIL* model training pipeline, where an uncertainty-aware consensus-assisted supervision process is conducted. Firstly, to quantify the reliability of each instance's prediction, we adopt Shannon Entropy ([352]) as the metric to measure the randomness of the information ([351]), which is referred to as the uncertainty in this work. Formally, given a C-dimensional softmax predicted class score $P_{x_{i,j}}^{(C)}$ from an input instance $x_{i,j}$, the uncertainty $I_{x_{i,j}}$ is defined as:

$$I_{x_{i,j}} = -\sum_{c=1}^{C} P_{x_{i,j}}^{(C)} \odot \log P_{x_{i,j}}^{(C)}, \tag{5.3}$$

where $\odot$ is Hadamard Product; C is the number of classes. In practice, we perform T times stochastic forward passes on each instance classifier under random dropout and Gaussian noise perturbed input for each input instance. Note that T is empirically set as 8 in this work. Therefore, under such self-ensemble mechanism, we obtain a set of softmax probability vectors: $\left\{P_{x_{i,j}}^{t}\right\}_{t=1}^{T}$, then the mean predicted class score $\tilde{P}_{x_{i,j}}^{(C)}$ is given as:

$$\tilde{P}_{x_{i,j}}^{(C)} = \frac{1}{T}\sum_{t=1}^{T} P_{x_{i,j}}^{t}, \tag{5.4}$$

thus based on equation (5.3) we can obtain the uncertainty $\tilde{I}_{x_{i,j}}$ for input instance $x_{i,j}$ as :

$$\tilde{I}_{x_{i,j}} = -\sum_{c=1}^{C} \tilde{P}_{x_{i,j}}^{(C)} \odot \log \tilde{P}_{x_{i,j}}^{(C)}. \tag{5.5}$$

With the quantified uncertainty $\tilde{I}_{x_{i,j}}$ for instance $x_{i,j}$, we normalise $\tilde{I}_{x_{i,j}}$ into $[0, 1]$ then

perform element-wise broadcasting multiplication between $\tilde{I}_{x_{i,j}}$ and softmax predicted class score $P^{(C)}_{x_{i,j}}$. In this way, uncertainty-weighted probability prediction $P^{(C)}_{\tilde{I}_{x_{i,j}}}$ for each instance $x_{i,j}$ is calculated as:

$$P^{(C)}_{\tilde{I}_{x_{i,j}}} = \tilde{I}_{x_{i,j}} \otimes P^{(C)}_{x_{i,j}}, \tag{5.6}$$

where $\otimes$ denotes the element-wise broadcasting multiplication. In other words, the operator g in our *UC-MIL* will consider the reliability of each $f(x_{i,N_i})$ in equation (5.2), and only the trustworthy slides are considered for the model to learn the features.

Secondly, under a certain perturbation, network predictions for memorised features that learned from noise change significantly, while those for generalised features do not ([196]). In other words, the predictions of a generalisable instance classifier should be robust to input perturbation, and the predicted class score that changes significantly under a certain perturbation hence highly suggests a noisy instance ([202]). Thus, we quantify the consensus regarding the standard deviation over a self-ensembling models' multiple outputs, with the same input but under various perturbations. Formally, for an instance $x_{i,j}$, given a set of softmax probability vectors $\left\{P^t_{x_{i,j}}\right\}_{t=1}^{T}$ and the mean predicted class score $\tilde{P}^{(C)}_{x_{i,j}}$, the standard deviation $\hat{P}^{(C)}_{x_{i,j}}$ of the predicted class score is defined as:

$$\hat{P}^{(C)}_{x_{i,j}} = \frac{1}{T}\sqrt{\sum_{t=1}^{T}(P^t_{x_{i,j}} - \tilde{P}^{(C)}_{x_{i,j}})^2}, \tag{5.7}$$

which is regarded as the metric of consensus in this work. With such quantified consensus achievement, we exclude the uncertain instances so as to guide the model to learn from more reliable instances. More specifically, the reliable instances j_r in bag X_i are selected *iif* $\hat{P}^{(C)}_{x_{i,j}}$ is smaller than a threshold γ. Formally, for bag X_i, the trustworthy instances set Ω is given by:

$$\Omega = \{x_{i,j} | \hat{P}^{(C)}_{x_{i,j}} < \gamma\}. \tag{5.8}$$

Notably, we perform extensive experiments to tune the hyper-parameter γ value, which is empirically set as 0.02 in this work. The number of trustworthy slices in Ω ranges from 16 to 45 for all of the data used in this work. This comes with the advantage that our framework can deal with CT volumes with any arbitrary number of slices.

Combining the uncertainty and the consensus scores discussed before, the whole optimisation procedure of the proposed *UC-MIL* methods in a single bag (X_i) can be found in Algorithm (1). An iteration process was used with a new set of bags ($X_1, ..., X_N$) to update the parameter of the instance classifier until convergence.

Algorithm 1 Uncertainty-aware Consensus-assisted *MIL*

Data: A bags of N_i instances: $X_i = \{x_{i,1}, ..., x_{i,N_i}\}$

Result: Instance classifier: $f(x_{i,j})$, $j \in [1, N_i]$

initialisation;
for $j \leftarrow 1$ **to** N_i **do**
    Calculate $\tilde{P}^{(C)}_{x_{i,j}}$ with Eq.(5.4);
    Calculate $\hat{P}^{(C)}_{x_{i,j}}$ with Eq.(5.7);
end
Calculate Ω with Eq.(5.8); // N_{i_r}: size of Ω
for $j_r \leftarrow 1$ **to** N_{i_r} **do**
    // x_{i,j_r}: instances from Ω
    Calculate $\tilde{I}_{x_{i,j_r}}$ with Eq.(5.5);
    Calculate $P^{(C)}_{\tilde{I}_{x_{i,j_r}}}$ with Eq.(5.6); // $P^{(C)}_{\tilde{I}_{x_{i,j_r}}}$: $f(x_{i,j_r})$'s output
end

In this way, whether a retrieved discriminative instance is trustworthy or noisy can be differentiated by the model during the training. The learnt classifier considers the uncer-

tainty level of the instance predictions to re-adjust boundaries (*i.e.*, providing more room to uncertain samples). This improves the generalisation ability of the proposed model for either imbalanced instances or weakly supervised learning mechanisms ([176]). Furthermore, our experiments prove that with the *UC-MIL* training, our model outperforms previous instance-level *MIL* methods by a large margin in the evaluation of generalisation ability. Notably, previous instance-level *MIL* methods conduct a promising classification results in the seen data (*i.e.* the evaluation of learning ability), however, drop dramatically on unseen data (*i.e.* the evaluation of generalisation ability).

During the training, we adopted the same method used in ([48]), which selects the top instances with maximum prediction probability within a bag as the bag's prediction. Such bag-level aggregation derives directly from the standard multiple instance assumption and is generally referred to as 'max-pooling' ([48]) and is shown in Fig. 5.4. With the proposed *UC-MIL*, we obtain temporary patient-level diagnosis results in the first stage. However, the instance level features are learned individually during the whole training process. In other words, only 2D level of information is considered in *UC-MIL*. Thus we aggregate both 2D and 3D features in the subsequent *BA-GCN*, which helps to make the diagnosis more reliable and accurate.

5.3.3 Diagnosis at both 2D and 3D Levels

In this section, we demonstrate the proposed *BA-GCN w.r.t.* the *COVID-19* diagnosis at both 2D and 3D levels. As discussed in Section 5.2.2, the correlations between different CT slices are essential for the *COVID-19* diagnosis. For *bag* X_i, we select the top D instances (slices) according to the ranked order of uncertainty-aware consensus-assisted instance prediction probability ($P^{(C)}_{\tilde{I}_{x_{i,j_r}}}$) from the corresponding trustworthy set Ω. Then we stack the slices along the depth channel as the 3D input for the proposed *BA-GCN*.

In this way, we can automatically select a fixed number of reliable slices from each CT volume, which avoids the labour-intensive manual selection process or other hand-craft slice sampling strategies that are adopted by the previous methods ([103, 145, 218, 308, 428]). Additionally, the extracted slice-level features of *UC-MIL* are used as the 2D feature maps input for the proposed *BA-GCN*. Specifically, for each of the D slices classifier in *UC-MIL*, we extract the feature map before the pooling layer and add an 1×1 convolution layer to reduce the channel size to 128. Then for each CT volume, we stack all the corresponding D feature maps along the depth channel as the '2D' input for the proposed *BA-GCN*. We represent the '2D' input as X_{2D} in this work. Notably, X_{2D} has the size of $D \times 128 \times 7 \times 7$. The size format follows (D $\times$ C $\times$ H $\times$ W), where D is number of slices; C is channel size; H and W represents height and width of feature maps, respectively. There are two primary modules in the proposed *BA-GCN*: (1) Backbone Network, (2) Bilateral Adaptive Graph Reasoning Module. The details for each of them are elaborated as follows.

Backbone Network

We firstly input 3D CT volumes as the inputs into a backbone network to extract features and learn the correlations between different slices at the 3D level. Different from previous methods ([218, 308, 428]), where the 3D extensions of *ResNet* ([143]) or *Inception-Net* ([378]) are used as the backbone, we adopt Multi-Fiber Network (*MF-Net*) ([69]) due to its superior ability to extract discriminative features in recognition tasks. *MF-Net* ([69]) is a sparsely connected 3D *CNN* backbone that costs a minimal computational overhead, but brings a boosted representation capability of features. The multiple separated lightweight residual units, called fibers, can effectively reduce the number of connections within the network and enhance the model efficiency. The advantage of *MF-Net* fits in and benefits our model in this specific task. Our ablation study results also prove that the *MF-Net*

based backbone outperforms *ResNet* or *Inception-Net* variants in this work. Specifically, the 3D Multi-Fiber Units can enhance the model efficiency while effectively reducing the number of computations. In detail, we extract the feature map before the pooling layer, then add a $1 \times 1 \times 1$ convolution layer to reduce the channel size to 128, and save it for the subsequent information aggregation process in the proposed *BA-GCN*. We refer to this feature maps as X_{3D} in this work. Notably, the X_{3D} has the same size as X_{2D}, with $D \times 128 \times 7 \times 7$.

Bilateral Adaptive Graph Reasoning Module

Given the feature maps extracted from *UC-MIL* as 2D level's information (X_{2D}) and the feature maps extracted from *MF-Net* Backbone as 3D level's information (X_{3D}), we propose a bilateral adaptive graph to aggregate the features from both 2D and 3D levels. In detail, a graph reasoning module is achieved by information exchange and propagation among different granularity levels of vertices. Additionally, our graph structure and the edge relationship are adaptively learnt during the reasoning process according to the 2D and 3D level features' own information. Thus, a bilateral adaptive graph representation can be learnt in an input-dependent way, rather than predefined hand-craft ones ([18,90,227]).

Classic Graph Convolution We begin with a review of classic graph convolution. Given a graph $G = (V, E)$, the normalised *Laplacian* matrix is defined as $L = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$, where I is the identity matrix, A is the adjacency matrix, and D is a diagonal matrix representing the degree of each vertex in V, such that $D_{ii} = \sum_j A_{i,j}$. Because the graph's *Laplacian* is a symmetric and positive semi-definite matrix, L may be diagonalised using the Fourier basis $U \in \mathbb{R}^{N \times N}$, resulting in $L = U \Lambda U^T$. Thus, the Fourier space spectral graph convolution of i and j may be described as $i * j = U((U^T i) \odot (U^T j))$. The columns of U correspond to the orthogonal eigenvectors $U = [u_1, ..., u_n]$, and $\Lambda = diag([\lambda_1, ..., \lambda_n]) \in$

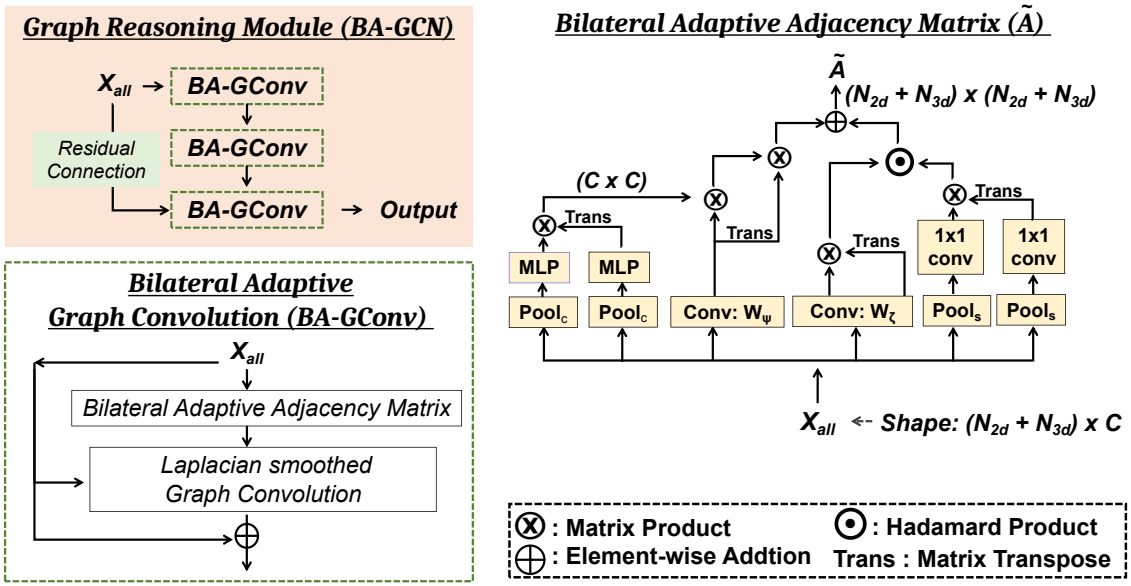


Figure 5.5: Overview of the proposed *BA-GCN*, Bilateral Adaptive Graph Convolution (*BA-GConv*) and Bilateral Adaptive Adjacency Matrix ($\tilde{A}$).

$\mathbb{R}^{N\times N}$ is a diagonal matrix with non-negative eigenvalues. Due to the fact that U is not a sparse matrix, this operation is computationally inefficient. [85] hypothesised that the convolution operation on a graph may be characterised by constructing spectral filtering with a kernel g_θ in Fourier space through a recursive Chebyshev polynomial. The filter g_θ is parameterised as a Chebyshev polynomial expansion of order K, such that $g_\theta(L) = \sum_k \theta_k T_k(\hat{L})$, where $\theta \in \mathbb{R}^K$ is a vector of Chebyshev coefficients, and $\hat{L} = 2L/\lambda_{max} - I_N$ is the rescaled *Laplacian*. $T_k \in \mathbb{R}^{N\times N}$ is the Chebyshev polynomial of order K. [182] further simplified the graph convolution to $g_\theta = \theta(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}})$, where $\hat{A} = A + I$, $\hat{D}_{ii} = \sum_j \hat{A}_{ij}$, and θ are the only remaining Chebyshev coefficients. The corresponding graph *Laplacian* adjacency matrix $\hat{A}$ is handcrafted, causing the model to learn a specific long range context pattern rather than the input-related one ([215]). Thus, we refer to the classic graph convolution ([182]) as handcrafted input-independent graph convolution.

Bilateral Adaptive Graph Convolution Given $X_{2D} \in \mathbb{R}^{N_{2d}\times C}$ and $X_{3D} \in \mathbb{R}^{N_{3d}\times C}$,

where C is the channel size; $N_{2d} = H_{2d} \times W_{2d} \times D$ and $N_{3d} = H_{3d} \times W_{3d} \times D$ are the number of spatial locations of 2D and 3D level of input features, which are referred to as the number of vertices. Note that H, W and D represent the height, width, and depth of the corresponding level of feature map, respectively. Firstly, we construct the bilateral adjacency matrix ($\tilde{A}$) in an adaptive way. The vertices of 2D and 3D (X_{2D}, X_{3D}) contribute to the adjacency matrix construction concurrently and adaptively. In detail, we stack them together and represent it as $X_{all} \in \mathbb{R}^{(N_{3d}+N_{2d})\times C}$, which is regarded as the input vertices of *BA-GConv* (shown in Equation. (5.12)). Then, we implement two matrices ($\tilde{\Lambda}^c$ and $\tilde{\Lambda}^s$) to execute channel-wise attention on the dot-product distance and to quantify spatially weighted relations between various input vertices embeddings, respectively. For example, $\tilde{\Lambda}^c(X_{all}) \in \mathbb{R}^{C\times C}$ is the matrix that contains channel-wise attention on the dot-product distance of the input vertex embeddings; $\tilde{\Lambda}^s(X_{all}) \in \mathbb{R}^{(N_{3d}+N_{2d})\times(N_{3d}+N_{2d})}$ is the spatial-wise weighting matrix, measuring the spatial relationships among different vertices.

$$\tilde{\Lambda}^c(X_{all}) = \Big(MLP\big(Pool_c(X_{all})\big)\Big)^{\mathrm{T}} \cdot \Big(MLP\big(Pool_c(X_{all})\big)\Big), \tag{5.9}$$

where $\cdot$ denotes matrix product; $Pool_c(\cdot)$ is the global max pooling for each vertex embedding; $MLP(\cdot)$ is a multi-layer perceptron with one hidden layer. On the other hand,

$$\tilde{\Lambda}^s(X_{all}) = \Big(Conv\big(Pool_s(X_{all})\big)\Big) \cdot \Big(Conv\big(Pool_s(X_{all})\big)\Big)^{\mathrm{T}}, \tag{5.10}$$

where $Pool_s(\cdot)$ represents the global max pooling for each position in the vertex embedding along the channel axis; $Conv(\cdot)$ is a 1×1 convolution layer. The data-dependent adjacency matrix $\tilde{A}$ is given by spatial and channel attention-enhanced input vertex embeddings. We

initialise the bilateral adjacency matrix $\tilde{A} \in \mathbb{R}^{(N_{3d}+N_{2d})\times(N_{3d}+N_{2d})}$ as:

$$\begin{aligned}\tilde{A} = \psi(X_{all}, W_\psi) \cdot \tilde{\Lambda}^c(X_{all}) \cdot \psi(X_{all}, W_\psi)^T + \\ \zeta(X_{all}, W_\zeta) \cdot \zeta(X_{all}, W_\zeta)^T \odot \tilde{\Lambda}^s(X_{all}),\end{aligned} \tag{5.11}$$

where $\cdot$ represents matrix product; $\odot$ denotes Hadamard product; $\psi(X_{all}, W_\psi) \in \mathbb{R}^{(N_{2d}+N_{3d})\times C}$ and $\zeta(X_{all}, W_\zeta) \in \mathbb{R}^{(N_{2d}+N_{3d})\times C}$ are both linear embeddings; W_ψ and W_ζ are learnable parameters. Fig.5.5 shows a detailed demonstration of the bilateral adjacency matrix $\tilde{A}$. Please note that the different granularity levels of relationships among vertices from 2D and 3D (X_{2D}, X_{3D}) are exploited in this bilateral graph, where the graph is adaptively built up according to the multi-granularity vertices' own correlations in a data-dependent way. With the constructed $\tilde{A}$, the normalised *Laplacian* matrix is given as $\tilde{L} = I - \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$, where I is the identity matrix; $\tilde{D}$ is a diagonal matrix that represents the degree of each vertex, such that $\tilde{D}_{ii} = \sum_j \tilde{A}_{i,j}$; notably a softmax is applied on $\tilde{A}$ for normalised adjacency weights. We calculated degree matrix $\tilde{D}$ with the same way that is used in ([215, 281]), to override the computation overhead. Given computed $\tilde{L}$, with X_{all} as the input vertex embeddings, we formulate the single-layer *BA-GConv* as :

$$Y = \sigma(\tilde{L} \cdot X_{all} \cdot W_G) + X_{all}, \tag{5.12}$$

where $W_G \in \mathbb{R}^{C\times C}$ denotes the trainable weights of the *BA-GConv*; σ is the ReLu activation function. Additionally, we include a residual connection to preserve the features of input vertices. $Y \in \mathbb{R}^{(N_{3d}+N_{2d})\times C}$ is the output vertex features. Empirically, Three layers of the proposed *BA-GConv* with residual connections build up a graph reasoning module (*BA-GCN* shown in Fig.5.5). After the *BA-GCN*, a convolution layer is added to reduce the channel size to one. Two layers of *MLP* with ReLu and Softmax as the activation

functions respectively are used to aggregate the output vertices features and predict the final patient-level diagnosis probability.

5.4 Experiments

5.4.1 Datasets

In this work, we perform experiments on three currently largest publicly available COVID-19 CT datasets: *CC-CCII* ([481]), *MosMed* ([294]) and *COVID-CTset* ([320]). All of the three datasets are used in PNG format in this work. The total number of slices per CT volume ranges from 16 to 375. They are utilised to evaluate the **learning ability** and **generalisation ability** of our proposed model, respectively. The details of these three datasets *w.r.t.* two evaluation settings are shown in Table.5.1. The *CC-CCII* ([481]) dataset contains three classes of *NCP*, *CP* and *Normal* and the other two datasets only contain two classes of *NCP* and *Normal*. We evaluate the learning ability of our proposed model on *CC-CCII* ([481]) dataset. On the other hand, we evaluate the generalisation ability of our proposed model . Firstly, in order to eliminate the effect of imbalanced data class distribution, we combine the *Normal* class's data of *CC-CCII* ([481]) with all of *MosMed* ([294]) dataset as *Train & Val* dataset. Then, the *COVID-CTset* ([320]) dataset is treated as the *External Test* dataset. We have shown this data setting in the middle of Table. 5.1. We elaborate the details of each datasets below.

- ***CC-CCII*** ([481]). The original *CC-CCII* dataset contains a total of 617,775 slices of 6,752 CT volume from 4,154 patients. However, it has several problems, such as corrupted data, duplicated and noisy slices, incomplete slices, non-unified data type, *etc.*. Please see Fig. 5.6 for details. Considerable effort has been made to build a clean dataset for training and evaluation. We have manually checked the whole

Table 5.1: Descriptions of the three COVID-19 CT datasets. Cleaned *CC-CCII* ([481]), *MosMed ([294])* and *COVID-CTset* ([320]) are three currently largest publicly available *COVID-19* CT datasets. # Patient and # Slices represent the number of patient and slices, respectively. *Train & Val* represent the subset that contains train and validation datasets. Note that we randomly select 10 % of *Train & Val* as the validation datasets.

Datasets	Classes	# Patients		# Slices	
		Train & Val	*Test*	*Train & Val*	*Test*
CC-CCII	*NCP*	414	133	24,255	10,330
	CP	773	186	59,080	12,509
	Normal	675	174	50,874	15,266
	Total	1,862	493	134,209	38,105
MosMed* + *CC-CCII* + *COVID-CTset	*NCP*	856	95	28,188	15,589
	Normal	929	95	59,439	12,718
	Total	1,785	190	97,627	28,307

dataset and removed the noisy data (damaged, duplicated and non-unified). Note that we only use complete scans with volume scan per patient to avoid information leakage during training and evaluation. After addressing the above problems, we build a clean *CC-CCII* dataset, which consists of 172,314 slices of 2,355 scans from 2,355 patients (shown in Table.5.1). Apart from the issues above, *CC-CCII* provided pre-segmented CT slices only but without original CT slices for part of the dataset. For example, in the clean *CC-CCII*, 59,256 slices of 740 volume from 740 patients are pre-segmented, and the rest 113,058 slices of 1,615 scans from 1,615 patients are not. Our experimental results proved that lung segmentation pre-process is necessary for the task of *COVID-19* classification, especially for models trained on *CC-CCII* ([481]) datasets. The details of the potential dataset issue related to the lung segmentation pre-process are discussed in Section 5.7.1. Besides, some qualitative visualisation results, such as GradCAMs ([348]), are shown in Fig.5.11 to prove the importance of lung segmentation pre-process in this task. To address the non-segmentation problem, we segmented the lungs of the non-segmented slices

with our trained model. Compared with the pre-segmented lung slices of *CC-CCII* ([481]), our model can segment more accurate lung regions. Qualitative results and comparisons are shown in Fig.5.7. As illustrated, our segmentation can generate a more smooth lung boundary and conduct fewer false positive predictions.

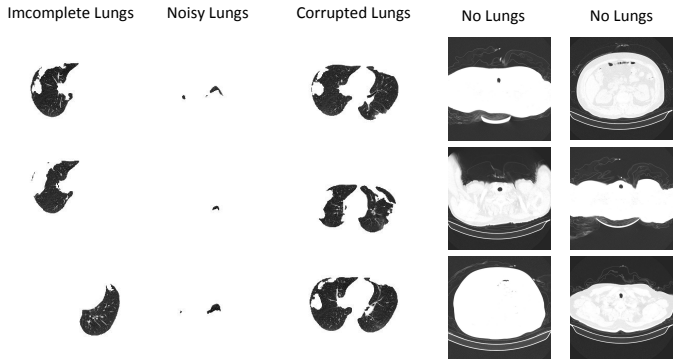


Figure 5.6: Examples of problematic slices from the original *CC-CCII* ([481]) dataset. Those noisy data will inevitably introduce perturbations into both the lung segmentation task and the COVID-19 diagnosis task.

- ***MosMed*** ([294]). *MosMed* dataset was collected from March 2020 to April 2020, within the outpatient CT centres in Moscow outpatient clinics, Moscow, Russia. The CT scans were performed on *Canon (Toshiba) Aquilion* 64 units with standard scanner protocols and 8mm inter-slice distance. The dataset contains 36,753 slices of 1,110 volume from 1,110 patients. Specifically, 28,188 slices of 856 volume are *NCP* cases, and the rest 8,565 slices of 254 volume are *Normal*. Additionally, 50 CT volume were annotated on the region of infection areas such as *GGO* and consolidation. However, the ground truth of lung region segmentation is not provided. We segmented the lung regions of all the slices with our trained segmentation model and used the cropped slices as the clean dataset for the *COVID-19* classification task. Please note that the data were provided in NIfTI format by [294], which were

converted to PNG format, where a window (window center: -600HU, window width: 1200HU) was applied for re-scaling and normalising the pixel values.

- ***COVID-CTset*** ([320]). *COVID-CTset* dataset was collected from Negin radiology located at Sari in Iran between March 5th to April 23rd, 2020. This medical center uses a *SOMATOM Scope* model and *syngo CT VC30easyIQ* software version for capturing and visualising the lung *HRCT* radiology images from the patients. The dataset contains 63,849 slices of 377 volumes from 377 patients. Specifically, 15,589 slices of volumes scans are *NCP* cases, and the rest 48,260 slices of volumes scans are *Normal*. We randomly select 95 out of 282 *Normal* volumes to construct a balanced external test dataset. Again the ground truth of lung region segmentation is not provided. Thus, pre-segmentation is performed with our trained segmentation model to build a clean dataset with cropped slices.

5.4.2 Annotation of *COVID-19* CT Images

As discussed in Section 5.2.4, previous methods, such as [446] and [122], pre-trained the lung segmentation model from non-COVID datasets (*i.e.* cancer nodule segmentation datasets: *NSCLC* ([183]) and *LUNA16* ([383])), then applied it to the *COVID-19* CT scans. The domain gap between different datasets would cause significantly performance drop. For example, the *GGO* regions are typical characterises of *NCP* cases, which is an unseen feature in the cancer nodule dataset. Thus, their pre-trained models are likely to treat it as background (similar examples are shown in the top left and top right of the Fig.5.7). To address the challenges and train a robust lung segmentation model, four trained medical students (trainee doctors after training on the annotation tasks) from the University of Liverpool manually annotated 7,768 slices of *NCP*, *CP*, and *Normal* scans from *CC-CCII* ([481]) datasets. In detail, the boundaries of the left and right lungs are

traced via *Labelme* ([403]) annotation tool. Among the annotated 7,768 slices, 6,045 slice of 190 patients are *NCP*, 1,202 slices of 10 patients are *CP*, 521 slices of 10 patients are *Normal*. In this way, our annotated slices contain *NCP*, *CP* and *Normal* examples, which addresses the domain gap between the train and test dataset.

5.4.3 Evaluation Metrics

Segmentation Metrics Typical segmentation metrics, such as Dice similarity score (*Dice*), Mean Absolute Error (*MAE*) and Balanced Accuracy (*B-Acc*), are applied. 95% confidence intervals were calculated using 2000 sample bootstrapping for *Dice*, *MAE*, and *B-Acc*. Specifically, *B-Acc* is the mean value of *Sensitivity* and *Specificity*; *MAE* is used to measure the pixel-wise error between the segmentation and ground truth. *MAE* is defined as:

$$MAE = \frac{1}{w \times h} \sum_{x}^{w} \sum_{y}^{h} |S_p(x, y) - S_{gt}(x, y)|, \tag{5.13}$$

where, w and h are the width and height of the ground truth GT_s, and (x, y) denotes the coordinate of each pixel in GT_s.

Classification Metrics Typical classification metrics, such as *Sensitivity*, *Specificity*, F1 score (*F1*), Precision, Receiver Operating Characteristic Curves (*ROC Curve*), Area Under the ROC Curve(*AUROC*), are used for the evaluation of classification. In particular, *F1* is introduced to eliminate the interference of data imbalance. 95% confidence intervals were calculated using De Long's method [86] for *AUROC* and using 2000 sample bootstrapping for *Sensitivity*, *Specificity*, *F1* and *Precision*.

5.4.4 Experimental Details

In this section, we describe the experimental implementation details for the lung segmentation and *COVID-19* classification tasks, respectively. All the training processes are

performed on an Amazon Web Services *p3.8xlarge* node with four *Tesla V100 16GiB GPUs* and our workstation with four *GEFORCE RTX 3090 24GiB GPUs*. All the test experiments are conducted on a local workstation with *Intel(R) Xeon(R) W-2104 CPU* and *Geforce RTX 2080Ti GPU*. Notably, we have conducted extensive experiments to evaluate the sensitivity of the hyper-parameters, where γ has been set at 0.1, 0.05, 0.02, 0.01, 0.005, and T has been set at 2, 4, 6, 8, 10. In conclusion, we found no significant difference in diagnostic performance with paired t-test ($p > 0.05$) in the two evaluation settings, which proves that our model is robust to the hyper-parameters. Thus, we set the value of γ and T at 0.02 and 8 empirically, respectively.

Lung Segmentation

Implementation Details The original slice image is resized into 224×224 from $512 \times 512$by bilinear interpolation for CT slices and by nearest neighbour interpolation for binary annotation masks. To augment the dataset, we randomly rotate and horizontally flip the training dataset with the probability of 0.3. The rotation ranges from -30 to 30 degree. Besides, a random crop of size 112×112 are also applied both on the input image and ground truth during the training. Among all of our annotated data, 60 % of which are randomly selected as *Train* dataset, 10% are *Val* dataset and 30 % are *Test* dataset. The network is trained end-to-end by an Adam optimiser ([181]) for around 400 epochs, with a start learning rate of 0.01 and a cosine decay schedule ([248]). The batch size is set at 126. We adopt standard Dice Loss ([289]) for training the lung segmentation model.

COVID-19 Classification

Implementation Details. The input image size is 224×224 after lung segmentation. Similarly, to augment the dataset, we randomly rotate, horizontally and vertically flip the

Table 5.2: Quantitative segmentation results of the lung regions on CT slices. The performance is reported as *Dice* (%), *B-Acc* (%) and *MAE* (%). 95% confidence intervals are presented in brackets. We performed experiments with classic segmentation methods such as *U-Net* ([337]), *U-Net++* ([509]), and cutting-edge methods such as *PraNet* ([100]), *RBA-Net* ([276]), *CABNet* ([278]), *GRB-GCN* ([285]) and *BI-Gconv* ([281]). Notably, we sampled 120 vertices for *CABNet* [278] and *RBA-Net* [276] to construct a smooth boundary.

Methods	Metrics		
	Dice (%)↑	*B-Acc (%)*↑	*MAE (%)*↓
U-Net	95.7 (93.2, 97.6)	**96.9** (95.0, 98.4)	1.49 (1.12, 1.68)
U-Net++	94.1 (92.2, 96.0)	95.0 (93.2, 97.5)	1.98 (1.56, 2.23)
PraNet	95.2 (94.0, 96.6)	96.0 (95.1, 98.0)	1.55 (1.38, 1.68)
RBA-Net	96.2 (95.2, 98.0)	96.8 (95.9, 98.0)	**1.45** (1.29, 1.56)
CABNet	95.4 (93.8, 96.7)	96.4 (94.7, 98.1)	1.60 (1.42, 1.78)
GRB-GCN	**96.6** (94.9, 97.9)	96.7 (95.8, 97.9)	1.50 (1.32, 1.68)
BI-GConv	96.3 (94.8, 98.0)	96.5 (94.7, 98.2)	1.52 (1.34, 1.69)

training dataset with the probability of 0.3. The rotation ranges from -30 to 30 degree. 10% of the *Train & Val* dataset are randomly selected as the validation dataset. The network is trained end-to-end for 400 epochs, with a start learning rate of 1e-4 and a cosine decay schedule ([248]). The optimiser is an Adam optimiser ([181]), the batch size is set at 48 and 36, for 2D and 3D *COVID-19* diagnosis training, respectively. We adopt standard Cross Entropy Loss for both 2D and 3D *COVID-19* classification respectively.

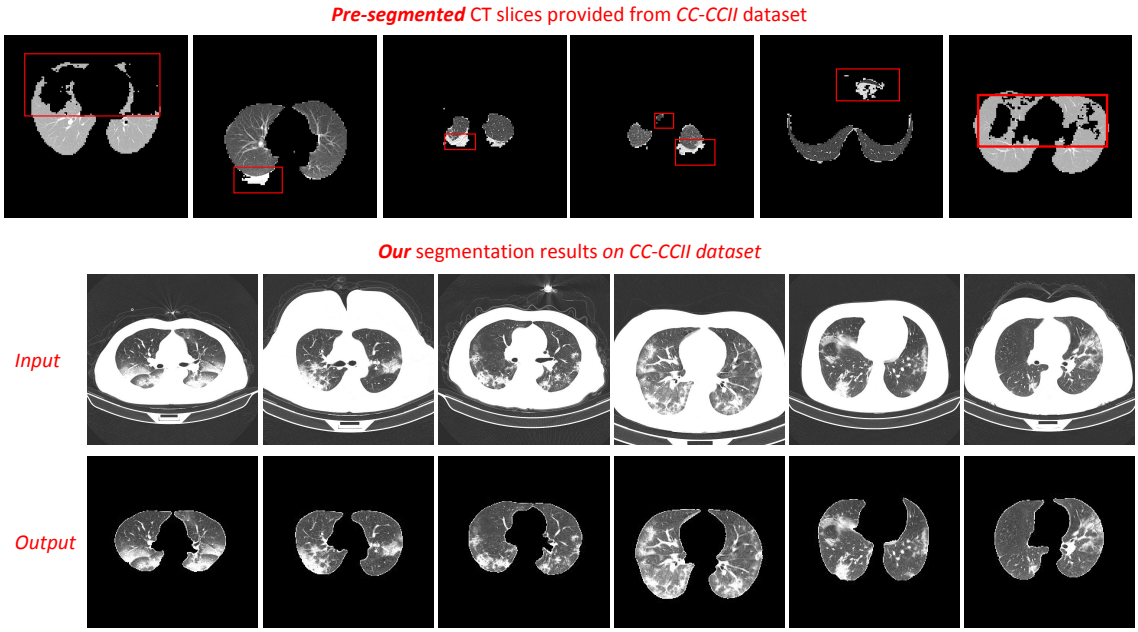


Figure 5.7: Qualitative comparison of pre-segmented slices and our segmentation results on *CC-CCII* ([481]) dataset. The top row is the pre-segmented slices that are provided by *CC-CCII* and the bottom row shows our segmentation examples on un-segmented cases. Red bounding boxes indicate the pre-segmented slices' false positive or false negative predictions. In particular, the top left and top right examples illustrate a typical false negative prediction, where the potential *GGO* regions may be treated as background, as the patient-level label for this case is *COVID-19* positive. Such false negative segmentation would perturb the subsequent *COVID-19* classification model training because there is no infection areas or diagnosis characteristics left in the segmented CT slices. On the other hand, our segmentation model can produce a complete lung region, even when there is a large number of infection regions (*e.g. GGO*). *Please note that CC-CCII only provides the pre-segmented CT slices without the original ones, thus we cannot intuitively compare the segmentation results with the same examples.*

Table 5.3: Quantitative comparisons between *Ours* and previous 3D CT based COVID-19 diagnosis methods, such as *CCT-Net* ([122]), *C19C-Net* ([21]), *COVNet* ([207]), *DeCoVNet* ([430]), *ASCo-MIL* ([135])). The performance is reported as *F1 (%)*, *Precision (%)*, *Specificity (%)*, *Sensitivity (%)*, *AUROC (%)*. 95 % confidence intervals are presented in brackets.

Methods	Learning Ability					Generalisation Ability				
	F1 (%)↑	*Precision(%)*↑	*Specificity(%)*↑	*Sensitivity(%)*↑	*AUROC(%)*↑	*F1 (%)*↑	*Precision(%)*↑	*Specificity(%)*↑	*Sensitivity(%)*↑	*AUROC(%)*↑
CCT-Net	76.8 (72.9, 80.6)	83.5 (81.0, 86.1)	84.4 (81.2, 87.4)	78.1 (74.6, 81.5)	96.1 (94.4, 97.1)	71.6 (62.8, 79.2)	86.6 (77.8, 94.2)	90.5 (84.2, 96.0)	61.1 (50.5, 71.3)	85.9 (80.1, 90.7)
C19C-Net	66.2 (61.5, 70.8)	71.2 (66.2, 75.9)	80.0 (76.8, 82.9)	70.2 (66.3, 74.0)	86.7 (83.9, 88.4)	70.4 (63.5, 76.2)	56.8 (48.7, 64.3)	29.5 (20.2, 38.8)	**92.6** (87.4, 97.8)	80.0 (73.2, 86.0)
COVNet	59.6 (54.9, 64.6)	73.7 (64.5, 79.4)	75.5 (71.5, 78.7)	68.0 (63.9, 72.0)	87.5 (84.8, 89.3)	33.6 (22.4, 43.7)	70.0 (52.0, 85.7)	90.5 (84.0, 96.0)	22.1 (14.1, 30.6)	71.5 (63.7, 78.6)
DeCoVNet	91.2 (88.5, 93.7)	91.6 (89.1, 94.0)	95.0 (93.4, 96.5)	91.3 (88.6, 93.7)	97.5 (96.7, 98.6)	68.8 (59.6, 76.2)	87.1 (78.2, 94.9)	91.6 (85.7, 96.7)	56.8 (46.5, 67.0)	85.1 (79.2, 90.2)
ASCo-MIL	76.5 (72.5, 80.6)	79.6 (76.1, 82.9)	86.2 (83.8,88.4)	77.9 (74.2, 81.5)	91.2 (88.9, 93.0)	60.7 (50.7, 69.7)	88.0 (78.4, 96.1)	93.7 (88.5, 97.9)	46.3 (36.4, 56.6)	82.1 (75.9, 87.8)
Ours	**94.9** (93.0, 96.8)	**95.1** (93.3, 96.9)	**97.1** (95.9, 98.2)	**94.9** (93.1, 96.8)	**98.7** (97.6, 99.4)	**88.0** (82.3, 92.7)	**96.3** (91.5, 100.0)	**96.8** (92.9, 100.0)	81.1 (72.9, 88.7)	**91.8** (84.6, 93.3)

5.5 Results

5.5.1 Lung Segmentation

Fig.5.7 shows the qualitative lung segmentation result of the pre-segmented slices (provided by *CC-CCII*) and our segmentation results on *CC-CCII*. Table. 5.2 shows the quantitative results of classic segmentation models, such as *U-Net* ([337]), *U-Net++* ([509]), and cutting-edge methods such as *PraNet* ([100]), *RBA-Net* ([276]), *CABNet* ([278]), *GRB-GCN* ([285]) and *BI-Gconv* ([281]). There are no significant differences between these models. Among them, *GRB-GCN* ([285]) achieves the best performance of 96.6 % *Dice*, outperforming *U-Net* ([337]) and *U-Net++* ([509]) by 0.9 and 2.7 %.

5.5.2 COVID-19 Diagnosis

This section provides the classification results in two evaluation settings with the pre-segmented COVID-19 CT data. Firstly, we train, validate and test our model on *CC-CCII* dataset (*seen data*) only, where there are three classes such as *Normal*, *NCP*, and *CP*. In this way, the learning ability of our model can be illustrated on the *seen data*. Secondly, in

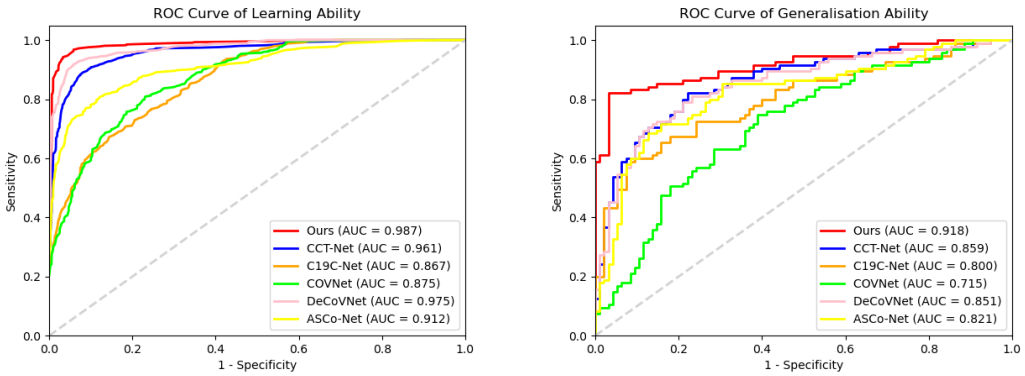


Figure 5.8: *ROC Curve* comparisons between *Ours* and previous 3D CT based COVID-19 diagnosis methods, such as *CCT-Net* ([122]), *C19C-Net* ([21]), *COVNet* ([207]), *DeCoVNet* ([430]), *ASCo-MIL* ([135])). Two evaluation settings of *learning Ability* and *Generalisation Ability* are presented.

order to address the unbalanced classes issue of *Mosmed*, we combine the *Normal* class's data from *CC-CCII* and all of the data from *MosMed*, to train and validate our model, while test on *COVID-CTset* (*unseen data*). There are two classes in this setting, such as *Normal* and *NCP*. In this way, we demonstrate the generalisation ability of our model on the *unseen data*. Generalisability is essential for the real-world COVID-19 diagnosis task, because of different domains of data *w.r.t.* scanning machine types, protocol standards, data sources. The details of data settings in these two schemes can be found in Table.5.1. The quantitative comparison results on respective test datasets of two evaluation settings are shown in Table.5.3, where previous 3D CT based COVID-19 diagnosis methods such as *CCT-Net* ([122]), *C19C-Net* ([21]), *COVNet* ([207]), *DeCoVNet* ([430]), *ASCo-MIL* ([135]) are presented. Notably, their results are reproduced by using their open-source code, and experiments are conducted under the same settings as *Ours* with our pre-segmented lung CT images.

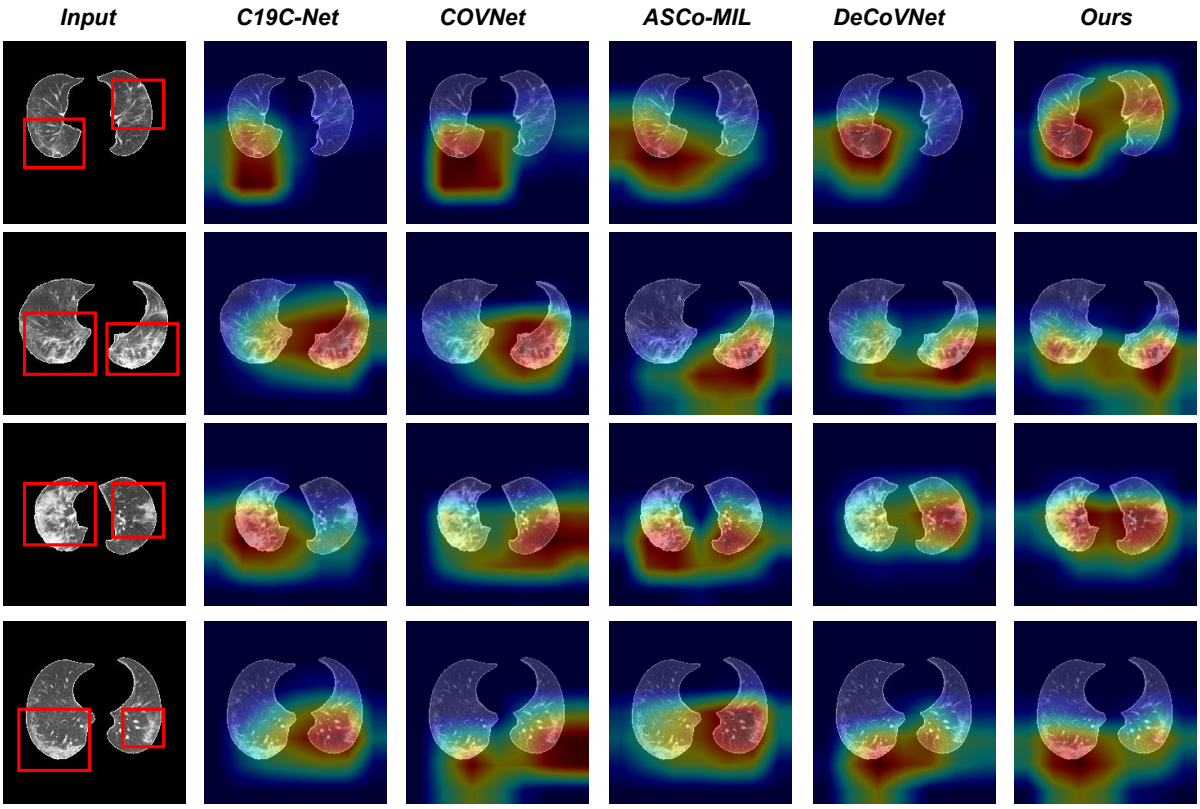


Figure 5.9: Qualitative comparisons between *Ours*, *C19C-Net* ([21]), *COVNet* ([207]), *ASCo-MIL* ([135]) and *DeCoVNet* ([430]). Specifically, attention heatmaps visualisation of *Grad-CAM* on *NCP* patients are presented in each row. *Ours* has a more precise and comprehensive activate area that encompasses more diagnosis characteristics, including *GGO*, multi-focal patchy consolidation and bilateral patchy shadows.

Learning Ability

Table.5.3 shows the quantitative comparison results in terms of the learning ability between *Ours* and previous 3D CT based COVID-19 diagnosis methods on *CC-CCII* dataset. *Ours* obtains an average of 94.9 *F1*, which outperforms the pooled 2D slice features based methods, such as *CCT-Net* ([122]), *C19C-Net* ([21]), *COVNet* ([207]) by 23.6 %, 43.4 % and 59.2 %, respectively. In addition, *Ours* outperforms the 3D level CNN based approaches *DeCoVNet* ([430]) by 4.1 %, outperforms the attention score based *MIL* method *ASCo-MIL* ([135]) by 24.1 %. Fig.5.8 demonstrates the *ROC Curve* comparison between the aforementioned methods. *Ours* achieves the best *AUROC* of 98.7 %. Notably, the macro-averaged performance (aka unweighted mean of per-class performance) of three classes with one vs rest calculation setting [1] on learning ability is presented in Table. 5.3 and Fig. 5.8.

Generalisation Ability

To evaluate the generalisation ability of the proposed model, we evaluate and compare *Ours* with previous 3D CT based COVID-19 diagnosis approaches with external test data (*unseen data*). The generalisation ability part of Table.5.3 shows the quantitative results. *Ours* achieves the best *F1* of 88.0 %, which outperforms the cutting-edge COVID-19 diagnosis methods *DeCoVNet* ([430]) and *ASCo-MIL* ([135]) by 27.9 % and 45.0 %. Fig.5.8 shows the *ROC Curve* comparison. *Ours* achieves the best *AUROC* of 91.8 %.

Attention Heat maps Visualisation

Fig.5.9 demonstrates the attention heat maps generated by using the gradient-weighted class activation mapping (*Grad-CAM*) ([349]). Specifically, *Grad-CAM* results on different slices of different *NCP* patients are presented in each row of the figure. We compare *Ours*

[1] https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html#sphx-glr-auto-examples-model-selection-plot-roc-py

Table 5.4: Computational efficiency. Model size, *FLOPs*, and inference time of different 3D CT based COVID-19 diagnosis methods on a $224 \times 224 \times D$ input volume.

	CCT-Net	*C19C-Net*	*COVNet*	*DeCoVNet*	*Ours*
Params (M)	24.8	23.8	23.5	0.35	15.0
FLOPs (G)	67.1	39.0	65.8	28.9	35.0
Inference Time (s)	1.2	1.2	1.1	1.1	1.1

with previous methods such as *C19C-Net* ([21]), *COVNet* ([207]), *ASCo-MIL* ([135]), *CCT-Net* ([122]), and present them in each column. *Ours* has a more accurate and comprehensive activate area that covers more diagnosis characteristics, such as *GGO*, multi-focal patchy consolidation and bilateral patchy shadows, which are highlighted within red bounding box in the figure. Notably, all the compared methods in Fig. 5.9 adopted at least the same D slices as ours to make the inference and prediction. Specifically, *C19C-Net* ([21]) and *COVNet* ([207]) used the same selected D slices, which is also aligned with their original implementation. *ASCo-MIL* ([135]) and *DeCoVNet* ([430]) used all of the slices in a CT scan to make the inference, thus includes the selected D slices.

Computational Efficiency

Table.6.4 presents the number of parameters (M), floating-point operations $(FLOPs)$ and inference time (s) of the compared models. Notably, ignoring the slices selection process of the first stage, we represent the proposed *BA-GCN* as *Ours* to compare with other methods in the Table.6.4. *Ours* adopted a light-weight backbone network of *MF-Net* to extract the 3D level of features, which leads to a relatively smaller model size as 15.0 M parameters.

Table 5.5: Ablation study of lung segmentation on *CCT-Net* ([122]), *DeCoVNet* ([430]) and *Ours*. *w/o Seg* represents without lung segmentation pre-process; *w/ Our seg* represents adopting our fully supervised lung segmentation method. The performance is reported as *F1 (%)*, *AUROC (%)*. 95 % confidence intervals are presented in brackets, respectively.

Methods	Learning Ability		Generalisation Ability	
	F1 (%)↑	*AUROC (%)*↑	*F1 (%)*↑	*AUROC (%)*↑
CCT-Net, w/o Seg	82.3 (80.0, 84.6)	97.2 (95.2, 98.5)	51.0 (48.7, 54.0)	65.3 (63.2, 68.1)
CCT-Net	75.0 (73.0, 78.1)	95.1 (92.5, 97.7)	69.0 (66.9, 72.1)	83.8 (81.1, 85.9)
CCT-Net, w/ Our seg	76.8 (72.9, 80.6)	96.1 (94.4, 97.1)	71.6 (62.8, 79.2)	85.9 (80.1, 90.7)
DeCoVNet, w/o Seg	93.9 (91.0, 95.5)	99.2 (97.1, 99.8)	57.3 (55.8, 60.2)	76.5 (74.1, 78.8)
DeCoVNet	88.7 (86.0, 90.3)	95.4 (93.2, 97.7)	66.7 (64.4, 68.9)	82.0 (80.0, 84.7)
DeCoVNet, w/ Our seg	91.2 (88.5, 93.7)	97.5 (96.7, 98.6)	68.8 (59.6, 76.2)	85.1 (79.2, 90.2)
Ours, w/o Seg	**96.8** (94.7, 98.9)	**99.4** (98.1, 99.7)	71.3 (69.2, 73.5)	84.3 (82.0, 86.6)
Ours	94.9 (93.0, 96.8)	98.7 (97.6, 99.4)	**88.0** (82.3, 92.7)	**91.8** (84.6, 93.3)

5.6 Ablation Study

We conduct thorough ablation studies, and all the results demonstrate our model's effectiveness. As an illustration, the ablation results for the lung segmentation and model components are elaborated as follows.

5.6.1 Need of Lung Segmentation Pre-process

Lung segmentation is an essential pre-processing step in this task. Please note that the original *CCT-Net* ([122]) adopted a pre-trained lung segmentation model on other CT datasets (non-COVID) and the original *DeCoVNet* ([430]) used an unsupervised approach to segment the lung regions as the pre-process for subsequent classification task. In this experiment, we used our pre-segmented lung CT images (*w/ Our seg*) to provide a more accurate cropped lung regions for their methods. Table.5.5 shows that *w/ Our seg* can

boost their original classification performance of *F1* by 2.4 %, 2.8 % and 3.8 %, 3.1 % in the *Learning Ability* and *Generalisation Ability* experiment settings, respectively. This can demonstrate the importance and the benefits of our fully-supervised lung-segmentation model in the task of 3D CT based COVID-19 classification. Additionally, Table.5.5 shows that the three methods without lung pre-segmentation (*w/o Seg*) can produce a better classification performance on the non-segmented CT data under the *Learning Ability* experiment setting, than the one with segmentation *w/ Our Seg*. However, the qualitative results (Fig.5.11) prove that, such model trained on non-segmented data, can only learn a specific format pattern of different classes rather than the real radiographic diagnosis characteristics (*i.e. GGO* for *NCP*), because of specific scanning machine types, protocol standards, data sources of different classes. Also, due to the evaluation setting under *Learning Ability* of test on *seen data*, such specific format patterns also exist in the test dataset, which helps the models achieve 'excellent' classification results, rather than learning the real diagnosis features.

Differently, under the experiment setting of *Generalisation Ability*, those methods *w/o Seg* conducts a terrible classification performance because an external test dataset (*unseen data*) is introduced to evaluate the trained model, where the aforementioned specific format patterns do not exist. This further demonstrates the importance of pre-segmentation, generalisation ability and external test dataset (*unseen data*) in this task. More visualisation comparisons and discussions related to this challenge are refereed to Section 5.7.1 and Fig.5.11.

5.6.2 Model Components

This section presents the results of our ablation study on model components. We evaluate the effectiveness of the proposed *UC-MIL*, *BA-GCN* modules, and present the quantitative

Table 5.6: Ablation study on the effectiveness of the proposed *UC-MIL* and *BA-GCN*. The performance is reported as *F1 (%)*, *AUROC (%)*. 95 % confidence intervals are presented in brackets, respectively.

Methods	Learning Ability		Generalisation Ability	
	F1 (%)↑	*AUROC (%)*↑	*F1 (%)*↑	*AUROC (%)*↑
Ours w/o BA-GCN	82.6 (79.0, 86.0)	94.6 (92.9, 96.2)	81.0 (73.9, 87.2)	83.9 (77.1, 89.8)
Ours w/o UC-MIL (random)	91.5 (89.1, 93.3)	97.4 (95.5, 98.8)	72.6 (70.9, 74.1)	87.0 (85.2, 88.7)
Ours w/o UC-MIL (symmetrical)	91.9 (90.1, 93.3)	97.9 (95.8, 98.8)	73.1 (71.9, 75.2)	86.8 (84.8, 88.0)
Ours	**94.9** (93.0, 96.8)	**98.7** (97.6, 99.4)	**88.0** (82.3, 92.7)	**91.8** (84.6, 93.3)

results in Table.5.6. Firstly, we remove the *BA-GCN* and keep the rest of our model, conduct *Ours w/o BA-GCN* in the Table. Secondly, we replace *UC-MIL* with random and symmetrical slice sampling rules to select the fixed number of slices for each CT scan in the same manner as ([145]). In these two cases, the proposed bilateral graph model becomes an unilateral graph model, because there is no 2D feature information included in the vertices features. Specifically, for both evaluation settings (*Learning and Generalisation Abilities*), *BA-GCN* helps our model gain an average 9.4 % performance boost *w.r.t. F1* and 4.3 % performance boost *w.r.t AUROC*; *UC-MIL* outperforms the hand-crafted slice sampling rules, *e.g. random* and *symmetrical*, by 12.5 % and 11.9 % *F1* on average, respectively.

Additionally, we conduct extensive experiments to evaluate the effectiveness of the proposed components inside *UC-MIL* and *BA-GCN* modules respectively, such as backbones, *Uncertainty-Aware* mechanism, *Consensus-Assisted* mechanism, *BA-GConv* layers, *etc.*. The experimental results are elaborated as follows, which prove their effectiveness.

UC-MIL

Backbone Network We conduct experiments to evaluate the effectiveness of different backbone models in the proposed *UC-MIL*. We adopt several classic 2D classification back-

Table 5.7: Ablation study on the effectiveness of the *UC-MIL*'s backbone networks and the proposed *Uncertainty-aware Consensus-assisted* mechanism. Specifically, we respectively replace the proposed *UC-MIL* to another two classic *MIL* methods, such as [48] (*w/ Instance-based*) and [158] (*w/ Embedding-based*). The performance is reported as *F1 (%)*, *AUROC (%)*. 95 % confidence intervals are presented in brackets, respectively.

Methods	Learning Ability		Generalisation Ability	
	F1 (%)↑	*AUROC (%)*↑	*F1 (%)*↑	*AUROC (%)*↑
Backbone				
w/ ResNet34	93.2 (90.9, 95.5)	98.0 (96.1, 99.1)	86.8 (84.7, 88.1)	90.5 (88.7, 92.0)
w/ ResWide50	93.3 (91.7, 95.0)	97.7 (95.4, 98.9)	86.0 (84.2, 88.1)	90.2 (88.4, 92.3)
w/ EfficientNetB3	90.2 (88.1, 92.3)	96.0 (93.9, 98.0)	84.6 (82.7, 86.1)	88.1 (86.5, 89.7)
w/ Res2Net50	91.7 (89.9, 93.2)	96.8 (94.7, 98.0)	85.0 (83.1, 87.2)	88.7 (86.6, 89.9)
Ours	**94.9** (93.0, 96.8)	**98.7** (97.6, 99.4)	**88.0** (82.3, 92.7)	**91.8** (84.6, 93.3)
Component				
w/o Uncertainty	92.2 (90.1, 94.1)	97.0 (95.8, 98.1)	86.2 (94.9, 88.3)	90.2 (88.1, 92.1)
w/o Consensus	92.2 (90.4, 94.6)	97.7 (95.1, 98.6)	85.8 (83.3, 87.0)	89.4 (87.2, 90.5)
w/ Instance-based	88.7 (86.0, 90.1)	95.5 (93.3, 96.8)	81.5 (80.0, 83.1)	85.7 (83.3, 87.1)
w/ Embedding-based	89.9 (87.4, 91.2)	95.9 (93.3, 97.6)	83.0 (81.0, 85.2)	87.0 (85.1, 88.9)
Ours	**94.9** (93.0, 96.8)	**98.7** (97.6, 99.4)	**88.0** (82.3, 92.7)	**91.8** (84.6, 93.3)

bones, such as *ResNet* ([143]) variants (*e.g.* 18, 34, 50, 101), and cutting-edge classification backbones such as *ResWide* ([474]) variants (*e.g.* 50, 101), *ResNeXt* ([450]) variants (*e.g.* 50, 101), *EfficientNet* ([380]) series (*e.g.* B0, B3, B5, B7) and *Res2Net* ([114]) variants (*e.g.* 50, 101). For each model's variants, we present the best performance in Table.5.7 for an intuitive comparison. *Ours* achieves the best performance of 94.9 % and 88.0 % *F1* with *ResNeXt50* and *ResNet18* as the backbone in *Learning Ability* and *Generalisation Ability* settings, respectively.

Uncertainty & Consensus Mechanism We evaluate the effectiveness of the proposed *Uncertainty-aware* mechanism and *Consensus-assisted* mechanism respectively. In detail,

we remove each of them correspondingly and remain the rest of the model unchanged, which are represented as *w/o Uncertainty* and *w/o Consensus* in Table.5.7. As a result, the reliable slices selection process will rely on the ranked order of consensus-assisted instance probability ($P^{(C)}_{x_{i,j_r}}$, $x_{i,j_r} \in \Omega$) and the ranked order of uncertainty-aware instance probability ($P^{(C)}_{\tilde{I}_{x_{i,j}}}$), respectively. Specifically, *Uncertainty-aware* and *Consensus-assisted* modules boost the performance of *F1* by 2.9 % and 2.9 % on *Learning Ability* and 2.0 % and 2.6 % on *Generalisation Ability*, respectively.

Multiple Instance Learning To further verify the usefulness of the proposed *UC-MIL*, we respectively replace it with another two classic *MIL* methods, [48] (*w/ Instance-based*) and [158] (*w/ Embedding-based*), shown in Table.5.7. Notably, *w/ Instance-based* can be seen as our *UC-MIl* but without *Uncertainty* and *Consensus* mechanisms. As for *w/ Embedding-based* ([158]), as we discussed in Section 5.2.1, all previous 3D CT based COVID-19 diagnosis methods ([77,135,221]) adopted its attention scoring system. Specifically, we adopted the same backbone framework as *Ours*, but a trainable attention score-based pooling mechanism from [158]. In detail, two fully-connected layers with *Softmax* as the activation functions are applied to learn a weighted average of instances (low-dimensional embeddings). We trained those two models ([48] [158]) with all of the training CT slices/instances under the same experiment settings as ours. In Table.5.7, *Ours* outperforms *w/ Instance-based* and *w/ Embedding-based* by an average of 7.5 % and 5.8 % *F1* on both evaluation settings. Notably, for *w/ Instance-based* ([48]), we selected the top D instances (CT slices) according to the ranking of the predicted probability of instances, which is straightforward to implement and has been adopted by previous MIL methods ([48,188,375]). On the other hand, for *w/ Embedding-based* ([158]), we used the ranked attention weights to select the corresponding top D instances, which is similar to the previous methods ([158,202,354]). Those selected top D instances were then used as the 3D

Table 5.8: Ablation study on the effectiveness of the *BA-GCN*'s backbone networks and the proposed *Bilateral Adaptive Graph Convolution*. Specifically, we respectively replace the proposed *BA-GConv* layer to another three cutting-edge graph reasoning based classification layers, such as *SGR* ([223]), *DualGCN* ([483]) and *GloRe* ([70]). The performance is reported as *F1 (%)*, *AUROC (%)*. 95 % confidence intervals are presented in brackets, respectively.

Methods	Learning Ability		Generalisation Ability	
	F1 (%)↑	*AUROC (%)*↑	*F1 (%)*↑	*AUROC (%)*↑
Backbone				
w/ 3D-ResNet50	93.2 (91.4, 95.1)	98.0 (96.3, 98.9)	87.1 (86.2, 88.7)	91.0 (89.7, 92.8)
w/ 3D-ResNeXt50	93.2 (91.7, 95.5)	97.7 (95.8, 98.8)	87.3 (85.8, 89.9)	91.2 (89.7, 93.0)
w/ 3D-EfficientNetB0	90.7 (87.1, 91.3)	95.5 (92.8, 97.0)	85.2 (82.5, 84.0)	89.9 (87.0, 91.2)
Ours (w/ MF-Net)	**94.9** (93.0, 96.8)	**98.7** (97.6, 99.4)	**88.0** (82.3, 92.7)	**91.8** (84.6, 93.3)
Component				
w/ SGR	90.3 (88.2, 92.8)	95.5 (93.0, 97.7)	85.7 (83.6, 87.6)	86.0 (83.9, 88.0)
w/ DualGCN	91.1 (89.4, 92.9)	96.0 (94.1, 97.8)	85.9 (83.8, 87.1)	86.3 (84.9, 87.7)
w/ GloRe	90.8 (88.5, 92.0)	95.9 (93.1, 97.0)	86.1 (84.2, 88.2)	86.6 (84.1, 88.0)
Ours	**94.9** (93.0, 96.8)	**98.7** (97.6, 99.4)	**88.0** (82.3, 92.7)	**91.8** (84.6, 93.3)

input for our proposed *BA-GCN*.

As we have noted, previous instance-level *MIL* methods yield promising classification results in the seen data (*i.e.* the evaluation of *Learning Ability*), but poor on unseen data (*i.e.* the evaluation of *Generalisation Ability*). On the other hand, *Ours* can achieve more consistent results on the *unseen* data, with the benefit of the proposed uncertainty-aware and consensus-assisted mechanisms.

Bilateral Adaptive Graph Convolution Network

Backbone Network. We conduct experiments to evaluate the effectiveness of different backbone models in the proposed *BA-GCN*. We adopt several classic classification back-

bones, such as *3D-ResNet* ([143]) variants (*e.g.* 18, 34, 50), and cutting-edge classification backbones such as *MF-Net* ([69]), *3D-EfficientNet* ([380]) variants (*e.g.* B0, B3, B5) and *3D-ResNeXt* ([450]) variants (*e.g.* 50, 101). For each model's variants, we present the best performance in Table.5.8 for an intuitive comparison. *Ours* achieves the best performance of 94.9 % and 88.0 % *F1* with *MF-Net* as the backbone in *Learning Ability* and *Generalisation Ability* settings, respectively.

Graph Convolution To further verify the usefulness of the proposed *BA-GCN*, we respectively replace it to another three cutting-edge graph-based reasoning methods, such as *SGR* ([223]), *DualGCN* ([483]) and *GloRe* ([70]), shown in Table.5.8. In detail, we retain the same input vertices (X_{all}) and replace the proposed *BA-GConv* layer to their corresponding graph convolution layers, where *SGR* makes use of the knowledge graph mechanism, *DualGCN* investigates the coordinate space and feature space graph convolution, and *GloRe* makes use of the projection and re-projection mechanisms to reason about the relationships of different regions. In this way, the compared GCNs will consider both 2D and 3D levels of information from the input vertices. Table.5.8 shows that *Ours* achieves more accurate and reliable results, and outperforms *SGR*, *DualGCN* and *GloRe* by an average of 3.9 %, 2.9 % and 3.4 % *F1* on both the evaluation settings.

5.7 Discussion

5.7.1 Hidden Challenges of the *COVID-19* Dataset

CC-CCII ([481]) is now the largest public available 3D CT dataset for the COVID-19 diagnosis, with patients' CT scans of *NCP*, *CP* and *Normal* classes. Many previous methods ([145,149,381,446]) reported evaluation results on it, however, rarely discussed the importance of pre-segmentation process. In Table.5.5, we present a better quantification

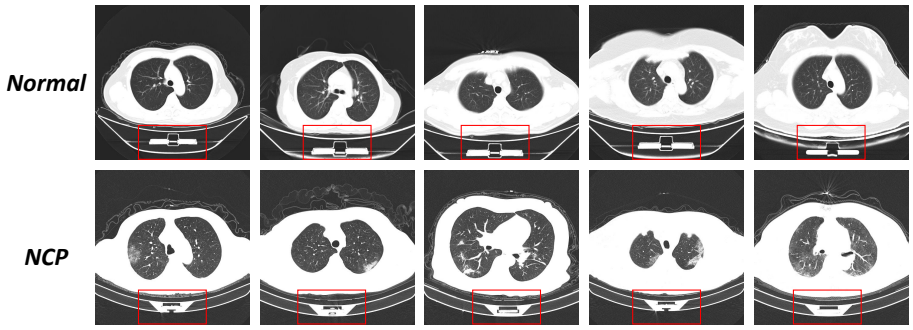


Figure 5.10: CT slices are randomly selected from different patients. The top and bottom rows represent *Normal* and *NCP* classes, respectively. Red bounding box highlights the differences between the scanner beds in the two classes.

results of training the proposed model without pre-segmentation process (*Ours, w/o Seg*), than the one with the pre-segmentation (*Ours*). Similar circumstances are also observed with previous methods in Table.5.5, such as *CCT-Net, w/o Seg* and *DeCoVNet, w/o Seg*. However, the trained models without pre-segmentation may only learn a specific format pattern of different classes, rather than the true radiographic diagnosis characteristics (*i.e. GGO* for *NCP*), because of specific scanning machine types, protocol standards, data sources for different classes in the dataset. For example, in Fig.5.10 we show ten randomly selected CT slices of different patients from *Normal* and *NCP* classes. The model can easily learn the difference between the specific scanner bed part of different classes (highlighted with red bounding box).

To further prove the necessity of pre-segmentation in this task, we visualise the trained model's attention heat maps, which are generated by using the *Grad-CAM*. In Fig.5.11, it shows that the models without pre-segmentation (*Ours, w/o Seg*) look at other regions (*e.g.* scanner bed) rather than the diagnosis characteristics part (*e.g.* GGO) of the lungs in the *NCP* CT images.

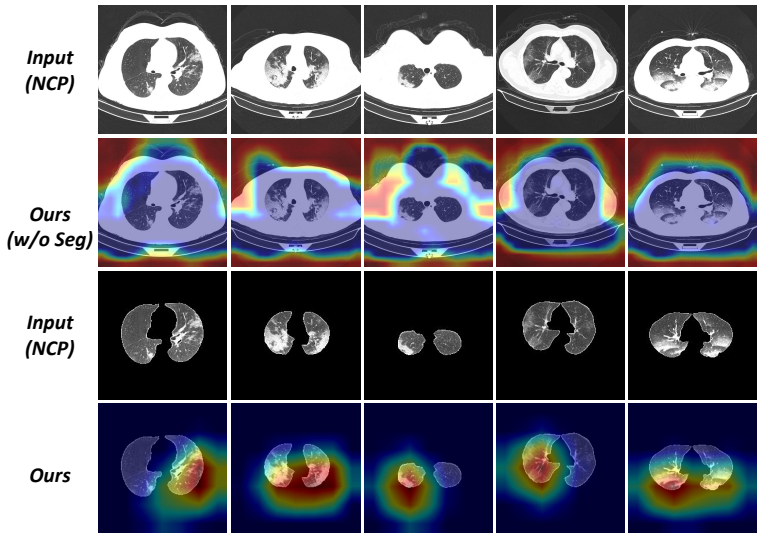


Figure 5.11: Qualitative comparison of *Grad-CAM* on the same input with and without pre-segmentation step. Models without pre-segmentation (*Ours, w/o Seg*) attend to other regions (*e.g.* scanner bed) rather than the discriminatory parts (*e.g.* GGO) of the lung regions in the *NCP* CT images.

5.7.2 Limitations of the Proposed Model

Our proposed model achieves accurate classification results on three largest public available CT dataset *w.r.t. Learning Ability* and *Generalisation Ability* evaluation settings. However, one limitation of our model is two-stage, which requires a relatively longer inference time or training time compared to other one-stage methods. This is because we proposed *UC-MIL* for 2D feature extraction and trustworthy slices selection on the first stage, then, we propose *BA-GCN* to extract 3D features, and aggregate the 2D and 3D information for a more comprehensive level of feature reasoning on the second stage. Such a design increases the diagnostic accuracy but also consumes more time to infer and train. Compared to the other methods in Table. 5.3, *Ours* takes 30.0 more hours on average for training the first stage of *UC-MIL* due to MIL's specific training mechanism. This is similar to *ASCo-MIL* ([135]), which is also a MIL-based method. However, we believe

the training model process is often one-off, while inference speed plays a more important role in evaluating the algorithm and applying to the real applications. Specifically, *Ours* requires approx 0.26 seconds more inference time per 3D CT volume for both evaluation settings on average. In addition, if ignoring the slice selection process in the first stage for both *Ours* and other compared methods, we have demonstrated in the Table. 6.4 that all the methods have a similar inference time. However, *w.r.t.* the diagnosis of COVID-19, the diagnostic accuracy would matter more than the inference speed. This highlights the need of a trade-off between accuracy and running time when applying AI models to real world applications. On the other hand, our proposed *UC-MIL* works as the automatically reliable CT slices selection step in the first stage, rather than the handcrafted slice sampling rules or manual slices selection of previous methods. In other words, previous methods also belong to the two-stage pipeline, where they need to select CT slices in a handcrafted way in the first stage. However, our method can automatically work with raw CT images without any manually designed pre-processing steps.

5.7.3 Future Work

Future studies building on this work should may wish to focus on the first stage of reliable slices selection, as the second stage of graph-based 2D/3D feature reasoning processes will rely mainly on the the selected slices and the 2D features from the first stage as the input. Consequently, a collection of noisy input slices will inevitably introduce noise into the second stage and in turn perturbing the training process. The ablation study experiments of *UC-MIL* and *BA-GCN* in Table. 5.7 and Table. 5.8 of the original manuscript further support this view, that is, unreliable slices of the first stage lead to lower performance in the diagnosis of second stage, especially in the generalisation ability evaluation.

A potential concern of using automatically selected top D CT slices of *UC-MIL* as the

3D input for the 3D CNN backbone in *BA-GCN* could be that the non-adjacent top D CT slices may lack abundant spatial correlations along the channel axis, which may lead to insufficient usage of the potential of 3D CNN. To address this concern, we have experimentally demonstrated that such 3D input can be used to boost the COVID-19 diagnosis performance via the extracted 3D features in both *Learning Ability* and *Generalisation Ability* settings in TABLE. 8, compared to the one without 3D features (*Ours w/o BA-GCN* in Table. 6). Also, the same circumstance occurred and has been observed by many previous CT-based COVID-19 diagnosis studies ([103, 145, 218, 308, 381, 428]), where they sampled a fixed number of slices from adjacent CT slices, to form a 3D input volume with non-adjacent CT slices. Moreover, they have all proved that such 3D volume can be used for 3D CNN to extract COVID-19 diagnosis-related features and also achieve satisfying results. An extensive analysis of the relations between 3D CNN and non-adjacent CT slices' effectiveness will be of interest in future studies.

5.8 Conclusion

We have proposed a novel and comprehensive framework for diagnosing COVID-19 using CT scans of an arbitrary number of slices. It takes advantage of both 2D and 3D features of CT images by utilising the proposed *UC-MIL* and *BA-GCN* modules. Our experiments have demonstrated that our framework can locate the diagnosis characteristics in both *seen* and *unseen* evaluation settings by the graph-based information aggregation of trustworthy 2D and 3D features. Our approach is anticipated to be widely applicable to real-world applications.

Chapter 6

Researching Auxiliary Task Learning with Implicit Graph Representations

In this chapter, I research the auxiliary-task enhanced implicit graph representation in the task of object counting. Specifically, this section proposes an adaptive auxiliary task learning-based approach for transport object counting problems such as humans and vehicles. These problems are essential in many real-world tasks such as video surveillance, traffic monitoring, public security, and urban planning, to aid intelligent transportation systems. Unlike existing auxiliary task learning-based methods, we develop an attention-enhanced adaptively shared backbone network to enable both task-shared and task-tailored features that are learned in an end-to-end manner. The network seamlessly combines a standard Convolution Neural Network (CNN) and a Graph Convolution Network (GCN) for feature extraction and feature reasoning among different domains of tasks. Our approach gains enriched contextual information by iteratively and hierarchically fusing fea-

tures across different task branches of the adaptive *CNN* backbone. The whole framework pays special attention to objects' spatial locations and varied density levels, informed by object (or crowd) segmentation and density level segmentation auxiliary tasks. In particular, thanks to the proposed dilated contrastive density loss function, our network benefits from individual and regional context supervision, along with strengthened robustness. Experiments on six challenging multi-domain datasets demonstrate that our method achieves superior performance compared with state-of-the-art auxiliary task learning-based counting methods. Our code is publicly available [1].

6.1 Introduction

Object counting by inferring the number of objects in images or video contents is a crucial yet challenging computer vision task. This paper is primarily motivated to address human crowd counting problems whilst being applicable to other domains such as vehicle counting. Due to the occurrence of crowd gatherings in many scenarios such as parades, concerts, and stadiums, a robust and accurate crowd counting model plays an essential role in multimedia applications for security alerts, public space design, transportation management *etc.* [112].

As a result of Convolutional Neural Network's (*CNN*)'s exceptional feature learning capability, the performance of crowd counting methods has been steadily enhanced. Recent state-of-the-art methods, such as [284, 306], have demonstrated that a density map regression paradigm yields satisfactory results. In these methods, given an input image, a *CNN*-based network is used to regress the corresponding density map; the sum of the pixel values in the density map represents the total number of counts in the image. There are a number of challenging issues [112] such as significant scale changes, wide variations in density levels, and complex scene backgrounds, however, there is still considerable room

[1] https://github.com/smallmax00/Counting_With_Adaptive_Auxiliary

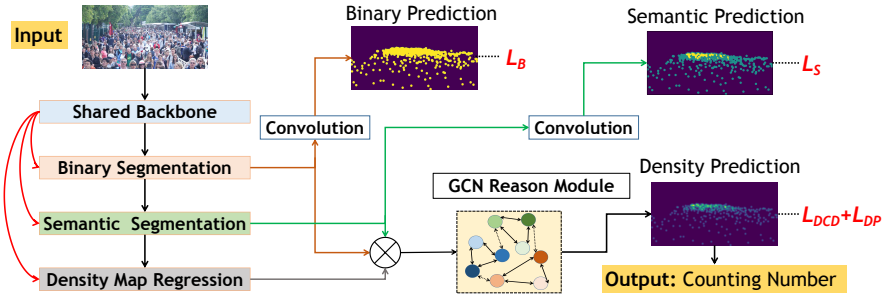


Figure 6.1: Overview of the proposed network structure in the scene of crowd counting. An attention-enhanced adaptively shared backbone network is proposed to enable both task-shared and task-tailored features learning. A novel Graph Convolution Network (GCN) reasoning module is introduced to tackle issues of cross-granularity feature reasoning among three different tasks. A novel loss function L_{DCD} is proposed to take into account more adjacent pixels for regional density difference, which strengthens the network's generalizability.

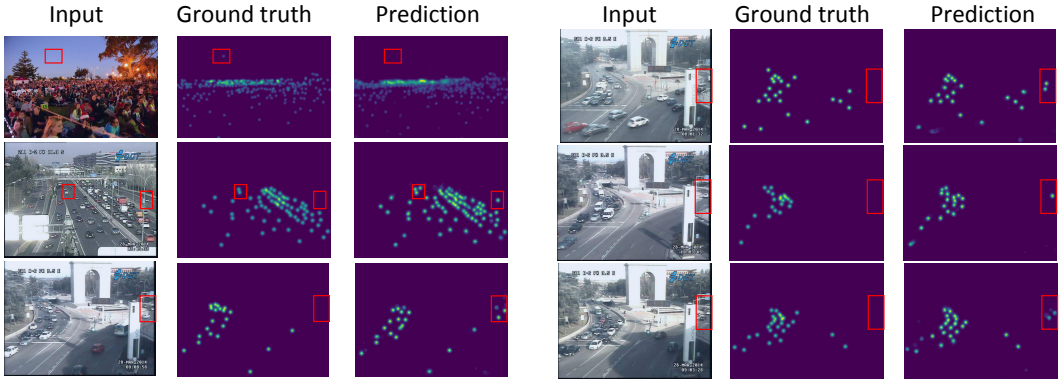


Figure 6.2: Comparison of our predictions and the ground truth. Our predictions are robust enough even when there are mislabeled or incorrectly labeled point annotations in the ground truth of crowd counting and vehicle counting datasets. Our model can indicate more accurate object locations or counting numbers compared with the ground truth. The red bounding boxes are used for better visualisation and comparison.

for counting performance improvement. Some previous methods [167, 229, 292, 359] rely on various types of information granularity in terms of 'auxiliary task learning' to address these issues. Using a single shared backbone network structure, these methods extract generalised features for all tasks. Unfortunately, this strategy may result in under-fitting, as the generalizable representation is frequently incapable of describing the comprehensive cross-granularity features across multiple tasks simultaneously [112]. Contrasting, our adaptive shared backbone network focuses on maximising the principal density map regression task and multi-granularity information augmentation from auxiliary tasks. Our backbone network has a multi-level information aggregation mechanism to repeatedly and hierarchically combine features from distinct stages and auxiliary branches. Note that, the term 'auxiliary task learning' is referred to as the feature learning of different density information granularity levels. Specifically, the crowd segmentation task and the density level segmentation task in Fig. 6.1 are the auxiliary tasks, and the density map regression task is the main task. We generated the ground truth of crowd segmentation and density level segmentation from the density map regression ground truth. Intuitively, no increase in information from the ground truth of auxiliary tasks is generated; however, the information is enhanced and specified through auxiliary tasks in terms of different density information granularity.

Given the auxiliary-task learning paradigm, we researched how to reason and fuse features from different tasks for density map regression. Crowd segmentation and density level segmentation feature domains have different granularity of representations. Direct fusion (element-wise multiplication or channel-wise concatenation) of three task branches' outputs might cause domain conflicts [252]. To improve counting accuracy, we exploited the nature of Graph Convolutional Networks (GCN) for information reasoning. GCN has showed promising reasoning ability on several computer vision problems, including scene

interpretation [217, 284] and image segmentation [275, 277, 278, 281, 283, 285], but has been rarely investigated in crowd counting. Our model projects a collection of pixels from a spatial-aware density feature map with similar density levels to each graph vertex and exploits a *GCN* to reason about the relations among graph vertices. This is different from a recent work [252], which directly treated cross-granularity feature maps as graph vertices and utilized a cascaded Graph Neural Network (GNN) to reason the cross-scale relationships.

In this work we present a novel loss function for density map regression. The commonly adopted Least Absolute Error (L1) or Least Square Error (L2) loss [236, 359, 488] assumes pixel-wise independence. However, two major flaws exist: (1) The estimated density map is over-smoothed [229], underestimating high-density regions and overestimating low-density parts. The model may focus on reducing count mistakes rather than regressing high-quality density maps, therefore it cannot reflect the true density levels. (2) Without a large receptive field, pixel-wise loss functions may ignore regional density level information during training [166]. Unbalanced low- and high-level density distributions might cause bias in training, reducing network resiliency. To overcome these concerns, we present a new loss function for density map regression called Dilated Contrastive Density Loss (L_{DCD}), where the density difference between dilated adjacent pixels provides extra regional supervision. Ablation studies conducted show that our proposed regional loss function outperforms pixel-wise losses in all datasets used in this work.

We conducted extensive experiments on seven well-known challenging counting benchmarks. Quantitative and qualitative results demonstrate that our model achieves state-of-the-art performance. To the best of our knowledge, we achieved the **best** counting performance among other auxiliary task-based counting methods on the NWPU-Crowd [419]

benchmark [2], which is currently the largest crowd counting benchmark. Our model is robust and generalizable, indicating incorrectly labeled or mislabeled object ground truths in the test datasets. Please refer to Fig 6.2 for more details.

In summary, this work makes the following contributions:

- We address the feature learning issues of the backbone network for auxiliary task-based methods in crowd counting challenges, by enabling task-shareable and task-specified feature learning simultaneously with a primary focus on the main task.
- We propose crowd segmentation and density level segmentation as auxiliary tasks in crowd counting with additional spatial crowd location and density level information enhancement. Moreover, a *GCN* model was proposed to reason about the cross-granularity feature relations between density map regression and other auxiliary tasks.
- We propose a novel loss function tailored for density map regression, strengthening the network's generalizability and improving the counting accuracy.

6.2 Related Work

In recent years, density map regression-based counting methods [22,65,92,219,233,234,236, 262,386,410,443,458,459,461,495,506] using *CNNs* have achieved good performance. As mentioned previously, they employ different learning strategies to address difficult issues such as variations in scale, alternate density levels, and complicated background scenes. Specifically, attention-based methods [55,97,151,166,288,368,410,422,453,477], auxiliary task-based methods [2,74,167,199,232,237,238,373,408,464,485], and different supervision-based methods [239,240,259,261,372,374,407,409,412,413,418,423] align closely with our proposed method presented in this work. We have elaborated the related works of the

[2] https://www.crowdbenchmark.com/nwpucrowd.html

aforementioned learning strategies in the following contents.

6.2.1 Attention-Based Counting

Visual attention mechanisms were applied among several works [55, 97, 151, 166, 288, 368, 410, 422, 453, 477] in crowd counting applications, which helps the network focus on valuable information and addresses several challenges. For example, *Miao et al.* [288] utilized a shallow feature-based attention module to highlight the regions of crowd interest and filter out the noise from background clutter. To tackle various density levels issues, *Jiang et al.* [166] employed an attention mask to refine the density map for adapting to different density levels. Furthermore, *Zhang et al.* [477] proposed the *Attention Neural Field* that incorporates non-local attention modules with conditional random fields to maintain multi-scale features and long-range dependencies, enabling control over the large-scale variation challenge of input crowd images. *Wan et al.* [406, 410] exploited the self-attention mechanism to adaptively generate density maps with different Gaussian kernel sizes, which is then used as the ground truth to supervise the model. The aforementioned methods adopt the attention mechanism as a feature enhancement module to implicitly address the crowd counting task challenges emphasised throughout this paper, including notable scale changes, large-scale density level variability, and complex scene backgrounds. Our model explicitly addresses these challenges through auxiliary tasks. On the other hand, our model adopts the attention mechanism to construct an adaptively shared backbone network, enabling task-shared and task-specific feature learning simultaneously.

6.2.2 Auxiliary Task-Based Counting

Recently, auxiliary task learning-based counting methods [2, 74, 167, 199, 222, 228, 232, 237, 238, 330, 373, 408, 464, 485] have attracted research attention because of their ability to cap-

ture extra granularity information and contextual dependencies for density map regression. Most methods utilize the potential of a model itself with auxiliary tasks, such as object detection, crowd segmentation, density level classification, *etc.*, to enhance the feature tuning for density map regression. For example, the task of patch-based density level classification [167, 242, 291, 359, 366, 367, 505] can enhance patch-wise density-level information, which helps to address the underestimation and overestimation problems of density map regression. However, it may be difficult to guide the pixel-wise density map regression via patch-wise density-level classification because of the gap between pixel-wise and patch-wise feature learning. In contrast, our model proposes a density level segmentation auxiliary task, which can be regarded as the pixel-wise density-level classification task. In this way, our model can enhance the pixel-wise density-level information to the pixel-wise density map regression task, aiming to address the challenges of wide variations of density levels.

Moreover, because the background regions in complex scenes contain confusing objects or similar appearances, the crowd segmentation task, adopted by previous methods [252, 292, 359, 421, 496], can provide spatial location information for the crowd, which highlights the foreground over the background and guides the network focus onto the region of interest. Our model also adopts the crowd segmentation task because of its superiority in spatial location information enhancement. In particular, *Luo et al.* [252] adopted crowd segmentation as the auxiliary task, then proposed a cascaded graph-based model to tackle the fusion of features between the crowd segmentation and density map regression tasks. This is similar to our learning paradigm, however, there are two significant difference: (1) They did not consider the density level information and only treated the features of the density map and crowd segmentation as the vertices in their proposed model. Alternatively, we incorporate the spatial information of crowd location, the semantic information of density level, and the main task of density map features, into the proposed vertices in

our model. (2) They treated the vertices equally. Specifically, they regarded the crowd segmentation and density map features as independent vertices, fusing and aggregating the information among them. However, the main task to estimate the counting number should be density map regression, hence they may introduce inevitable noise into the training process if the auxiliary task takes over. Differently, we project a collection of pixels from a spatial-aware density feature map with similar density levels to each graph vertex, thereby enhancing the main task vertices' spatial location awareness. Also, we project the long-range density level dependency among every pixel into the adjacency matrix, boosting the main task vertices' semantic density level awareness. Please see Section 6.3.5 and Fig. 6.5 for details.

6.2.3 Learn to Count with Different Supervisions

Instead of tackling the counting task through different learning frameworks or strategies, recent methods [50, 76, 239, 240, 259, 261, 372, 374, 407, 409, 412, 413, 418, 420, 423] have paid attention to the way of supervisions. For example, *Sravya et al.* proposed a bin loss [374] to enable the data distribution-aware optimization, which helped to address the domain variation challenges from different crowd data sources. *Song et al.* [372] studied the counting problem in a different way, where a combination of *Euclidean* loss and *Cross Entropy* loss was used for point location learning, instead of density map regression. Along the same line, *Bayesian* loss was proposed by [261] to provide more reliable supervisions at each annotated point. Alternatively, *Wan et al.* [409] studied the combination of pixel-wise loss and point-wise loss, which investigated the density map representation through an unbalanced optimal transport problem. [407] proposed a novel loss function to address the spatial annotation noise during training, where a weighted MSE term and a pixel-wise correlation term were involved. Recently, [412] proposed a distribution matching loss to

tackle the weakened generalizability of Gaussian smoothed density maps. Moreover, *Wang et al.* [413] treated the counting with density maps as a classification problem, where a Cross-Entropy loss was used to classify each patch into certain intervals.

The aforementioned methods introduced different loss functions to supervise a model, such as point locations, bounding boxes, matching, ranking, classification, *etc.*. However, the mainstream counting methods still rely on pixel-wise supervision with the density map ground truth [112], such as the *L1* or *L2* loss functions. In this work, we propose a Dilated Contrastive Density Loss (L_{DCD}) to improve the pixel-wise loss' receptive field and to increase the regional supervision.

6.3 Methods

6.3.1 Ground Truth Generation

Following [201], given a set of N images $\{I_i\}_{i=1}^{N}$ with corresponding point annotations $\{P_i\}_{i=1}^{N}$, the ground truth of the density map $\{D_i\}_{i=1}^{N}$ is generated by filtering the points with a normalized Gaussian kernel. The total object count number T_i of image I_i can be attained by summing all pixel values of the density map D_i.

The ground truth mask of the crowd segmentation task is generated from the density map ground truth. Given a set of N density maps $\{D_i\}_{i=1}^{N}$, the value for the pixel in the mask $\{B_i\}_{i=1}^{N}$ is set to 1 if its pixel value in the density map is larger than zero, otherwise it is set to 0 .

The ground truth mask used by the density level segmentation task is also generated from the density map. For pixel p in input image i, its density level class $S_{p,i}$ is given as:

$$S_{p,i} = \min_{i=1,..,N} \left(\frac{D_i(p) - \min(D_i)}{\max(D_i) - \min(D_i)} \times L, L \right), \tag{6.1}$$

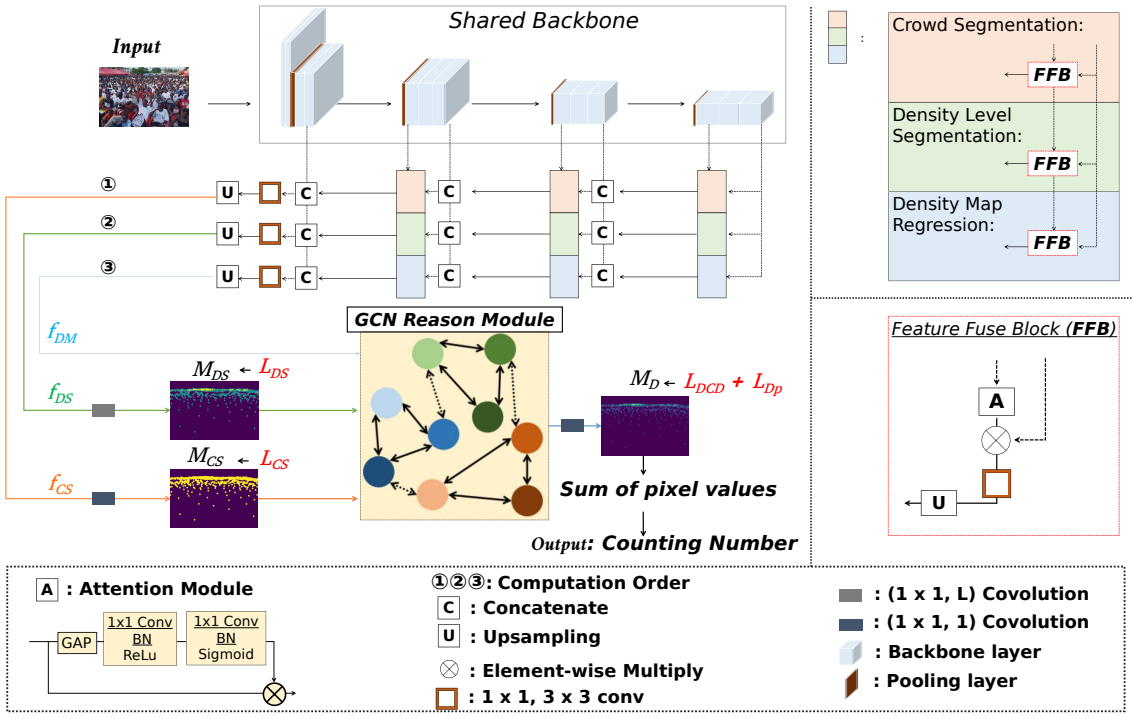


Figure 6.3: Illustration of our proposed network. The adaptively shared backbone network has three outputs of f_{CS}, f_{DS}, f_{DM}, representing crowd segmentation, density level segmentation, and density map regression branches' output feature map, respectively. The order of their involvements indicates that the density map regression branch can benefit from the extra density level and crowd spatial supervision from the other two branches gradually.

where L represents the overall levels of density. Following previous patch-based density level classification methods [167,359], we set L equal to 4 in our work. D_i is the pixel value in the i_{th} density map ground truth. Specifically, given a density map and Eq. 6.1, we can generate the density level map with L levels of object density. In other words, we set all the pixels of the density map into L categories or classes according to their own pixel value. In this way, each pixel is assigned to a semantic label to represent the high-level sparseness or denseness.

6.3.2 Task Adaptive Backbone Network

Intuitively, our motivation is that the backbone network should be able to produce both universal (or generic) and specialised features that are applicable to all tasks and can also be tailored to specific tasks. To this end, instead of using a shared backbone network to extract generalizable features for different tasks, we propose an auxiliary-task based adaptive backbone network to allow the model to extract discriminative features for the auxiliary tasks, thus helping to improve the performance of the main task. Fig. 6.3 shows the detailed structure of the proposed network, which consists of a shared backbone and three attention-based task-adaptive branches. To make a fair comparison with previous auxiliary task-based methods, such as [167, 252, 341, 367], *etc.*, the truncated VGG-16 [365] is used as the backbone network. However, it can be replaced by any other robust network structure; we have reported the counting performance with other powerful network backbones in TABLE. 6.5. The shared backbone adopts the first 13 layers of VGG-16 to extract multi-level features. To exploit the global contextual dependencies, we propose a Feature Fuse Block (*FFB*), which aggregates and fuses the outputs from posterior layers back to the preceding layers hierarchically and iteratively, with up-sampling, concatenation and convolution operations. This provides improvements in extracting the full spectrum of

semantic and spatial information across different stages and resolutions. The up-sampling is performed by using a bilinear interpolation algorithm. The convolution operation aims to reduce and match the corresponding feature map channel size between different stages.

With the aggregating process from high-level features to low-level features, the task-adaptive attention module is applied in three different task branches; details of the attention module are shown in the bottom left of Fig. 6.3. Each attention module consists of a global average pooling (GAP) layer to capture global context through different feature map channels, generating an attention tensor to lead the emphasis of feature learning. Then, two blocks with a convolutional layer followed by a Batch Normalization *(BN)* [160] layer with *ReLu* and sigmoid as the activation functions are added. For the convolutional layer filter, the kernel size is 1×1. Element-wise multiplication is then performed between the outputs of a particular layer of the shared backbone and the task-specific attention module, which filters out the unrelated and redundant features from the backbone with respect to different auxiliary tasks and the main task. Therefore, the shared backbone can learn a generalizable representation, while the attention-based branches can extract task-specific features simultaneously in an end-to-end manner. The ablation study experiments proved that the attention-based adaptive backbone could boost the counting performance.

Apart from the aforementioned network structure component in three attention-based task-adaptive branches, we also introduce a cross-granularity feature fusing operator in a particular order to focus on optimizing the density map regression task. Specifically, the crowd segmentation branch is applied to the shared backbone first to select the corresponding discriminative spatial features. Then, we applied the density level segmentation branch on the shared backbone and crowd segmentation branch, which can enhance the additional contextual density level information into the main task. At last, the main task of the density map regression branch is applied.

6.3.3 Auxiliary Tasks

With three outputs from the task adaptive backbone network, we built two auxiliary tasks and a main task: crowd segmentation, density level segmentation, and density map regression. We detail each of them subsequently.

Crowd Segmentation. We introduce crowd segmentation as one of the auxiliary tasks for two reasons. Firstly, the pixel value of the density map should be zero in areas devoid of people. However, the predicted density map can be inaccurate and noisy when the background is cluttered and complex. The task of crowd segmentation provides a spatial focus to the density map regression procedure by setting the pixel values of non-crowd regions to zero. Secondly, given the standard setup of single density map regression, pixels within a specific range of the point annotations should contribute more to the final counting results; however, most irrelevant pixels dominate the loss [112]. In order to circumvent this constraint, crowd segmentation can provide additional information enhancement in terms of the spatial indicator via a standalone loss function.

Given an input image $I_i \in \mathbb{R}^{3\times H\times W}$, we can get the output of the crowd segmentation branch in the backbone network, $f_{CS} \in \mathbb{R}^{C\times H\times W}$, where H and W represent the height and width of the feature map; C is the channel size. Then, we apply a convolution layer with filter parameters $\theta_{CS} \in \mathbb{R}^{1\times 1\times 1}$, followed by a sigmoid activation function. Through this operation, we can generate a probability map to calculate the crowd and background probability. The single channel crowd segmentation probability map M_{CS} is defined as: $M_{CS} = Sigmoid(\theta_{CS}, f_{CS}) \in \mathbb{R}^{1\times H\times W}$. Fig. 6.4 demonstrates an example of the location map, which is the M_{CS} after using 0.5 as the thresholding, resulting in a binary map. The colors represent different classes, where there is a foreground class and background class. Crowd segmentation focuses on the spatial information, and indicates the geometry-aware supplementary as the auxiliary task.

Density Level Segmentation. Density map regression is a pixel-wise task that focuses on the learning of low-level features but may disregard high-level semantic information, such as the density level information [386]. However, such semantic information is critical in the counting system because the density map's pixel values should rely not solely on their own pixel-wise characteristics but also on regions with varying densities [236]. To address the issues, we perform density level segmentation as another auxiliary task. Compared with previous patch-based density level classification methods [167, 359, 366, 367], our proposed pixel-based density level segmentation can provide pixel level density information and high-level semantic features at the same time. Fig. 6.4 demonstrates an example of the density level map, where colors represent different classes. From class 3 down to class 0, the density level decreases. Density level segmentation focuses on the semantic information, and indicates the density level-aware supplementary as the auxiliary task. Upon the output of the density level segmentation branch of the backbone network $f_{DS} \in \mathbb{R}^{C\times H\times W}$, a convolution layer with filter parameters $\theta_{DS} \in \mathbb{R}^{L\times 1\times 1}$ and a softmax activation function are applied. The prediction of the density level segmentation branch M_{DS} is defined as: $M_{DS} = softmax(\theta_{DS}, f_{DS}) \in \mathbb{R}^{L\times H\times W}$, where L is the number of density levels.

6.3.4 Density Map Regression

Intuitively, the different granularity features of density levels and spatial crowd locations need to be further analysed for fusion into a combined reasoned feature to feed to density map regression branch. To this end, with the predicted crowd segmentation output M_{CS} and density level segmentation output M_{DS} as the auxiliary information granularity, we input them along with the feature map derived from the density map branches $f_{DM} \in \mathbb{R}^{C\times H\times W}$ into the *GCN* reasoning module to understand the relationship among themselves. Subsequently, the output feature map $f_{DM'} \in \mathbb{R}^{C\times H\times W}$ of the *GCN* reason-

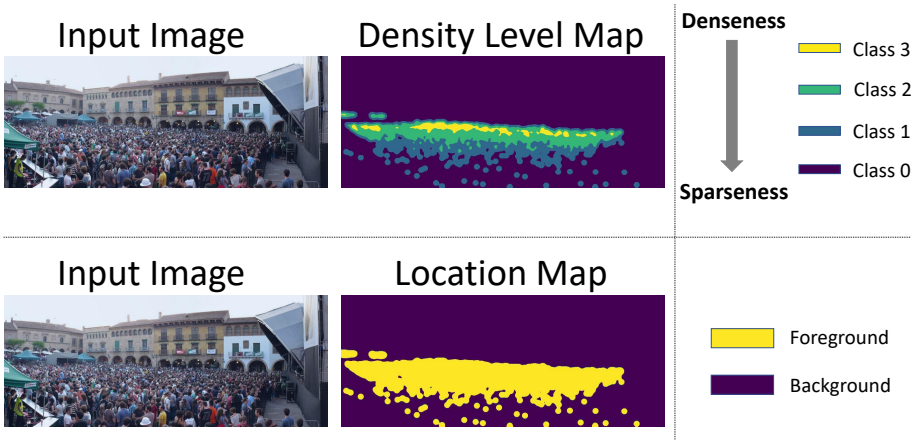


Figure 6.4: Example of the density level map (top) and location map (bottom). For the density level, the colors represent different classes, which corresponds to different density levels. From class 3 down to class 0, the density level decreases from denseness to sparseness. The class 0 represents the background, where there is no objects. As for the location map, the colors represent the different classes, where there is a foreground class and a background class.

ing module is reduced into one-channel through a 1×1 convolution layer with a $ReLU$ activation function.

6.3.5 *GCN* Reasoning Module

Deep feature extraction and fusion have been explored in previous studies, such as discriminant correlation analysis [13, 14], and multi-canonical correlation analysis [10–12], where they adaptively selected and fused CNN features from different layers, such that resulting representations have a high linear correlation. Following the same line, we propose a GCN model to fuse the correlated and supplementary features from auxiliary tasks that contribute to the counting task.

Different granularity representations are utilised for the crowd segmentation and density level segmentation feature domains. Direct fusion (element-wise multiplication or channel-wise concatenation) of the outputs of three task branches may lead to domain

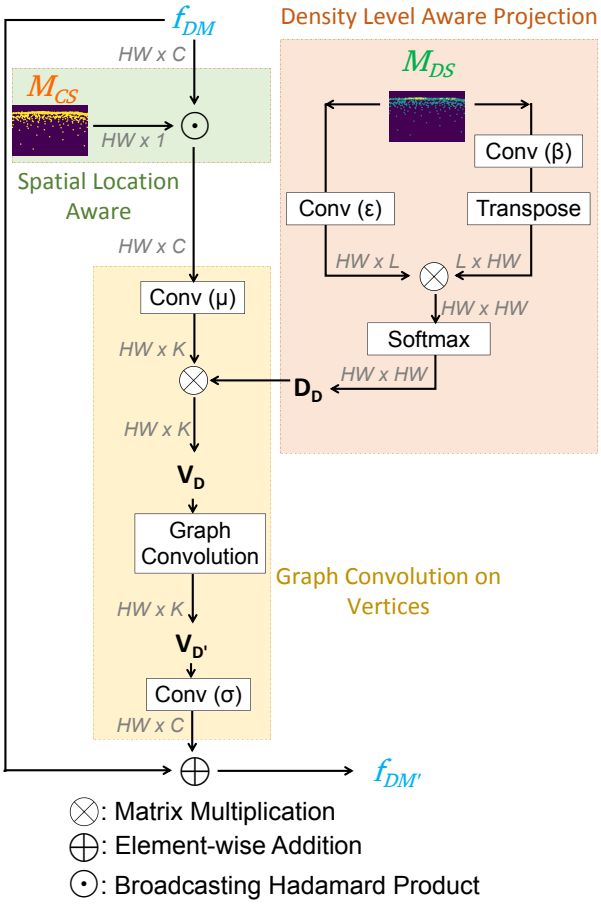


Figure 6.5: Architecture of the proposed *GCN* reasoning module. $f_{DM} \in \mathbb{R}^{C\times H\times W}$ is the feature map of the density map regression branch, $C = 32$ is the channel size; $M_{CS} \in \mathbb{R}^{1\times H\times W}$ is the prediction of the crowd segmentation branch; $M_{DS} \in \mathbb{R}^{L\times H\times W}$ is the prediction of density level segmentation branch, $L = 4$ is the number of density levels; $D_D \in \mathbb{R}^{HW\times HW}$ is the density level dependency matrix; $V_D \in \mathbb{R}^{K\times HW}$ is the constructed vertex features and $V_{D'} \in \mathbb{R}^{K\times HW}$ is the output vertex features after *GCN*, $K = 16$ is the number of vertices. $f_{DM'} \in \mathbb{R}^{C\times H\times W}$ is the output feature map after *GCN* reasoning.

conflicts [252]. Our *GCN* reason model projects a collection of pixels from a spatial-aware density feature map with similar density levels to each graph vertex and exploits a *GCN* to reason about the relations among graph vertices. In other words, our graph is formed with fused-information of spatial locations and density levels from auxiliary tasks via initialising the adjacency matrix and vertices (D_D and V_D shown in Fig. 6.5). The proposed *GCN* reasoning module structure is shown in Fig. 6.5. In detail, there are three primary modules: *Spatial Location Aware* module, *Density Level Aware Projection* module, *Graph Convolution on Vertices* module.

Spatial Location Aware Module. Before projecting the density map feature map f_{DM} into the graph vertices, we directly applied the broadcasting Hadamard Product operation between the crowd segmentation output M_{CS} and the density map regression branch's feature map f_{DM}. There are two underlying reasons for this: (1) M_{CS} is a one-channel crowd segmentation map, with encoded probabilities of the non-crowd regions' pixel values approaching zero and crowd regions' pixel values approaching one; the value of one serves as a filter to zero out the non-crowd region's pixel value of the density map. (2) the broadcasting Hadamard Product can achieve crowd spatial awareness for every channel of f_{DM} through zeroing out the non-crowd region's pixel value. This addresses the challenge of complex scene backgrounds in crowd images.

Density Level Aware Projection Module. As mentioned above, the pixel-wise density level information can help to address the challenges of large variations of density levels in crowd images. However, direct broadcasting Hadamard product between the density map branch's feature map f_{DM} and the density level output M_{DS} may result in domain conflicts [252]. We exploited the nature of *GCN* and projected the density level information into the graph vertices for further reasoning, which benefited the long-range relationship reasoning ability of *GCN* and the multi-granularity information enhancement from density

level. Inspired by the non-local module [429], we encoded the long-range density level dependency among every pixel. Give the feature map M_{DS}, the density level dependency matrix $D_D \in \mathbb{R}^{HW \times HW}$ is defined as:

$$D_D = softmax\Big(\epsilon(M_{DS}) \otimes \beta^{\mathsf{T}}(M_{DS})\Big), \tag{6.2}$$

where Conv β and Conv ϵ are two convolution layers with 1×1 kernel size, respectively. The dependency matrix D_D can be regarded as a pixel-wise attention map, where pixels with similar density levels are assigned larger weights. The dependence matrix might itself reflect the pixel-by-pixel density level dependency. In addition, with Eq. 6.2, we projected the density level map as a precondition to the graph domain via matrix multiplication, which simultaneously improves high-level semantic dependence.

Graph Convolution on Vertices. In this module, we learnt how to reason the region-based relationship in the density map through GCN in graph domain. Formally, the constructed vertices V_D is defined as:

$$V_D = D_D \otimes \mu(f_{DM} \odot M_{CS}), \tag{6.3}$$

where $\otimes$ is matrix multiplication; $\odot$ is the broadcasting Hadamard product. Specifically in Eq. 6.3, we projected the spatial aware feature map of f_{DM} into graph domain with K vertices, and each vertex is represented by an embedding of shape $H \times W$. This is achieved by Conv (μ), which is a 1×1 convolution layer. Furthermore, we projected the dependency matrix D_D to the graph domain through matrix multiplication, resulting in the vertex features $V_D \in \mathbb{R}^{K \times HW}$. The projection aggregated pixels have similar density levels to graph vertices, where each vertex represents a region in the crowd image. With the constructed vertices (V_D), the long-range region-wise relationship is further reasoned

in the graph domain through *GCN*. Formally, the output vertices of our proposed *GCN* ($V_{D'}$) are calculated as:

$$V_{D'} = ReLU\Big((I - A) \otimes V_D \otimes W_D\Big), \tag{6.4}$$

where I is the identity matrix; $A \in \mathbb{R}^{HW \times HW}$ denotes the adjacent matrix that encodes the graph connectivity to learn; $W_D \in \mathbb{R}^{K \times K}$ is the weights of the *GCN*. The adjacent matrix A is randomly initialized but can learn and update the edge weights from vertex features along the training process. The identity matrix I serves as a residual connection that alleviates the optimization difficulties. Specifically, in Eq. 6.4, we reasoned over the region-wise relations by propagating information across vertices with a single layer *GCN*. Specifically, we fed the constructed vertex features V_D into a first-order approximation of spectral graph convolution [182], resulting the output vertex features $V_{D'} \in \mathbb{R}^{K \times HW}$. Based on the learned graph, the information propagated across all vertices leads to the finally reasoned relations between regions. After graph reasoning, a collection of pixels embedded within one vertex share the same context of features modeled by a graph convolution. Then, we re-projected the vertex features in the graph domain to the original pixel grids. Given the reasoned vertices $V_{D'}$, we applied Conv (σ), which is a 1×1 convolution layer. Finally, we summed up the re-projected and the original density feature maps to form the final feature map. The final pixel-wise density feature map $f_{DM'}$ is thus computed as: $f_{DM'} = f_{DM} + \sigma(V_{D'})$. This can be regarded as the residual connection.

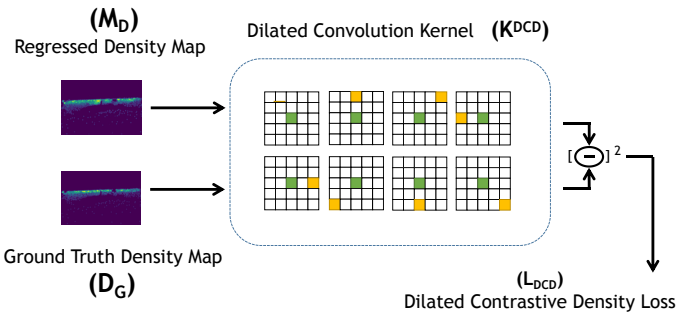


Figure 6.6: Dilated Contrastive Density Loss (L_{DCD}). There are eight dilated contrastive kernels with green, white, yellow blocks representing 1, 0, -1, respectively. The least-square error of two outputs from the regression and ground truth is treated as the final L_{DCD}.

6.3.6 Loss Function

The whole network is end-to-end trainable, which includes four loss functions. The total loss function is defined in Eq. 6.5 as follows:

$$L_{total} = L_{CS} + L_{DS} + \gamma \cdot (L_{Dp} + L_{DCD}), \tag{6.5}$$

where γ is empirically set as 2, which is a hyper-parameter to trade-off between the auxiliary losses and main loss. Please note that extensive experiments have been conducted to determine the weights of the losses for the two auxiliary tasks. We found that there is no significant difference of counting performance with respect to different weight values; thus, we set them both equal to 1 in the loss function. Binary cross-entropy (L_{CS}) is used for the crowd segmentation auxiliary task; categorical cross-entropy (L_{DS}) is used for the density level segmentation auxiliary task; *L2* loss is used for pixel-wise density map regression supervision (L_{Dp}). However, the pixel-wise *L2* loss assumes pixel-wise independence, which results in an over-smooth density map prediction [229] and the underlying bias from unbalanced low- and high-level density distributions of crowd images. To ad-

dress this issue, we propose a Dilated Contrastive Density Loss (L_{DCD}), where we take into account more adjacent pixels for regional density difference. In detail, we applied a single layer convolution on the regressed density map M_D and the ground truth density map D_G. The single layer convolution has eight filters; each filter contains a dilated kernel with a fixed value (*e.g.* 1, 0, and -1). The least-square error of the calculated regional dilated contrastive values from the regressed and ground truth density map is the output of L_{DCD}. To this end, we define L_{DCD} in Eq. 6.6 as below:

$$L_{DCD} = \sum_{i} ||K_i^{DCD} \otimes M_D - K_i^{DCD} \otimes D_G||_2^2, \tag{6.6}$$

where K_i^{DCD} is the i^{th} dilated contrastive convolution kernel, $i \in [1, 8]$. Details of the kernel are shown in Fig. 6.6, where a 3×3 convolution layer with the dilated rate of 2 is applied; this gives a larger receptive field as 5×5. The kernel value is empirically set as 0 ,-1, and 1 because we do not find any significant difference regrading different kernel values. On the other hand, the kernel value is designed to achieve a contrastive learning purpose to include regional relationships among pixels instead of single pixel-wise L2 or L1 loss. We performed extensive experiments to evaluate the effectiveness of the proposed L_{DCD} loss; quantitative results in the *Ablation Study* (Section 6.5.4) demonstrates that the proposed L_{DCD} loss can improve the counting accuracy not only for our model but also for previous single *L2* loss-based methods.

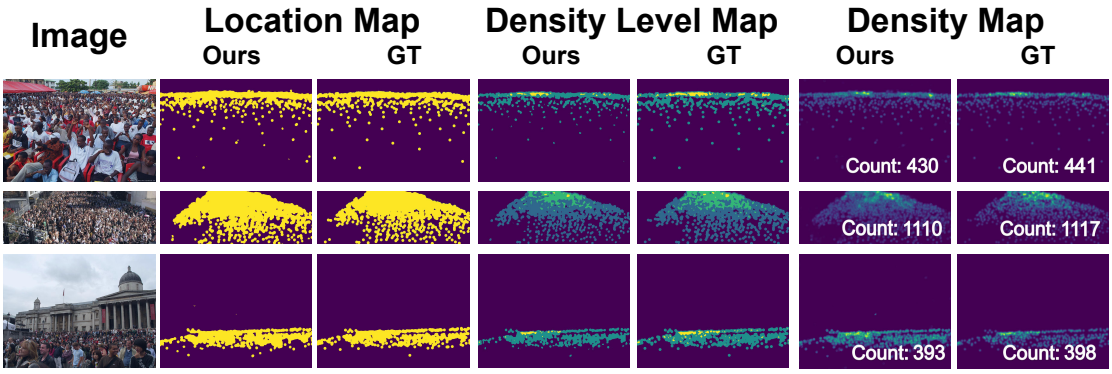


Figure 6.7: Qualitative results of the density, crowd location and density level map in *SHA* test dataset. Our model can produce accurate density maps compared with the ground truth (GT), along with accurate auxiliary crowd segmentation and density level segmentation results.

6.4 Experiments

6.4.1 Datasets

ShanghaiTech [488] consists of 1,198 images, containing a total amount of 330,165 people with head centre point annotations. This dataset has been divided into two parts: ***SHA*** includes 482 images, in which crowds are mostly dense (33 to 3139 people); ***SHB*** includes 716 images, where crowds are sparser (9 to 578 people). Each part is divided into training and testing subsets as specified in [488]. ***UCF-QNRF*** [156] is a large crowd dataset, consisting of 1,535 images with around 1.25 million annotations in total. The number of people in these images varies largely with a wide range spanning from 49 to 12,865. As indicated by [156], for training, 1,201 images are used, the remaining 334 images form the test set. ***JHU-Crowd++*** [369] is a recent challenging large-scale dataset that contains 4,372 images with 1.51 million annotations. The dataset includes several challenging scenes such as weather-based degradation and illumination variations *etc.*. This dataset is divided into 2,272 images for training, 500 images for validation, and 1,600 images for testing.

NWPU-Crowd [419] is currently the largest public crowd counting dataset, containing 5,109 images with over 2.13 million annotations. The dataset includes 3,109 training images, 500 validation images and 1,500 test images. Moreover, inspired by the potential of crowd counting, we conducted experiments on commonly used vehicle counting dataset: ***Trancos*** [127] with 403 images for training, 420 images for validation and 421 images for testing. These experiments further demonstrate our model's robustness and applicability for different real-world applications.

Note that, for ShanghaiTech (*SHA*, *SHB*), *UCF-QNRF*, and *DCC* dataset, we use 10% of the given training images as the validation dataset.

6.4.2 Implementation Details

To augment the dataset, we randomly cropped the input images, density maps, crowd segmentation masks, and density level segmentation masks with fixed size 128×128 at a random location, then randomly flipped the image patches horizontally with a probability of 0.3. We trained our model with 400 epochs for all experiments, with a starting learning rate of $1e-4$ and a cosine decay schedule [248]. The batch size is set to 96. Five-fold cross-validation is used for fair comparison and hyper-parameter tuning is applied in all settings. We implemented the proposed method with *PyTorch 1.7, CUDA 10.2* using *Python 3.6*. All the training processes are performed on a server with four TESLA V100, and all the test experiments are conducted on a local workstation with *Intel(R) Xeon(R) W-2104 CPU* and *Geforce RTX 2080Ti GPU*. Our model takes average 19.5 hours to train on *JHU-Crowd++* [369] and *NWPU-Crowd* [419] datasets and average 8.5 hours to train on *ShanghaiTech* [488], *UCF-QNRF* [156] and *Trancos* [127]. Our implementation code is publicly available at: https://github.com/smallmax00/Counting_With_Adaptive_Auxiliary.

6.4.3 Evaluation Metrics

To evaluate the counting performance, we adopted Mean Absolute Error (MAE) and Root Mean Squared Error ($RMSE$). Since Mean Absolute Error (MAE) and Root Mean Square Error ($RMSE$) cannot measure the counted objects' locations, Grid Average Mean absolute Error ($GAME$) is used to indicate counting accuracy over local regions. $GAME$ is defined in Eq. 6.7 as below:

$$GAME(L) = \frac{1}{N}\sum_{n=1}^{N}(\sum_{l=1}^{4^L}|y_n^l - \hat{y}_n^l|), \tag{6.7}$$

where N is the total number of images, y_n^l and $\hat{y}_n^l$ are the ground truth and estimated counts in the local region l of n^{th} image. 4^L denotes the number of non-overlapping regions which cover the full image. When L equals to 0, $GAME$ is equivalent to MAE.

6.5 Results

6.5.1 Counting Results

In this section, we present our experimental results on the crowd and vehicle counting tasks in comparison to other **auxiliary-task based** state-of-the-art crowd counting methods. These experiments further demonstrate our model's robustness and applicability in multiple domain datasets. In the Discussion (Section 6.5.5), we show that our model could indicate some mislabeled or incorrectly labeled point annotations from the ground truth of the test dataset. This highlights our approach's generalizability and the potential issue of imperfect ground truth in object counting datasets.

Crowd Counting Results. We performed experiments to validate our model's performance in five challenging crowd counting datasets. Fig. 6.7 shows qualitative results;

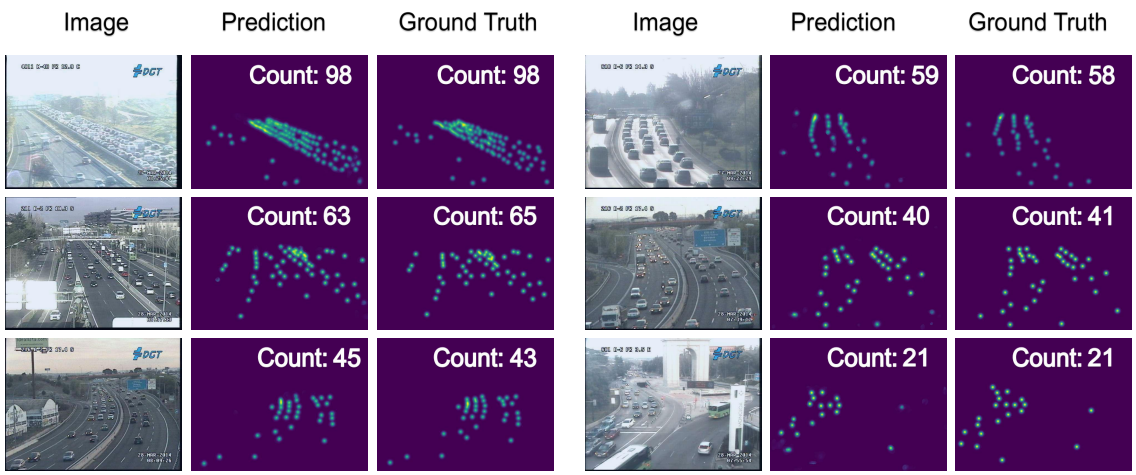


Figure 6.8: Qualitative results on the Trancos dataset. The density map ground truth and our predictions are shown, with counting number presented in the figure. Our model adapts well with scale variations, where the scale of the vehicles varies from the distance between the camera and vehicle locations. Specifically, the vehicles that are far from the camera only contain a few pixels in the image, while the near-camera vehicles have more pixels. The scale of such pixel occupation changes can be well handled by our methods and the predicted density maps can clearly show the location correspondence.

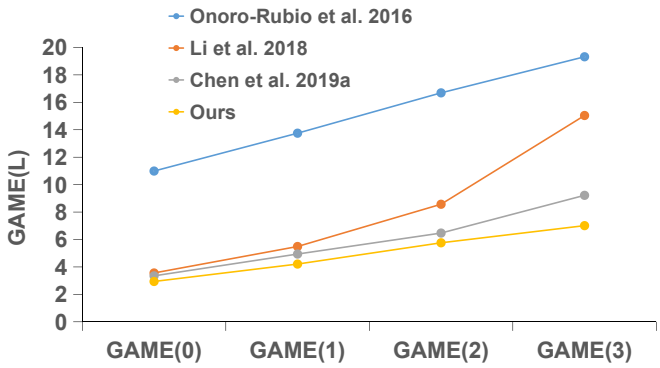


Figure 6.9: Comparison of $GAME$ performance on the *Trancos* dataset among the proposed approach and the state-of-the-arts, such as *Onoro-Rubio et al.* [306], *Li et al.* [219], *Chen et al.* [65]. Note that, a small range of increase among different $GAME$ values indicates that our method counts and localizes overlapping vehicles more accurately.

specifically, we presented the predictions from auxiliary task branches (crowd segmentation and density level segmentation masks) to demonstrate our model's cohesion, along with the spatial location and density level variation's contribution of auxiliary branches. To make a fair comparison, we only compared our model with previous auxiliary task learning-based counting methods. TABLE. 6.3.6 shows that our method outperforms other methods in terms of MAE on all five datasets. In particular, our model outperforms the patch-based density level classification based method *HA-CCN* [366] by 14.7% via average MAE. Notably, the *JHU-Crowd++* dataset [369] and *NWPU-Crowd* dataset [419] are recent publicly available datasets, which are more challenging due to large variations in scale, occlusion, and complex weather scenes. Specifically, *NWPU-Crowd* is the current largest crowd counting benchmark [3]. To the best of our knowledge, we achieved the greatest performance among other auxiliary task-based methods. Except the auxiliary-based methods shown in TABLE. 6.3.6, our method gains a superior reduction than single-task learning-based methods as well, for example, scale variation was able to enhance CACC (100.1

[3] https://www.crowdbenchmark.com/nwpucrowd.html

Table 6.2: Results on vehicle (*Trancos*) counting dataset. Our model achieves superior performance to the previous state-of-the-art methods.

Methods	Trancos	
	MAE	*RMSE*
PPPD [269]	9.7	-
CSRNet [219]	3.5	5.1
BL-Crowd [261]	2.9	6.7
MD-Crowd [412]	3.1	6.6
Auto-Scale [453]	2.9	6.1
SUANet-Fully [284]	4.9	6.9
SASNet [373]	2.9	4.7
Gau-SANet [76]	2.5	**2.8**
STNet [418]	3.8	5.0
ASCC [166]	3.8	4.9
DM-Count [412]	3.9	5.2
P2PNet [372]	3.8	4.9
WSNet [151]	4.3	5.8
Ours	**2.3**	4.8

MAE) [236] by 18.3% and the dilated kernel-based method CSR-Net (85.9 *MAE*) [219] by 4.8% via *MAE*.

Vehicle Counting Results. We conducted experiments on vehicle (*Trancos* [127]) counting datasets to show our model's broad applicability and robustness. Fig. 6.8 shows the qualitative results, and TABLE. 6.2 shows the quantitative results compared with the previous state-of-the-art methods. Due to the different scenes in the vehicle counting dataset, such as less occlusion, no scale variation, no complex background *etc.*, the contribution of some components of our model will be lessened because we designed our model especially for crowd counting tasks; still, our model achieves superior performance when compared with previous methods. Specifically, our model outperformed the distribution matching supervised methods *BL-Crowd* [261], *MD-Crowd* [412], *P2PNet* [372] and *DM-Count* [412]

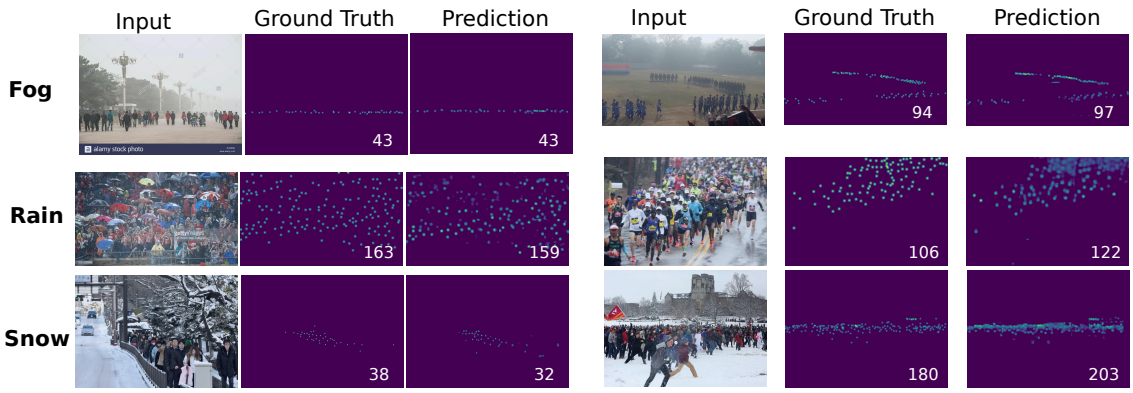


Figure 6.10: Qualitative results on different weather conditions of the *JHU-Crowd++* dataset. The density map ground truth and our predictions are shown, with the counting number presented in the figure. In total, three conditions, fog, rain, and snow, are demonstrated in the respective rows of the figure. Our model can handle severe weather degradation well and indicates precise crowd locations.

by 20.7, 25.8, 39.5 and 41.0 % of MAE; outperformed the auxiliary task assisted methods *Auto-Scale* [453], *SASNet* [373], *STNet* [418] and *ASCC* [166] by 20.7, 20.7, 39.5, and 39.5 % of MAE. Notably, *WSNet* [151] is specially designed for traffic density estimation and vehicle counting, where an attention-based Transformer [399] is used to extract the local-global consistent features. This is because the traffic scenario can be easily affected by weather and scale changes, which results in weakened semantic and spatial content of the captured images. Our proposed graph-based multi-granularity information fusion paradigm had a similar intuition, to enhance the relevant semantic and spatial information. Our model outperformed *WSNet* [151] by 46.5 % by MAE in Trancos test dataset. Furthermore, we present local comparison performance through the $GAME$ metric to indicate the model's ability to recognize the objects' locations. Fig. 6.9 shows the comparison results in terms of the $GAME$ on the Trancos dataset. As illustrated, our method localizes and counts overlapping vehicles more accurately.

Results on Weather Changes Among the seven datasets used in this work, *JHU-*

Crowd++ [369] provided the weather condition-based labels. For example, the test dataset (a total of 1600 images) contained 168 images weather labels; for example, 49 images are labeled as 'rain'; 78 images are labeled as 'snow'; 64 images are labeled as 'fog'. In this section, we provide the quantitative and qualitative counting results on different weather conditions. Following *JHU-Crowd++* [369] benchmark's setting, we report the counting performance on the test images with weather labels. Specifically in TABLE. 6.3, our method achieved 110.2 MAE and 598.2 RMSE, which outperformed previous state-of-the-art methods *LSC-CNN* [341], and *MBTTBF* [368] by 38.1 and 20.5 % MAE. Benefiting from the proposed auxiliary task and the graph-based multi-granularity feature fusion mechanism, our model can extract the spatial and semantic features from the input image, especially when weather degradation causes a weakened image quality. Fig. 6.10 shows the qualitative results of our model under different weather conditions. Our model can handle the severe weather degradation well, which is critical in the intelligent transportation system because weather can easily affect traffic scenarios.

6.5.2 Auxiliary Task Results

In this section, we report the performance of the two auxiliary tasks. The commonly used segmentation metric Intersection over Union (IoU) is used to evaluate the auxiliary tasks' performance. In detail, we achieved average 88.7 % IoU for the crowd segmentation task and 81.0 % IoU for the density level segmentation task on the five crowd counting datasets. Fig. 6.7 shows examples of those tasks' predictions from our model.

6.5.3 Computational Efficiency

Table.6.4 presents the number of parameters in millions (M), floating-point operations ($FLOPs$) and inference time in millisecond (ms) of the compared models. Our model

Table 6.3: Results on *JHU-Crowd++* [369] counting dataset under weather setting. We follow the *JHU-Crowd++* [369] benchmark's setting and report the counting performance. Our model achieves superior performance to the previous state-of-the-art methods.

Methods	JHU-Weather	
	MAE	*RMSE*
CSRNet [219]	141.4	640.1
SA-Net [50]	154.2	685.7
CACC [236]	155.4	617.0
DSSI-Net [233]	229.1	760.3
MBTTBF [368]	138.7	631.6
LSC-CNN [341]	178.0	744.3
JHU-Crowd++ [369]	138.6	654.0
SFCN [420]	122.8	606.3
BL-Crowd [261]	140.1	675.7
Ours	**110.2**	**598.2**

adopts *VGG-16* [365] as the backbone, which leads to a relatively smaller model size of 18.8 M parameters, compared to other models, such as *LSC-CNN [341]* (35.1 M), *ASCC* [166] (30.4 M), and *SASNet* [373] (38.9 M). On the other hand, our model is computationally effective, only requiring 8.5 *FLOPs*. This is comparable to other light-weight models such as *DM-Count* [412], *SUANet-Fully* [284], and *BL-Crowd* [261]. Due to the auxiliary task-based nature, our model required a relatively longer inference time, such as 8.8 *ms* per image. However, our method can still be used for a real-time counting application (*inference speed* $>$ *24 frame per second*).

6.5.4 Ablation Study

We investigated the effect of each component in our proposed model. All ablation experiments were performed with the same settings detailed in the Implementation Details (Section 6.4.2).

Table 6.4: Computational efficiency. The number of parameters in millions (M), floating-point operations ($FLOPs$) and inference time in millisecond (ms) of different counting methods on a fixed size of 128 $\times$ 128 input image.

Methods	*Params* (M)	*FLOPs* (G)	*Inference Time* (ms)
DM-Count [412]	21.5	6.7	1.9
SUANet-Fully [284]	15.9	6.5	5.3
LSC-CNN [341]	35.1	25.4	4.6
BL-Crowd [261]	21.5	6.7	1.9
ASCC [166]	30.4	10.2	3.2
SASNet [373]	38.9	14.6	7.8
Ours	18.8	8.5	8.8

Table 6.5: Results of using different backbone networks on five crowd counting datasets.

Methods	*SHA*		*SHB*		*QNRF*		*JHU-Crowd++*		*NWPU-Crowd*	
	MAE	*RMSE*	*MAE*	*RMSE*	*MAE*	*RMSE*	*MAE*	*RMSE*	*MAE*	*RMSE*
VGG-16 [365]	**57.0**	98.6	7.1	12.3	**85.3**	129.4	**66.6**	254.9	**76.4**	**327.4**
VGG-19 [365]	59.7	99.8	8.4	13.2	87.8	144.0	73.7	320.1	79.9	360.0
ResNet-50 [143]	57.8	**96.6**	**7.0**	**11.7**	85.5	**128.7**	77.9	318.1	79.3	344.4
ResNet-101 [143]	61.1	100.8	9.1	14.5	93.3	147.9	69.7	**253.3**	81.4	361.5

Ablation on Different Network Backbones We evaluated the effectiveness of different backbone networks on the five crowd counting datasets. The counting performance is shown in TABLE. 6.5 with several different backbone networks. In general, *VGG*-based backbone networks achieved comparable counting performance, compared with *ResNet*-based backbone networks in relatively large-scale datasets, such as *QNRF*, *JHU-Crowd++* and *NWPU-Crowd*. While, *ResNet*-based backbones work better on small-scale counting datasets, such as *SHA* and *SHB*. We report our model's performance with *VGG*-16 backbone network in TABLE. 6.3.6 for a fair comparison with previous methods.

Ablation on Auxiliary Tasks and Model Components. In this section, we evaluate the effectiveness of the auxiliary tasks, adaptively shared backbone network, and *GCN*-enabled reasoning module. Please note that, in order to eliminate the performance improvement from a bigger model, we add feed-forward *CNN* blocks containing (3 $\times$ 3 convolution with Batch Normalization) into other ablation study models in TABLE. 6.6

to maintain a similar model size as ours (18.8 million parameters). Firstly, we compared the single task density map regression network, in which we removed the GCN reasoning module, the auxiliary learning branches, and the adaptively shared backbone branches, to form a single column network structure (*Single Column*). Then we added two auxiliary branches separately and simultaneously after the single shared backbone's output to form an auxiliary learning mechanism (*w/ Crowd Seg*, *w/ Density Seg*, *w/ Both Auxiliary*). To further improve the performance, we designed and added an adaptive backbone network to enable the task-shared and task-specific features to be learned simultaneously (*w/ Adaptive Crowd Seg*, *w/ Adaptive Density Seg*, *w/ Both Adaptive Auxiliary*). Furthermore, we evaluated the proposed *GCN* reasoning module's effectiveness, which can propagate region-based density level information across the image (*Ours*). The effect of each structural component is presented in Fig. 6.6. As illustrated, the proposed auxiliary task learning mechanism (*w/ Both Auxiliary*) is reduced by 14.3% over the single-task learning method (*Single Column*) via average *MAE* on two datasets, the task adaptive backbone (*w/ Both Adaptive Auxiliary*) reduces 6.8% over the single shared backbone (*w/ Both Auxiliary*), and the *GCN* reasoning module further reduces 6.7%. Qualitative comparison results of different modules' effectiveness in terms of predicted density maps are shown in the Fig. 6.11, where the crowd segmentation auxiliary (*w/ Adaptive Crowd Seg*) can help the model to focus on the features in the region of interest and filter out the background (first and second rows). On the other hand, the density level segmentation auxiliary (*w/ Adaptive Density Seg*) can help to estimate more accurate density levels across the whole density map (second and third rows). We highlighted the different areas among those ablated models' density map predictions with red bounding boxes for better visualization and comparison.

Moreover, in TABLE. 6.7, we further indirectly evaluate the auxiliary tasks' effec-

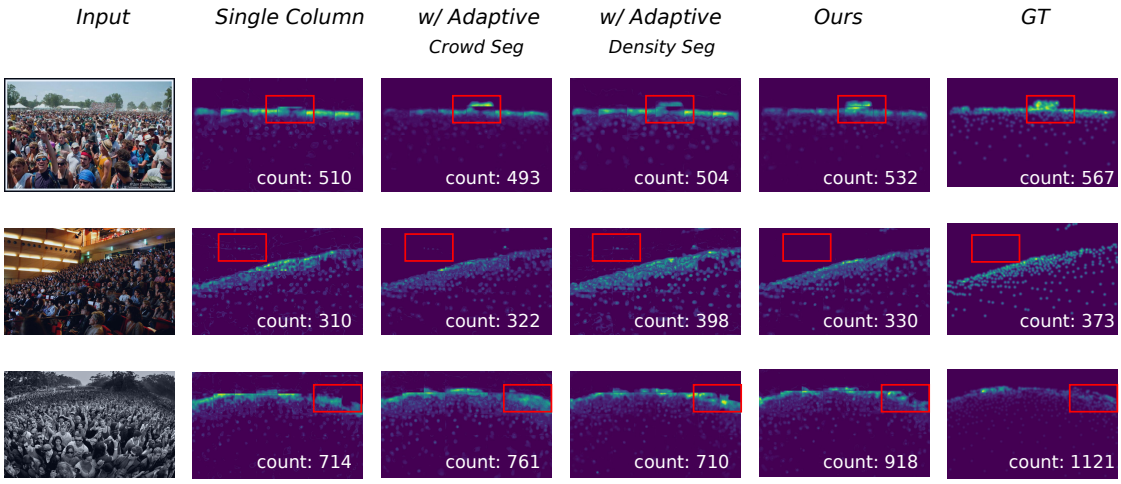


Figure 6.11: The qualitative results of ablation studies about auxiliary tasks. The red bounding boxes are used for better visualization and comparison. *Ours* and *w/ Adaptive Crowd Seg* can know the crowd's spatial regions (first and third rows), and filter out the background noise (second row). On the other hand, *Ours* and *w/ Adaptive Density Seg* can estimate more accurate density levels across the whole density maps (second and third rows).

tiveness in this work. Specifically, for other ablation study models except for *Ours*, we maintained the same network structure as *Ours* to keep the same model size (18.8 million parameters) but switched off the two auxiliary tasks' loss functions. In TABLE. 6.7, it proves that the supervision from multi-granularity information of auxiliary tasks contributes to the final counting performance in this work. Without L_{CS} and L_{DS} losses, the counting error increases by an average of 21.75 % on the *SHA* and the *JHU-Crowd++* datasets via *MAE*.

Ablation on Graph Reasoning Module. In this section, we evaluate the effectiveness of the proposed graph reasoning module. We specially designed our graph reasoning module to incorporate the auxiliary tasks and for fusing information into the adjacency matrix to form the information-fused graph. So for the ablation study, we had to only apply other *GCN* on the density map. Firstly, we employed the classic graph convolution [182]

Table 6.6: Ablation study results on network structure components. Each component of our network contributes to the final prediction.

Methds	*SHA*		*JHU-Crowd++*	
	MAE	*RMSE*	*MAE*	*RMSE*
Single Column	71.3	122.3	99.3	391.0
w/ Crowd Seg	67.4	117.0	81.6	343.6
w/ Density Seg	68.1	119.9	86.1	360.0
w/ Both Auxiliary	65.2	115.2	77.3	311.7
w/ Adaptive Crowd Seg	61.3	104.6	75.7	300.9
w/ Adaptive Density Seg	63.8	108.1	76.9	307.8
w/ Both Adaptive Auxiliary	60.8	100.3	71.9	278.9
Ours	**57.0**	**98.6**	**66.6**	**254.9**

Table 6.7: Ablation study results on auxiliary tasks. Maintaining the same model structure (model size) and turning off auxiliary tasks' loss functions can implicitly prove that the auxiliary tasks contribute to the final counting.

Methds	*SHA*		*JHU-Crowd++*	
	MAE	*RMSE*	*MAE*	*RMSE*
w/o L_{CS}	64.4	107.7	78.7	310.5
w/o L_{DS}	62.0	104.8	74.9	302.2
w/o L_{CS} and L_{DS}	67.1	115.2	93.0	377.5
Ours	**57.0**	**98.6**	**66.6**	**254.9**

Table 6.8: Ablation study results on graph reasoning modules. Only our proposed graph reasoning module can efficiently utilize the auxiliary information from other tasks to complement the density map regression task.

Methds	*SHA*		*JHU-Crowd++*	
	MAE	*RMSE*	*MAE*	*RMSE*
classic GCN	67.1	109.0	79.2	308.7
SGR [223]	60.3	101.0	73.1	301.0
DualGCN [483]	63.8	105.7	80.8	307.3
GloRe [70]	61.0	105.4	71.3	317.7
Ours	**57.0**	**98.6**	**66.6**	**254.9**

Table 6.9: Ablation study results on the dilated rate of the proposed loss function L_{DCD}. When the dilated rate is 2 and the corresponding receptive field is 5, our model can achieve the best counting performance on the *SHA* and *JHU-Crowd++* datasets.

Dilated Rate	*SHA*		*JHU-Crowd++*	
	MAE	*RMSE*	*MAE*	*RMSE*
1	60.1	103.5	70.1	299.0
3	58.7	101.7	68.7	288.4
4	59.2	101.3	68.0	287.6
2 (Ours)	**57.0**	**98.6**	**66.6**	**254.9**

Table 6.10: Ablation study results (*MAE*) on our combined loss (contrastive and *L2* loss), compared with single *L2* loss (*base*). Moreover, we applied the combined loss function to optimize previous single *L2* loss based methods to demonstrate that the counting performance can be improved with the help of regional density difference-based loss function L_{DCD}).

Methods	*SHA*		*JHU-Crowd++*	
	Base	*w/ contrastive*	*Base*	*w/ contrastive*
MCNN [488]	110.2	**108.1**	188.9	**168.3**
CSRNet [219]	68.2	**65.9**	85.9	**84.1**
CACC [236]	62.3	**60.8**	100.1	**97.9**
Ours	59.5	**57.0**	70.8	**66.6**

to reason the correlations between regions in density feature maps (f_{DM}). Additionally, we adopted potent graph convolution operations to show the superiority of our proposed *Graph Reasoning Module*. In detail, we applied the *SGR* [223], *DualGCN* [483], and *GloRe* module [70] respectively, where the *SGR* module exploited a knowledge graph mechanism; *DualGCN* explored the coordinate space and feature space graph convolution; and *GloRe* utilized a projection and re-projection mechanism to reason the semantics between different regions. These methods achieved state-of-the-art performance on different computer vision tasks, however, they can only process single task rather than using auxiliary information. Tab. 6.8 shows that our model achieves more accurate and reliable results than [182] and outperforms the *SGR*, *DualGCN*, and *GloRe* by 7.2 %, 20.0 % and 6.6 % in terms of mean *MAE* on the two test datasets.

Ablation on Loss Function. We performed experiments to evaluate the receptive field through different dilated rates in the proposed dilated contrastive density loss function L_{DCD}. In detail, we changed the dilated rate of the 3×3 convolution layer into 1, 2, 3, 4, which resulted in the receptive field of the L_{DCD} being like 3, 5, 7, 9. TABLE. 6.9 shows the comparison results; when the dilated rate is 2, our model achieves the best performance on *SHA* and *JHU-Crowd++* datasets.

Furthermore in TABLE. 6.10, we conducted experiments to evaluate the effectiveness of the proposed dilated contrastive loss function, in which we removed the L_{DCD} and kept the rest of the network constant with the same trade-off hyper-parameters (*Base*). Furthermore, we applied the proposed combined loss function (*w/ contrastive*) into previous single *L2*-based methods [219, 236, 488]. We re-implemented their network with their open-source code and used the same experimental setting as our method. TABLE. 6.10 shows the comparison results of our proposed combined loss function; as illustrated, with regional density difference supervision of L_{DCD}, our model attains a 3.5% reduction compared

with single *L2* loss function via average *MAE* on two datasets. Our proposed L_{DCD} also helps to reduce the original *MCNN* [488] by 6.4%, the *CSRNet* [219] by 2.7%, and the *CACC* [236] by 2.3% over average *MAE* on two datasets. Please note that we did not compare with other loss functions that were proposed in the recent crowd counting models [261, 372, 407, 409, 412, 413]. Those methods are not pure density map regression-based methods, thus it is unfair to compare.

6.5.5 Discussion: Comparison with Ground Truth

Underlying labeling errors (noisy ground truth) exist in most datasets due to human annotator error. However, a robust model can omit noisy ground truths during training and produce a more accurate prediction. This section showed that our model could indicate some mislabeled or incorrectly labeled point annotations of the ground truth in the test dataset. This highlights the generalizability of our approach and the potential issue of the imperfect ground truth in object counting applications. Fig. 6.2 shows a wrongly labeled point annotation (top left) case of the crowd counting test dataset, and the other cases are mislabeled point annotation of vehicle counting test dataset. We highlighted the incorrectly labeled or mislabeled area with red bounding boxes for better visualization and comparison.

6.5.6 Limitation and Future Work

In this work, we presented an object counting framework assisted by auxiliary multi-granularity information, achieving cutting-edge counting performance in seven large-scale counting datasets. This significantly contributes to transportation systems, including many applications such as security alerts, public space design, *etc.*. However, one limitation of our method is that the complexity of inference is increased due to the enlarged num-

ber of optimized tasks. This is a typical issue of auxiliary-task based counting methods [2, 74, 167, 199, 232, 237, 238, 373, 408, 464, 485], which has been discussed before. However, our method only required 8.8 milliseconds per image, which is comparable to other single-task-based methods (please refer to TABLE. 6.4). In other words, our method can also be used for a real-time counting application (*inference speed* $>$ *24 frame per second*). The trade-off between accuracy and complexity can be determined when applied to a real-world task.

A future extension of our work could be multiple objects tracking (MOT), such as vehicles or crowd tracking. Most of the MOT approaches [178, 298, 442] follow the classic paradigm of tracking-by-detection, where object trajectories are obtained by associating per-frame outputs of object detectors. Recently, a new prediction scheme [330, 437] is gaining attention that uses a tracking-by-counting mechanism. Specifically, using the crowd density maps, the detection, counting, and tracking of multiple targets as a network flow program is achieved. In the future, our model could be integrated into such learning pipelines to tackle MOT with dense crowds or vehicles.

6.6 Conclusion

We proposed an auxiliary-task-based object counting methodology via a graph-based multi-granularity information fusion paradigm. The proposed task-adaptive backbone enabled the task-shared and task-specific features to be learned simultaneously. We have demonstrated its potential in maintaining state-of-the-art performance upon seven challenging benchmarks. Our approach is anticipated to be widely applicable in the real world.

Chapter 7

Researching Explicit Graph Representations in Medical Image Segmentation

I study the geometry structure via the explicit graph representation learning in this section. Specifically, I applied the proposed methods on the task of biomedical image segmentation but with a novel paradigm. Specifically, this section proposes a straightforward, intuitive deep learning approach for (biomedical) image segmentation tasks. Different from the existing dense pixel classification methods, I develop a novel multi-level aggregation network to directly regress the coordinates of the boundary of instances in an end-to-end manner. The network seamlessly combines standard convolution neural network (CNN) with Attention Refinement Module (ARM) and Graph Convolution Network (GCN). By iteratively and hierarchically fusing the features across different layers of the CNN, our approach gains sufficient semantic information from the input image and pays special attention to the local boundaries with the help of ARM and GCN. In particular, thanks to the proposed aggre-

gation GCN, our network benefits from direct feature learning of the instances' boundary locations and the spatial information propagation across the image. Experiments on several challenging datasets demonstrate that our method achieves comparable results with state-of-the-art approaches but requires less inference time on the segmentation of fetal head in ultrasound images and of optic disc and optic cup in color fundus images.

7.1 Introduction

The accurate assessment of anatomic structures in biomedical images plays an important role in the management of many medical conditions or diseases. For instance, fetal head (FH) circumference in ultrasound images is a critical indicator for prenatal diagnosis and can be used to estimate the gestational age and to monitor the growth of the fetus [398]. Similarly, the size of the optic disc (OD) and optic cup (OC) in color fundus images is of great importance for the diagnosis of glaucoma, an irreversible eye disease [307]. Manual annotation of this kind of structures by delineating their boundaries in clinics is unrealistic as it is costly, time consuming, labor intensive, and subject to human experience and errors. Automatic segmentation of biomedical images is believed to be able to help improve the efficiency of workflow in clinical scenarios. Inspired by the way clinicians annotate images, I propose an aggregated network to solve the segmentation tasks through directly regressing the locations of objects' boundaries, and demonstrate the effectiveness of the network in the segmentation of FH in ultrasound and OD & OC in color fundus images, respectively.

The biomedical image semantic segmentation task remains a challenging problem in the field of computer vision. The commonly-used deep learning-based semantic segmentation methods [61,141,376] (top row of Fig. 7.1) classify each pixel of an image into a category or class. These methods benefit from Convolution Neural Networks (CNN)'s excellent ability to extract high-level semantic features. Being a part of the understanding of scenes or global

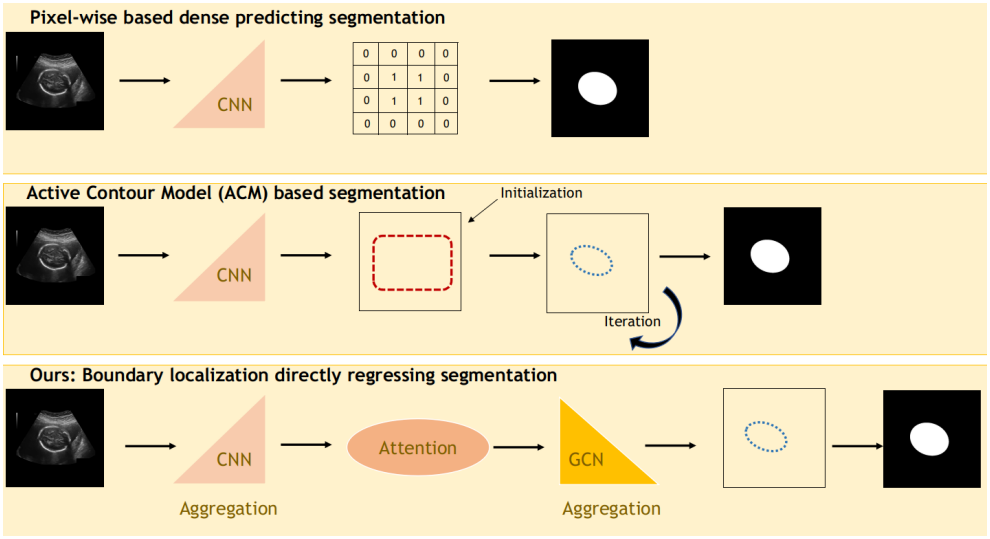


Figure 7.1: Three different segmentation paradigms by deep learning. Top row: pixel-wise based methods [61,109,141] that classify each pixel into objects or background. Middle row: active contour based methods [73, 266] that need iterative optimization in action to find the final contours. Bottom row: our proposed method that directly regresses the locations of object boundaries by information aggregation through CNN and GCN, enhanced by an attention module.

contexts, these methods need to learn the object location, object boundary, and object category from the high-level semantic information and local location information [247]. However, they suffer from the loss of local location information at the pixel-level [69], because a large receptive field corresponds to a small feature map, and this dilemma has increased the difficulties of dense prediction tasks. In order to solve this problem, approaches in [59, 492] either maintain the resolution of the input image with dilated convolution, or capture sufficient receptive fields with pyramid pooling modules. The insights behind these methods indicate that the spatial information and the receptive field are both important to achieving high accuracy. However, it is hard to meet these two requirements simultaneously with CNN [468]. In particular, it is often challenging to maintain enough spatial information of the input image.

To address the aforementioned challenges, I follow a straightforward and intuitive methodology that human operators take to segment objects and regard segmentation as a regression task. Compared with the preserving abstraction of spatial details [59, 492], I use a combination of CNN, ARM, and GCN to directly regress the boundary locations of the instances in the Euclidean space. Our method is different from the recent polygon-based active contour models (ACM) methods [73, 132, 266] (middle row of Fig. 7.1), which need to initialize the boundaries and iteratively find the final object boundaries for a new image. On the contrary, I directly supervise the model to learn the precise location of boundaries and produce the boundaries without iteration during inference. Compared with the pixel-wise based methods, our method needs to learn and extract more spatial information to regress the location directly. To address this issue, the local spatial information propagation nature of GCN is exploited. GCN has recently been applied to many low-level tasks, such as scene understanding [217], semantic segmentation [61], and pose estimation [494], because GCN can propagate the information through neighbor nodes (short range) and hence allow the model to learn local spatial correlation structure.

I propose an aggregated GCN decoder with graph vertices sampling from sparse to dense, which contributes to globally propagate the spatial relationship information across the whole image. This will provide greater representational power and more sufficient information propagation than previous segmentation methods based on Conditional Random Fields or Markov Random Fields [15, 245]. Thus, I can directly regress explicit boundary location with the Euclidean space coordinate representation. This strategy addresses the concerns of the recent works [448, 455], which share the similar idea but convert the Euclidean space representation into polar representation, and regressing the low-level distance between the center point and boundary points. They found that CNN cannot regress the Euclidean space coordinate representation of the boundary well as some more noise may

be added, and the CNN may not maintain enough spatial information [448,455]. Our proposed aggregation GCN can handle this issue well, and our experiment results prove that. Besides, those methods' performance may suffer from the low-quality of center point, so, Xie *et al.* [448] utilized center sample methods to classify and selected high-quality center points to improve the segmentation result. In contrast, our methods can directly regress the boundary location without any further center selection process. As for the proposed CNN aggregation mechanism, some low-level features are unnecessarily over-extracted while object boundaries are simultaneously under-sampled. In order to extract more useful and representative features, I apply the ARM working as a filter between CNN encoder and GCN decoder, which cooperates with the GCN to gain more effective semantic and spatial features, especially the boundary location information from CNN.

In summary, this work makes the following contributions:

- I take a straightforward and intuitive approach to (biomedical) image semantic segmentation and regard it as a direct boundary regression problem in an end-to-end fashion.
- I propose aggregating mechanisms on both CNN and GCN modules, to enable them to reuse and fuse the contextual and spatial information. The additional attention mechanism helps the GCN decoder to gain more useful semantic and spatial information from the CNN encoder.
- I apply a new loss function tailored for object boundary localization that will help to make update step size adaptive to the error values during the training stage.

It is envisaged that the proposed framework may serve as a fundamental and strong baseline in future studies of biomedical semantic segmentation tasks.

7.2 Related Work

7.2.1 Pixel-based Methods

Fully Convolution Neural Networks (FCNs) [247] and U-Net architectures [337] are widely used in semantic segmentation tasks [61,141]. These methods are aimed at extracting more spatial information or extending the receptive field that is of pivotal importance in semantic segmentation tasks. However, it is still difficult to capture longer-range correspondence between pixels in an image [484].

Aggregation module In order to gain global contextual dependencies of an image, methods like [376,471,492,508] proposed to fuse multi-scale or multi-level features through aggregating across semantic and spatial feature domains. Zhao *et al.* [492] proposed a pyramid network that utilizes multiple dilated convolution blocks [469] to aggregating global feature maps on different scales. Other approaches such as Deeplab methods [59–61] exploited parallel dilated convolution with different rates to extract features at an arbitrary resolution and preserve the spatial information. However, it is still hard to efficiently learn the discriminative feature representation as many low-level features are unnecessarily over-extracted. Therefore, these aggregation methods may result in an excessive use of information flow.

Attention mechanism Alternatively, some other algorithms exploited the benefits of attention mechanism to integrate local discriminative representation and global contextual features. For example, DANet and CSNet [110, 295] used the attentions in spatial and channel dimensions respectively to adaptively integrate local features with their global dependencies. Furthermore, Zhao *et al.* proposed the point-wise spatial attention network [493], which connected each position in the feature map with all the others through self-adaptive attention maps to harvest local and long-range contextual information flexibly

and dynamically. In this work, an ARM module is also used to supervise our model to learn discriminate features from input images.

7.2.2 Polygon-based Methods

Instead of assigning each pixel with a class, some recent methods [73, 132, 266, 448, 455] started to predict the position of all vertices of the polygon around the boundary of the target objects. The recent work [448, 455] used polar coordinates to represent object contours. Both methods achieved comparable results with pixel-based segmentation methods in instance segmentation tasks. Also, the combination of FCNs and Active Contour Models (ACMs) [173] has been exploited. Some methods formulated new loss functions that were inspired by the ACMs principles [68, 133] to tackle the task of ventricle segmentation in cardiac MRI. Other approaches used the ACMs as a post-processor of the output of an FCN, for example, Marcos *et al.* [266] proposed a Deep Structured Active Contours model that combined ACMs and pre-trained FCNs to learn the energy surface of the reference map. These ACM-based methods achieved state-of-the-art performance in many segmentation tasks. However, there are still two main limitations. First, the contour curve must be initialized, while the initialized curve is far away from the ground truth, it may be insufficient to optimize or make an inference. Second, due to the iterative inference mechanism of ACMs, they require a relatively longer running time during training and testing.

7.2.3 GCNs in Segmentation

GCNs have been applied to image segmentation tasks recently, as they can propagate and exchange the local short-range information through the whole image to learn the semantic relations between objects [360, 484]. In 2D image semantic segmentation tasks, Li *et al.*

proposed a Dual Graph Convoltional Network (DGCNet) [484], which applied two orthogonal graphs frameworks to compute the global relational reasoning of the whole image and the reasoning process can help the whole network to gain rich global contextual information. Another work [360] proposed by Shin *et al.* shared the similar idea, and utilized GCN to learn the global structure of the shape of the object, which reflected the connectivity of neighbouring vertices. Apart from using GCN to learn global contextual information from 2D input, our approach also exploits spatial and local location information. Compared with a recent similar work [279], our method further exploit the relations between low-level and much more high-level vertex information in GCN decoder and perform a 'skip up sampling' in terms of Graph convolutions between two layers. This operation helps our model further extract feature correlations among different layers.

7.3 Method

7.3.1 Graph Representation

The manually annotated object boundaries are extracted from the binary image and equally sampled into N vertices with the same angle interval $\Delta\theta$ (e.g. N = 360, $\Delta\theta = 1°$). The geometric center of the boundary represents the center vertex. I describe the object contour with vertices and edges as $B = (V, E)$, where V has $N + 1$ vertices in the Euclidean space, $V \in \mathbb{R}^{N\times 2}$, and $E \in \{0, 1\}^{(N+1)\times(N+1)}$ is a sparse adjacency matrix, representing the edge connections between vertices, where $E_{i,j} = 1$ means vertices V_i and V_j are connected by an edge, and $E_{i,j} = 0$ otherwise. Every two continuous vertices on the contour are connected with an edge and are both connected to the center vertices with another two edges to form a triangle. For the OD and OC segmentation, their contours are sampled separately while the geometric centre of the OC is shared as the centre vertex. Thus, there are 360 triangles

and 361 vertices for instances in FH images and 720 triangles and 721 vertices for OD and OC images. For more details, please refer to the supplementary material.

I directly use the coordinates in the Euclidean space to represent all the vertices and exploit the semantic and spatial correspondence between the inputs' instance and boundaries. Besides, our boundary representation method is not sensitive to the center point as the boundary does not have too many correlations with the center point.

7.3.2 Graph Fourier Transform & Convolution

According to [79], the normalized Laplacian matrix is $L = I - D^{-\frac{1}{2}}ED^{-\frac{1}{2}}$, where I is the identity matrix, and D is a diagonal matrix that represents the degree of each vertex in V, such that $D_{i,i} = \sum_{j=1}^{N} E_{i,j}$. The Laplacian of the graph is a symmetric and positive semi-definite matrix, so L can be diagonalized by the Fourier basis $U \in \mathbb{R}^{N\times N}$, such that $L = U\Lambda U^T$. The columns of U are the orthogonal eigenvectors $U = [u_1, ..., u_n]$, and $\Lambda = diag([\lambda_1, ..., \lambda_n]) \in \mathbb{R}^{N\times N}$ is a diagonal matrix with non-negative eigenvalues. The graph Fourier transform of the vertices representation $x \in \mathbb{R}^{N\times 3}$ is defined as $\hat{x} = U^T x$, and the inverse Fourier transform as $x = U\hat{x}$. The spectral graph convolution of i and j is defined as $i*j = U((U^T i) \odot (U^T j))$ in the Fourier space. Since U is not a sparse matrix, this operation is computationally expensive. To reduce the computation, Defferrard *et al.* [85] proposped that the convolution operation on a graph can be defined in Fourier space by formulating spectral filtering with a kernel g_θ using a recursive Chebyshev polynomial [85]. The filter g_θ is parametrized as a Chebyshev polynomial expansion of order K, such that

$$g_\theta(L) = \sum_{k=1}^{K} \theta_k T_k(\hat{L}) \tag{7.1}$$

where $\theta \in \mathbb{R}^K$ is a vector of Chebyshev coefficients, and $\hat{L} = 2L/\lambda_{max} - I_N$ represents the rescaled Laplacian. $T_k \in \mathbb{R}^{N\times N}$ is the Chebyshev polynomial of order K, that can be

recursively computed as $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$ with $T_0 = 1$ and $T_1 = x$. Therefore, the spectral convolution can be defined as

$$y_j = \sum_{i=1}^{F_{in}} g_{\theta_{i,j}}(L)x_i \tag{7.2}$$

where x_i is the i-th feature of input $x \in \mathbb{R}^{N \times F_{in}}$, which has F_{in} features, with $F_{in} = 2$ in this work and $y \in \mathbb{R}^{N \times F_{out}}$ is the output. The entire filter operation is computationally faster and the complexity drops from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$ [41].

7.3.3 Graph Vertices Sampling

To achieve multi-level aggregated graph convolutions on different vertex resolutions, I follow [323] to form a new topology and neighbour relationships of vertices. More specifically, I use the permutation matrix $Q_d \in \{0,1\}^{m \times n}$ to down-sample m vertices, m = 360 or 720 in our work. Q_d is gained by iteratively decreasing vertices, which uses a quadratic matrix to keep the approximations of the surface error [115]. The down-sampling is a pre-processing, and the discarded vertices are saved with barycentric coordinates. I conduct up-sampling with another transformation matrix $Q_u \in \mathbb{R}^{m \times n}$. The up-sampled vertices V_u can be obtained by a sparse matrix multiplication, i.e., $V_u = Q_u V_d$, where V_d are down-sampled vertices.

7.3.4 Proposed Aggregation Network

Our novel aggregation graph regression network is motivated by fusing features hierarchically and iteratively [376, 471, 508], which consists of an image context encoder, an attention refinement module and a vertex location decoder. Both the encoder and decoder contain aggregation mechanisms through up-samplings and down-samplings, which provide improvements in extracting the full spectrum of semantic and spatial information across

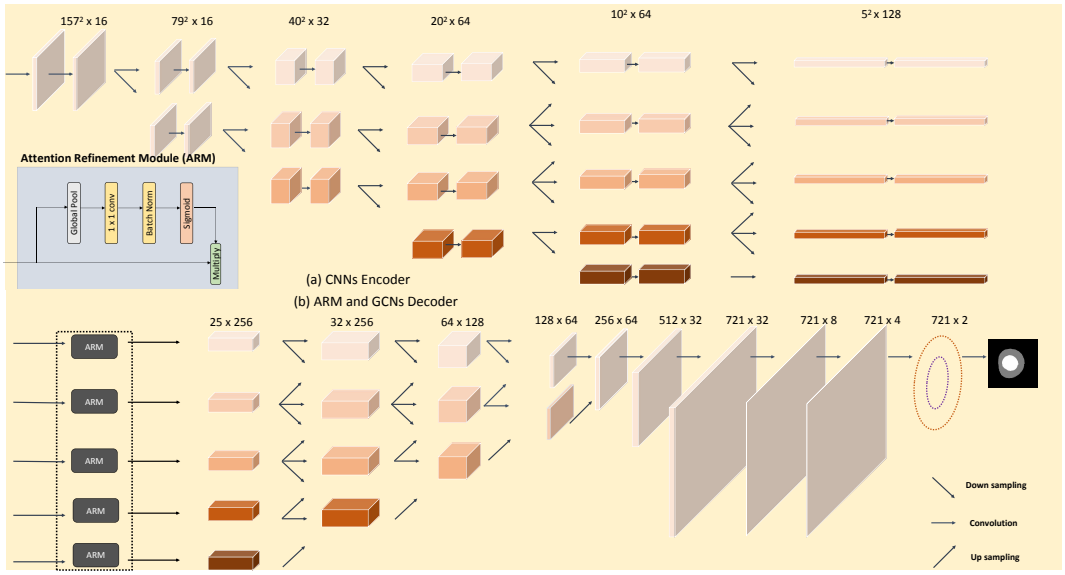


Figure 7.2: Overview of our proposed network structure. The size of feature maps of the CNN encoder and vertex maps of the GCN decoder for each stage (columns) are shown. In the CNN encoder, the horizontal black arrow represents CNN convolutional operations that are achieved by a standard CNN Residual Block [143] with kernel size 3 x 3, stride 1, followed by a Batch Normalization (BN) layer [159] and Leaky ReLU as the activation function. The down-sampling is conducted by setting stride size as 2, the lower level feature is bi-linearly up-sampled by a factor 2. In the GCN decoder, down-sampling and up-sampling are conducted by graph vertices sampling, which is described in Section 3.3, and the horizontal black arrow represents residual graph convolution (ResGCN) blocks [204] with polynomial order 4. The horizontal blue arrow achieves 'skip up sampling' with vertices number four times up sampled in terms of graph vertices sampling method via retained vertices. In this figure, the example is for OD and OC segmentation, and for FH segmentation, the convolution operation will be the same. Still, the feature map and vertex map size will be different because of different input size and number of contours of instances.

stages and resolutions. Besides, the attention module plays an essential role to guide the feature learning and refine the output from the CNN encoder, then passes to the GCN decoder through multi-paths. In Section 5.3, our ablation study demonstrates that the proposed aggregation module helps to extract more useful information, and the attention module helps to refine the extracted features from the encoder to guide feature learning

better.

Semantic Encoder Fig. 7.2 (a) shows the detailed structure of our semantic encoder, which maintains high-resolution representations by connecting low-to-high resolution convolutions in parallel, where multi-scale fusions are repeated across different levels (rows). Our encoder is designed to lessen the spatial information loss and extract a wider spectrum of semantic features through different receptive fields. The encoder takes input images of shape 314×314×3 (Fundus OD & OC images) or 140×140×1 (Ultrasound FH images), with operations of up-sampling and down-sampling. The aggregation block can extract and reuse more features across various resolutions and scales, which helps to reduce spatial information loss during the encoding process.

Attention Module: I propose an Attention Refinement Module (ARM) to refine the features from the outputs of the encoder. As Fig. 7.2 (a) & (b) shows, ARM contains five attention blocks, and each block employs global average pooling to capture global context through the different channels, and conducts an attention tensor to lead the emphasis of feature learning through a convolution layer followed by a BN layer and sigmoid as the activation function. For the filter, the kernel size is 1×1, and the stride is 1. This design can refine the output features of each stage in the Semantic Encoder, which easily integrates the global context information.

Spatial Decoder The decoder takes refined multi-paths outputs from the attention module, then employ ResGCN blocks [204] through different stages and levels, which has been shown that as layers go deeper, ResGCN blocks can prevent vanishing gradient problems. As Fig. 7.2 (b) shows, our decoder fuses and reuses the features extracted by ResGCN blocks through different stages. Benefits from the graph Vertices sampling, our decoder can regress the location of the vertices from sparse to dense, which allows the ResGCN blocks to hierarchically extract spatial location information from refined outputs of the

attention module. For each ResGCN Block, it consists of 4 graph convolution layers, and each graph convolution layer is followed by a Batch Normalization layer [159] and Leaky ReLU as the activation function. After ResGCN blocks and graph vertices up-samplings, the number of vertices is up-sampled from 25 to 721, and each vertex is represented by a vector of length 32. Different from [279], Our decoder further explored the relations between low and high level resolution of vertices features, which improves the performance and is shown in Section 4. At last, three graph convolution layers are added to generate 2D object contour vertices, which reduces the output channels to 2, as each contour vertex has two dimensions: x and y. With the output from the decoder, I connect every two consecutive vertices on the boundary to form a polygon contour as the final segmentation result.

7.3.5 Loss Function

L2 and L1 loss have been widely used in regression tasks, such as object detection [120,142] and human pose estimation [389]. However, it is difficult for the L1 loss to find the global minimization in the late training stage without fine-tuning of the learning rate. L2 loss is sensitive to outliers which may result in unstable training in the early training stage.

In this work I solve segmentation as a contour vertices location regression problem. Following Wing-loss [106] and Smooth-L1 loss [119], I adopt a new loss function, Fan-loss (Fig. 7.3) that can take small update steps when reaching small range errors in the late training stage and can remain stable training during the early training stage. This loss function is defined as:

$$L(x) = \begin{cases} W[e^{(|x|/\epsilon)} - 1] & \text{if } |x| < W \\ |x| - C & \text{otherwise} \end{cases} \tag{7.3}$$

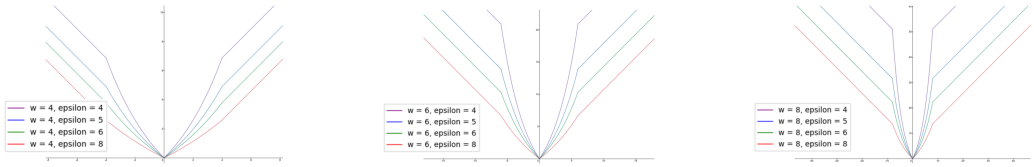


Figure 7.3: The loss function plotted with different parameter settings, where w controls the non-linear part and epsilon (ϵ) limits the curvature.

Where W is non-negative and decide the range of the non-linear part, ϵ limits the curvature between $(-W, W)$ and $C = W - W[e^{(|w|/\epsilon)} - 1]$ connects the linear and non-linear parts. After several evaluation experiments, the parameter W is set to 8 and ϵ to 5 for FH segmentation and W = 6, $\epsilon = 5$ for OD & OC segmentation. For the OD & OC segmentation tasks, I integrate a weight mask and assign more weights to the vertices that belong to the OC, as OC is usually difficult to segment because of poor image quality or low color contrast.

7.4 Experiments

7.4.1 Datasets

I evaluate our approach with two major types of biomedical images on two segmentation tasks respectively: fundus images of retinal for OD & OC segmentation, and ultrasound images of the fetus for FH segmentation.

Fudus OD & OC images: 2068 images from five datasets are merged together. 190 fundus images are randomly selected as the retina test dataset, the rest 1878 fundus images are used for the training. Considering the negative influence of non-target areas in fundus retina images, I first localize the disc centers by detector [329] and crop to 314 × 314 pixels and then transmit into our network. **Refuge** [307] consists of 400 training images and 400 validation images. The pixel-wise OD & OC gray-scale annotations are provided. **Drishti-**

GS [370] contains 50 training images and 51 validation images. All images are taken centered on OD & OC with a field-of-view of 30 degrees. The annotations are provided in the form of average boundaries. **ORIGA** [490] contains 650 fundus images. The OD & OC boundaries were manually marked by experienced graders from the Singapore Eye Research Institute. **RIGA** [6] contains 750 fundus images from **MESSIDOR** [84] database. The OD and OC are labeled manually by six ophthalmologists and the mean OD and OC are used ad the ground truth. **RIM-ONE** [111] contains 169 fundus images, annotated by five different experts.

Ultrasound FH images: The HC18-Challenge dataset are used which contains 999 two-dimensional (2D) ultrasound images with size of 800 $\times$ 540 pixels collected from the database of Radboud University Medical Center [398]. I apply zero-padding to each image to 840 $\times$ 840 pixels, and then resize into 140 $\times$ 140 as the input image, then I randomly select 94 images as the test dataset, and the model is trained on the rest 905 images.

7.4.2 Implementation Details

To augment the dataset, I randomly rotating the input image of training dataset for both segmentation tasks. To be specific, the rotation ranges from -15 to 15 degree. I randomly select 10% of training dataset as the validation dataset. I use stochastic gradient descent with a momentum of 0.9 to optimize the Fan-loss. The number of graph vertices for FH is sampled to 361, 256, 128, 64, 32, 25 crosses five stages with Graph Vertices Sampling introduced in Section 3.3. I trained our model for 300 epochs for all the experiments, with a learning rate of 1e-2 and decay rate of 0.997 every epoch. The batch size is set as 48. All the training processes are performed on a server with 8 TESLA V100 and 4 TESLA P100, and all the test experiments are conducted on a local workstation with Geforce RTX 2080Ti.

Tasks / Methods	OC		OD			FH		
	Dice Score	AUC	Dice Score	AUC	vCDR	Dice Score	AUC	HD(mm)
U-Net [337]	0.9016	0.9186	0.9522	0.9648	0.0674	0.9625	0.9688	1.79
M-Net [109]	**0.9335**	**0.9417**	0.9230	0.9332	0.0488	-	-	-
U-Net++ [508]	0.9198	0.9285	0.9626	0.9777	0.0469	0.9701	0.9789	1.73
DANet [110]	0.9232	0.9327	0.9654	0.9726	0.0450	0.9719	0.9786	1.69
DARNet [73]	0.9235	0.9339	0.9617	0.9684	0.0455	0.9719	0.9790	**1.52**
PolarMask [448]	0.9238	0.9366	**0.9670**	**0.9782**	**0.0419**	0.9723	0.9780	1.66
DeepLabv3+ [61]	**0.9308**	**0.9406**	0.9669	0.9779	0.0467	**0.9779**	**0.9819**	**1.58**
CGRNet [279]	0.9246	0.9376	**0.9688**	**0.9784**	**0.0438**	**0.9738**	**0.9796**	**1.58**
Our method	**0.9255**	**0.9385**	**0.9697**	**0.9791**	**0.0421**	**0.9746**	**0.9801**	**1.47**

Table 7.1: Segmentation results on retina test dataset for OD & OC and on HC18-Challenge [398] for FH. The performance is reported as Dice score (%), AUC (%), mean absolute error of Hausdorff distance (HD) for FH and mean absolute error of the vertical cup-to-disc ratio (vCDR) for OD & OC. The top three results in each category are highlighted in bold.

7.5 Results

In this section, I present our experimental results on the OD & OC and FH segmentation task in comparison to other state-of-the-art methods. I compare our model with other state-of-the-art methods, including U-Net [337], PolarMask [448], M-Net [109], U-Net++ [508], DANet [110], DARNet [73], DeepLabv3+ [61], CGRNet [279] through running their open public source code. Dice score and Area Under the Curve (AUC) are used as the segmentation accuracy metrics. The results of an ablation study are shown in order to demonstrate the effectiveness of the proposed aggregation mechanism, attention mechanism and loss function, respectively.

7.5.1 Optic Disc & Cup Segmentation

The retinal dataset I used is merged from five different fundus OD & OC images datasets. In terms of different dataset sources, they may contain different annotation standards for ground truths by different doctors. However, our model still achieve good performance, which shows the robustness and generalizability of our model. Fig. 7.4 shows some quali-

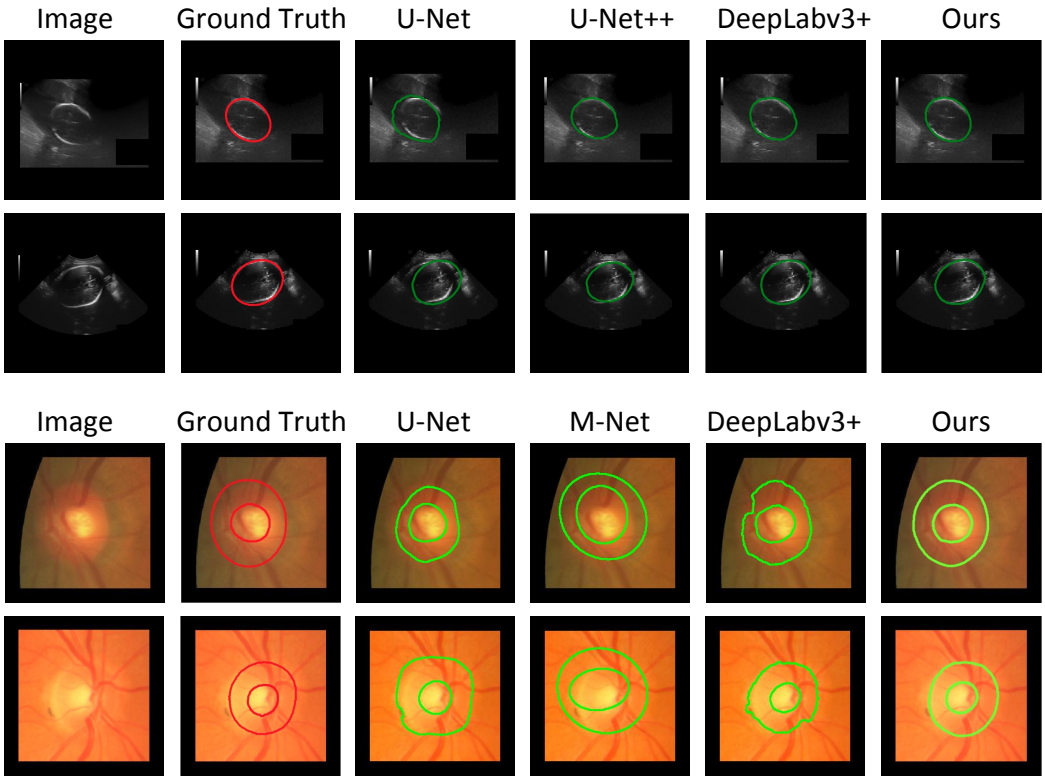


Figure 7.4: Qualitative results of segmentation on the testing images of the fundus dataset and HC18-Challenge [398]. Top two rows are the ultrasound FH segmentation results, and the bottom two rows are the fundus OD & OC segmentation results.

tative results. I achieve 0.9697 and 0.9255 Dice similarity score on OD & OC segmentation respectively, which are comparable with other pixel-wise based state-of-the-art methods even without any bells and whistles (e.g. multi-scale training, ellipse fitting, longer training epochs, etc.). Tab. 7.1 provides the results of ours and the other methods. As for the inference speed, our model uses 64.1 milliseconds (ms) per image that is faster than PolarMask [448] (72.1 ms) and DeepLabv3 [61] (323.9 ms). In the supplementary material, I also show some 'failed' cases compared with the ground truth. According to the comments from an anonymous expert at the Liverpool Reading Center, our model produces more

Tasks / Loss Function	OC		OD		FH	
	Dice Score	AUC	Dice Score	AUC	Dice Score	AUC
L1	0.9111	0.9259	0.9546	0.9639	0.9505	0.9688
L2	0.9105	0.9210	0.9551	0.9666	0.9440	0.9568
Smooth-L1 [119]	0.9088	0.9114	0.9523	0.9655	0.9394	0.9454
Fan-Loss						
weight mask = 0	0.9184	0.9220	0.9618	0.9739		
weight mask = 3	0.9221	0.9337	0.9649	0.9769		
weight mask = 5	**0.9255**	**0.9385**	**0.9697**	**0.9791**	**0.9746**	**0.9801**
weight mask = 7	0.9175	0.9240	0.9624	0.9720		
weight mask = 9	0.9107	0.9213	0.9600	0.9705		

Table 7.2: Performance comparisons between different loss function and weight mask parameter settings on the OD & OC segmentation and the FH segmentation respectively. For weight mask = 5, our model achieves best performance on the OD & OC segmentation.

accurate results than the ground truth. This highlights the potential issue of imperfect ground truth in many deep learning applications.

7.5.2 Fetal Head Segmentation

Tab. 7.1 and Fig. 7.4 shows the quantitative and qualitative results, our model achieves 0.9746 Dice similarity score and 0.9801 % AUC, which outperforms DARNet [73] and DANet [110] by 0.3%. Our model (59.1ms) is faster than PolarMask [448] (65.5 ms) and Deeplabv3+ [61] (290.3ms) for per image inference.

7.5.3 Ablation Study

I investigate the effect of each component in our proposed model. All the ablation experiments are performed with the same setting as section 4.2 described. The performance in the form of Dice score and AUC are reported in Fig. 7.5, Tab. 7.2 and 7.3. The best performance in each experiment is highlighted in bold. For more qualitative results, please refer to the supplementary material.

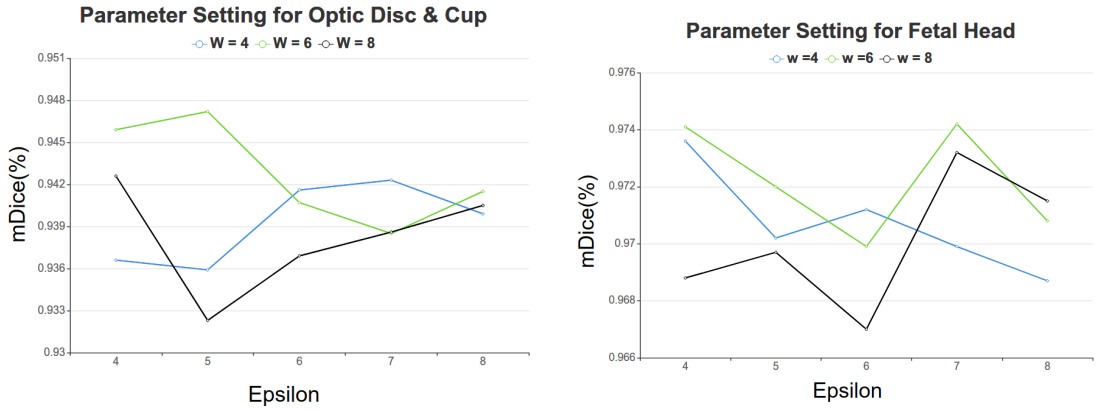


Figure 7.5: A comparison of different parameter settings (w and ϵ) for Fan-loss function, measured in terms of the mean Dice score on the fundus dataset for OD & OC. With w = 6, ϵ = 5, our model achieves the best performance (0.9255 & 0.9697). On the HC18-Challenge test dataset [398] for FH segmentation, with w = 6, ϵ = 7, our model gains the best results 0.9746). It shows that our network is not sensitive to these parameters as no significantly different results are found.

Ablation on Parameters of Loss Function I perform Experiments to evaluate the effect of parameter settings of Fan-loss function. When w = 6, ϵ = 5, our model achieve the best performance on OD & OC segmentation test dataset, and w = 6, ϵ = 7, for FH segmentation test dataset. For more details, please refer to Fig. 7.5.

Ablation on Loss Function I conduct experiments to evaluate the effectiveness of the loss function. I compare with L1, L2, Smooth-L1 [119] loss functions, which are commonly used in the regression problem. Tab. 7.2 shows the quantitative results on OD & OC and FH segmentation tasks respectively. As illustrated, Fan-loss function attains a superior performance over the other three loss functions. In particular, it achieves a mean Dice score that is 1.6% relatively better than that of L1 loss function on OD & OC and 2.7% relatively better than L1 loss function on FH segmentation. Tab. 7.2 shows comparing with no-weight mask loss function, our proposed weight mask helps to improve OD & OC segmentation results by 0.79% when weight mask = 5 is used.

Ablation on Angle Interval Experiments are conducted to evaluate the effect of different angle intervals $\Delta\theta$ for vertices sampling. The larger angle interval indicates the smaller number of vertices sampled on the contour. With $\Delta\theta = 1°$, our model achieves best performance on both the FH segmentation and the OD & OC segmentation. The results are shown in supplementary material.

Ablation on Structure Components In this section, I evaluate the effectiveness of our aggregation module, attention module and GCN decoder. First, I compare with no-aggregation structure network, in which I remove all the aggregation parts and attention modules to form a standard encoder-decoder network structure. Then I add aggregated CNN and GCN module to form an aggregation network. To further improve the performance, I design an attention module, and the effect of the attention module is presented in Tab 7.3. Furthermore, I evaluate the effectiveness of proposed GCN decoder and replace the GCN with CNN, which are the same as I used in the encoder. As illustrated, for the FH segmentation, the proposed aggregation module helps to improve 1.83% on Dice score over the no-aggregation method, the ARM module further improves 0.47%, and GCN decoder further improves 1.11%. For the OD & OC segmentation, the aggregation module improves 1.17 % on average by Dice score, the ARM improves 0.64%, and the GCN decoder improves 1.73%.

7.5.4 Data Representation

The left graph of Fig. 7.6 illustrates how fetal head (FH) boundaries are represented to make it compatible for GCN. The boundary is represented by equally sampled vertices along it and its geometric center is defined as the center vertex. Each triangle consists of three vertices and three edges where two vertices are from the boundary and the other is the center vertex. Then, the vertices locations and their geometric relationships defined by

Tasks / Methods	OC		OD		FH	
	Dice Score	AUC	Dice Score	AUC	Dice Score	AUC
No Aggregation (Encoder + Decoder)	0.9025	0.9065	0.9589	0.9665	0.9567	0.9690
Aggregation	0.9207	0.9303	0.9624	0.9660	0.9700	0.9776
Aggregation + ARM (with CNN decoder)	0.9099	0.9178	0.9529	0.9635	0.9639	0.9758
Aggregation + ARM (Our method)	**0.9255**	**0.9385**	**0.9697**	**0.9791**	**0.9746**	**0.9801**

Table 7.3: Ablation study on different structure components of the loss function (w = 6, $\epsilon = 5$ for FH segmentation and w = 6, $\epsilon = 7$ for OD & OC).

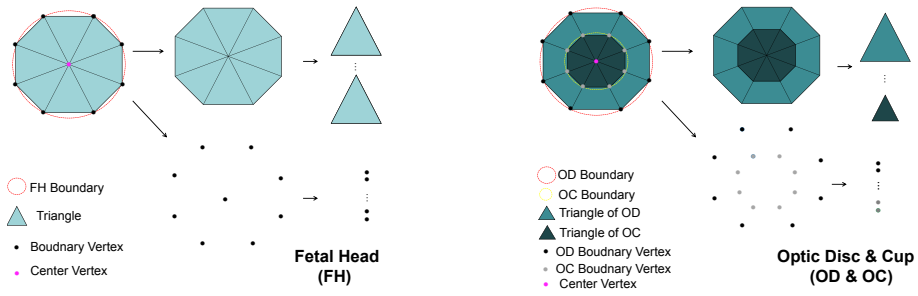


Figure 7.6: Illustration of the object contours representation, left: Fetal Head, right: Optic Disc and Optic Cup.

an adjacency matrix from the triangulations can be used by GCN. For the optic disc (OD) and optic cup (OC) segmentation, the centre of the OC is shared as the centre vertex. However, triangulations are made for both the OD and OC, as demonstrated by the right graph of Fig. 7.6.

Tasks / Angle Interval	OC		OD		FH	
	Dice Score	AUC	Dice Score	AUC	Dice Score	AUC
40°	0.9025	0.9094	0.9153	0.9231	0.9416	0.9503
18°	0.9104	0.9195	0.9489	0.9555	0.9516	0.9560
10°	0.9196	0.9284	0.9584	0.9648	0.9603	0.9695
5°	0.9239	0.9307	0.9629	0.9716	0.9710	0.9777
2°	0.9245	0.9377	**0.9691**	0.9783	0.9739	**0.9799**
1°	**0.9255**	**0.9385**	**0.9697**	**0.9791**	**0.9746**	**0.9801**

Table 7.4: Ablation study on different angle interval samplings. With angle interval = 1° or 2°, our model achieves comparable segmentation results on the OD & OC and FH segmentation tasks, and at the end, angle interval = 1° is chosen for our model. Dice score (%) and AUC (%) are reported for the segmentation on OD & OC and FH test dataset.

7.5.5 Ablation Study on Angle Interval

In this section, I demonstrate the robustness of our model *w.r.t* the different hyper-parameters of angle intervals. As the angle interval changes, the number of vertices will vary as well. This results in smooth or rough boundaries, which affect the final segmentation performance significantly. As Table. 7.4 shows that our model can achieve comparable segmentation performance on two tasks when the angle interval is less than 5 °.

7.5.6 Discussion: Comparison with Ground Truth

For each retina image, when the average Dice score of OD & OC segmentation is lower than 0.85 or our model's segmentation is deviated much from the ground truth, it will be regarded as a 'failed' case. Results of some 'failed' cases in the OD and OC segmentation are shown in Fig. 7.7. I overlaid segmentations by using our model (green), and the ground truth (red) for better comparison with the center points shown. An expert from an anonymous accredited ophthalmology reading center confirmed that for these cases our segmentations are more accurate than the ground truth. This highlights the robustness of our model as well as the limitations of the ground truth made from manual annotations.

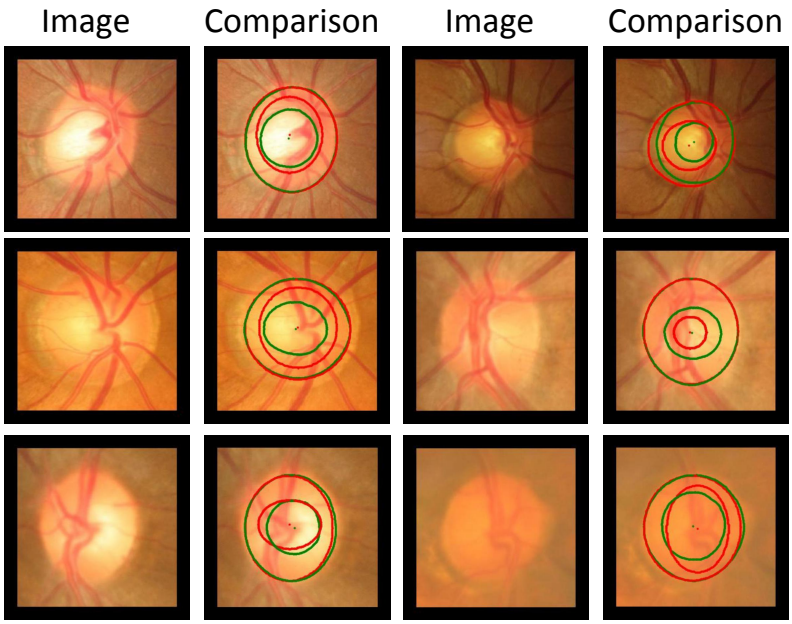


Figure 7.7: Illustration of the comparison between our segmentation (green) and the ground truth (red) in some 'failed' cases. The ground truth has inaccurate OC boundaries for most of the cases (The top right corner one is inaccurate in both OC and OD boundaries). Our model can produce more accurate boundaries than the ground truth according to an expert from an anonymous expert at an accredited ophthalmology reading center.

7.5.7 More Qualitative Results

In Fig. 7.8 and Fig. 7.9, I showed the effect of L1 loss, L2 loss, Smooth-L1 loss, and the Fan-loss function on the segmentation of FH and OD and OC, respectively. Intuitively, Fan-loss function produces more faithful and accurate results.

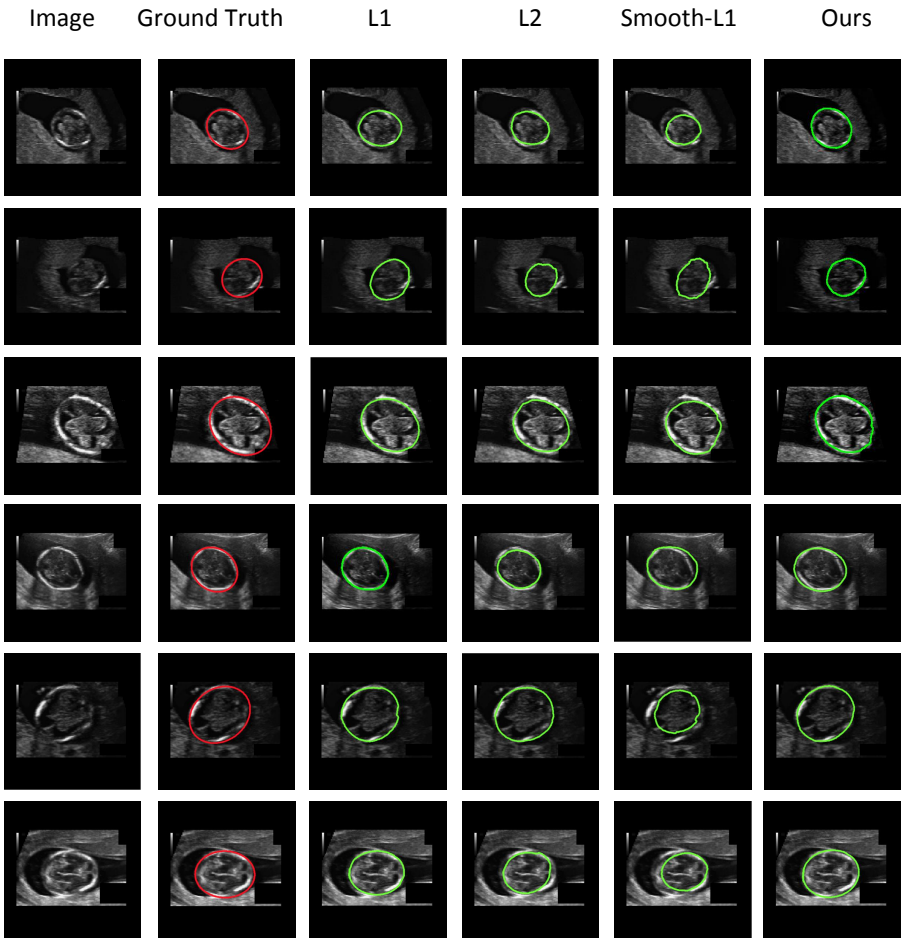


Figure 7.8: Comparison in fetal head segmentation when different loss functions are used. The Fan-loss function can produce more accurate and faithful boundaries. In each row, from left to right is the original image, ground truth, segmentations of using L1 loss (L1), L2 loss (L2), smooth-L1 loss (Smooth-L1) and ours.

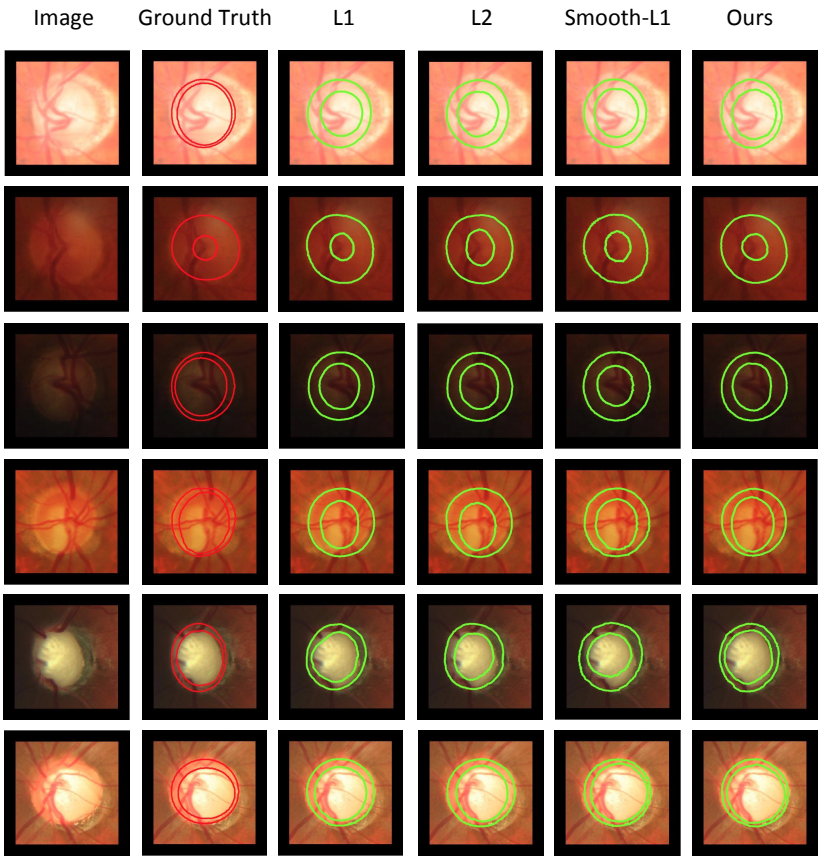


Figure 7.9: Comparison of the OD and OC segmentations by using different loss functions. The Fan-loss function can produce more accurate boundaries, especially for the OC. In each row, from left to right is the original image, ground truth (GT), segmentations of using L1 loss (L1), L2 loss (L2), smoothed-L1 loss (Smooth-L1) and ours.

7.6 Conclusion

I propose a straightforward regression method for segmentation tasks by directly regressing the boundary of the instances instead of pixel-wise dense predictions. I have demonstrated its potentials on the segmentation problems of the fetal head and optic disc & cup. In the future work, I will study to extend the proposed model to tackle 3D biomedical image

segmentation tasks.

Chapter 8

Researching Dense Geometric Data with Explicit Graph Representations

In this chapter, I address the challenge that tackling large-scale nodes' (vertices) location tasks with graph-structured datasets. In detail, I applied the proposed method on the task of 3D face reconstruction task with a large amount of face vertices. Specifically, I propose a novel multi-level aggregation network to regress the coordinates of the vertices of a 3D face from a single 2D image in an end-to-end manner. This is achieved by seamlessly combining standard convolutional neural networks (CNNs) with Graph Convolution Networks (GCNs). By iteratively and hierarchically fusing the features across different layers and stages of the CNNs and GCNs, our approach can provide a dense face alignment and 3D face reconstruction simultaneously for the benefit of direct feature learning of 3D face mesh. Experiments on several challenging datasets demonstrate that our method outperforms state-of-the-art approaches on both 2D and 3D face alignment tasks.

8.1 Introduction

Face alignment and 3D face reconstruction are two interrelated problems in the field of computer vision and graphics research and industrial applications. Face alignment aims to locate specific 2D face landmarks, which is essential for most facial image applications such as face recognition [404], facial expression recognition [162] or head pose analysis [89]. However, problems such as occlusions, large pose, and extreme lighting conditions make it a difficult task. In the past decades, researchers started to solve face alignment problems through 3D facial reconstruction by exploring the strong correlations between 2D landmarks and 3D faces. Since the introduction of 3D Morphable Model (3DMM) in 1999 [32], several methods have been proposed to extend it to restore a 3D face mesh from a 2D facial image [96,161,390,394], which can provide both 3D face reconstruction and dense face alignment results. Convolution Neural Networks (CNNs) have been researched in many computer vision area such as classification [38,39,316], segmentation [88], registration [67], *etc.*. More recently, it has been used to directly regress the parameters of 3DMM model from images [171, 333, 510]. However, the performance of these model-based methods are still limited by the face reconstruction from a low-dimensional subspace of parametric 3DMM model.

To address this problem, different strategies have been proposed by using the recent deep learning methods to regress the 3D face coordinates from 2D representations, such as Projected Coordinate Code (PNCC) [333], quantized conformal mapping [7], depth images [347] and conformal UV maps [105]. Although these methods can regress the 3D geometry from 2D representations, their performance is often susceptible to the noise introduced by the 2D representation process from isolated mesh points.

Graph Convolution Networks (GCNs) have recently shown great potential to tackle non-grid like data such as 3D face meshes [293]. If it is used to perform convolution

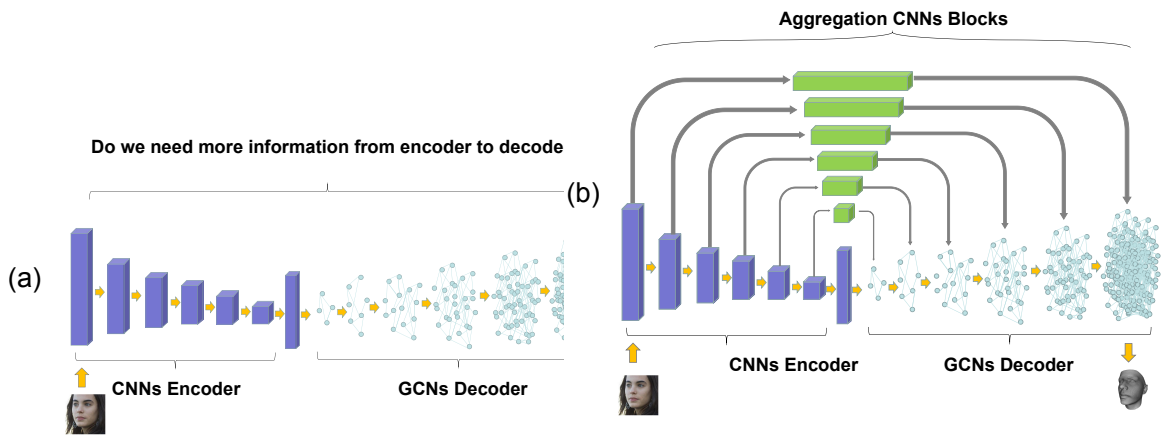


Figure 8.1: Diagrams illustrating the difference between a mesh encoder-decoder and our proposed method. (a) An encoder-decoder structure used by existing methods [507] to regress 3D face mesh from latent embeddings. (b) Our method. As illustrated, our model fuses and reuses multi-level spatial and semantic features from an input face, which works as extra input information to help GCNs decoder to reconstruct the coordinates of face vertices better.

on 3D meshes directly, it will necessitate 2D representations as required by the previous methods and thus reduce (or avoid) noise in the 2D representation. CoMA [323] proposes a mesh encoder-decoder to learn a non-linear representation on the 3D face surface and reconstructs the 3D face mesh via GCNs. Following CoMA, [507] propose an encoder-decoder network, which encodes input images into latent embeddings then decodes the embeddings to 3D face mesh with GCNs. I believe that, during the encoders downsampling process, some content information from face image will be lost. As for the decoder, the only input is the latent embeddings, which cannot adequately represent low-level semantic information and high-level spatial image features of the input face.

In this work, I propose an end-to-end approach that directly learns multi-level regression mappings from image pixels to 3D face mesh vertices by seamlessly combining CNNs and GCNs for 3D face alignment and reconstruction. In this model, I perform feature

learning on face meshes and utilize additional multi-level features fused from the input image in a hierarchical manner that helps GCNs regress more accurate 3D face vertices. Our model attains superior performance on 2D and 3D face alignment tasks to state-of-the-art methods. In particular, our model outperforms other methods by a large margin on the large pose face alignment problem, because with the help of aggregative feature learning, our model gains more useful information from visible parts of the input face image, which helps GCNs better regressing the invisible mesh vertices. Our model is light-weight and only needs 16.0 ms to provide 3D face vertices on a test image.

8.1.1 Contributions

Our approach works well with all kinds of face images, including arbitrary poses, facial expressions and occlusions. The contributions of our work are as follows:

1.) To the best of our knowledge, this is the first time that 3D facial geometry is directly recovered from 2D images in an end-to-end fashion through fusing features from different levels enabled by connections between CNNs and GCNs. I demonstrate that low-level semantic information and the high-level spatial feature can be fully utilized to estimate 3D facial geometry. This is different from the recently proposed encoder-decoder networks [507], which only use low-level latent embeddings. 2.) I propose a novel light-weight and efficient aggregation network to regress more accurate 3D face mesh vertices from corresponding in-the-wild 2D facial images. For training, I propose a new loss function for facial landmarks localization, which helps to prevent taking large update steps when approaching a small range of errors in the late training stage. 3.) Comprehensive experiments have been undertaken on several challenging datasets to evaluate the performance of the new model. The quantitative and qualitative results confirmed its superiority to other state-of-the-art approaches. In particular, our model outperforms previous methods

on 2D and 3D large pose face alignment tasks by more than 18% relative improvement.

8.2 Related work

8.2.1 3D Morphable Models

3DMM is an affine parametric model of face geometry where the texture is learned from high-quality face scans [32]. It is a PCA-based implementation that produces new shape instances from a combination of linear bases of the training images. Recent approaches [96, 170, 171, 332, 510] can be seen as an extension of 3DMM by estimating the 3DMM parameters by using CNN networks in a supervised manner. [171,332,510] proposed using cascaded CNNs to approximate the non-linear optimization function and to regress the 3DMM parameters iteratively. They demonstrated the effectiveness of CNNs in solving the complex mapping function from a 2D face image to 3DMM parameters, but it took a long time to train the network due to the iterations. [96,170,394] proposed end-to-end CNNs to directly estimate the 3DMM parameters. In particular, [394] used a very deep CNN to regress shape and texture parameters of 3DMM for 3D face recognition to improve the discriminative identity of reconstructed face meshes. Other methods like [33,34,87,116,117] focused on optimization-based texture generation methods, for example, Booth *et al.* [34] used 3DMM fits to in-the-wild images and Principal Component Pursuit with missing values to complete the unobserved texture. Both [87,117] employed Generative Adversarial Networks (GANs) to learn a powerful generator of facial texture, in particular, Gecer *et al.* [117] used differentiable rendering layer to self-supervise model to learn the texture information.

8.2.2 Geometric Deep Learning

GCNs have shown their superior ability on several computer vision tasks such as scene understanding [272,284], image segmentation [276,278,280,286], *etc.*. CNNs are effective on Euclidean data such as images but not good at non-Euclidean domains such as grids in face mesh [40]. To overcome the disadvantages of CNNs, GCNs from geometric deep learning have recently been proposed. Bruna *et al.* [41] proposed convolutions in the spectral domain defined by the eigenvectors of Laplacian graphs whereas the filters were parametrized with a smooth transfer function. Still, it is expensive to compute and unable to extract low-level features on the graph. ChebyNet [85] solved the computational complexity problem with Chebyshev polynomial functions, which directly applied it to Laplacian graphs without computing the Fourier basis. CoMA [323] applied ChebyNet to 3D face meshes to find a low-dimensional non-linear representation of faces with an encoder-decoder structure. By spectral graph convolution and mesh sampling operations, it achieves state-of-the-art results in 3D face mesh generation.

8.2.3 Aggregation Network

Aggregation networks have shown powerful ability in visual recognition tasks because these tasks require rich information that spans channels or depth, scales, and resolutions [471].

Densely connected networks (DenseNets) [153] aggregated across channels and depths. It improved the induction of recognition through propagating features and losses from skip connections, which concatenated every layer in stages. Feature pyramid networks (FPNs) [226] aggregated features across different resolutions and scales. It restricted features through adjusting resolutions and semantics and aggregated over the degrees of a pyramidal component progressive system by top-down and parallel associations. Instead of a skip-connection design, RefineNet [225] introduced a refine module to extract the

Figure 8.2: Qualitative results of face alignment on AFLW2000-3D dataset [510]. Top row: Sparse face alignment results with 68 landmarks plotted, including eyes, eyebrows, nose, mouth, and jawline. Middle row: Faces rendered with the reconstructed depth map. Bottom row: Dense face alignment results with all the 53,215 landmarks plotted. Note, although the results are good as shown by these faces in front view, it may seem the overlays dislocated for faces of side views because the reconstruction is only for the front view as the ground truth available for training is front view.

multi-scale features between encoder and decoder. MCUA [302] used multi-level context ultra-aggregation to combine intra and inter level features for stereo matching. Likewise, DFANet [205] aggregated discriminative features through sub-networks and sub-stages cascade, respectively. RefineNet, MCUA, and DFANet showed good performance on 2D semantic segmentation through aggregating features. Compared to these methods, our proposed aggregation block can fuse and reuse multi-level features iteratively and hierarchically across different layers and stages. Our model combines CNNs and GCNs, solving 2D to 3D face reconstruction and dense face alignment task simultaneously.

8.2.4 Recent Work

On the basis of [323], [75] proposed an intrinsic adversarial architecture to reconstruct more detailed 3D face mesh, and [507] reconstructed the 3D face mesh from a 2D image,

in particular, CMD [507] added additional texture information in the graph structure to simultaneously regress coordinates and colour of the mesh. However, [507] used encoder-decoder networks to reconstruct the 3D mesh, and our work is different from them. As they all utilize encoder to encode the 2D image into latent embeddings with CNNs and decoder that reconstructs the 3D face mesh with GCNs from the latent embeddings. It is believed that only using latent embeddings to represent 2D information is not enough, as some low-level semantic features cannot be represented properly and feature information will be lost during the down-sampling or encoding process. Furthermore, the same situation happens in the up-sampling process. The decoder cannot recover the lost resolution and semantic information very well when latent embeddings are the only input information.

For the 3D face problem, overall facial structure is fixed, semantic information is not very rich, so the low-level semantic information and high-level spatial features may both be valuable. I propose a multi-level regression mappings mechanism between each down-sampled 2D image feature and corresponding 3D face mesh features, equipped with a resolution preserved and feature aggregated network structure. I focus on fusing different depth features along different paths in networks. Our proposed method gains superior performance on 2D and 3D face alignment tasks, especially in the large pose face alignment problem, because our model gains more useful information from visible parts of input face images, which help GCNs to better regressing the invisible mesh vertices.

8.3 Method

8.3.1 Data Representation

I represent the 3D face mesh with vertices and edges, $F = (V, A)$ where V has N vertices in 3D Euclidean space, $V \in \mathbb{R}^{N\times 3}$, and $A \in \{0,1\}^{N\times N}$ is a a sparse adjacency matrix,

representing the edge connections between vertices, where $A_{i,j} = 1$ means vertices V_i,V_j are connected by an edge, and $A_{i,j} = 0$ otherwise.

8.3.2 Graph Fourier Transform

Following [79], the non-normalized graph Laplacian is defined as $L = D - A \in \mathbb{R}^{N \times N}$, with D a diagonal matrix representing the degree of each vertex in V, such that $D_{i,i} = \sum_{j=1}^{N} A_{i,j}$. The Laplacian of the graph is a symmetric and positive semi-definite matrix, so L can be diagonalized by the Fourier basis $U \in \mathbb{R}^{N \times N}$, that $L = U \Lambda U^T$. The columns of U are the orthogonal eigenvectors $U = [u1, ..., un]$, and $\Lambda = diag([\lambda_1, ..., \lambda_n]) \in \mathbb{R}^{N \times N}$ is a diagonal matrix with real, non-negative eigenvalues. The graph Fourier transform of the face representation $x \in \mathbb{R}^{N \times 3}$ is defined as $\hat{x} = U^T x$, and the inverse Fourier transform as $x = U\hat{x}$.

8.3.3 Spectral Graph Convolution

The convolution operation on a graph can be defined in Fourier space by formulating mesh filtering with a kernel g_θ using a recursive Chebyshev polynomial [85]. The filter g_θ is parametrized as a Chebyshev polynomial expansion of order K such that

$$g_\theta(L) = \sum_{k=1}^{K} \theta_k T_k(\hat{L}) \tag{8.1}$$

where $\hat{L} = 2L/\lambda_{max} - I_N$ represents rescaled Laplacian, and parameter θ_k is a vector of Chebyshev coefficients. $T_k \in \mathbb{R}^{N \times N}$ is the Chebyshev polynomial of order K, that can be recursively computed as $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$ with $T_0 = 1$ and $T_1 = x$. Therefore, the spectral convolution can be defined as

$$y_j = \sum_{i=1}^{F_{in}} g_{\theta_{i,j}}(L)x_i \tag{8.2}$$

where input $x \in \mathbb{R}^{N \times F_{in}}$ has $F_{in} = 3$ features, as the face mesh of vertices is 3D and $y \in \mathbb{R}^{N \times F_{out}}$ is the output. This approach is computationally faster and complexity drops from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$, compared with [41].

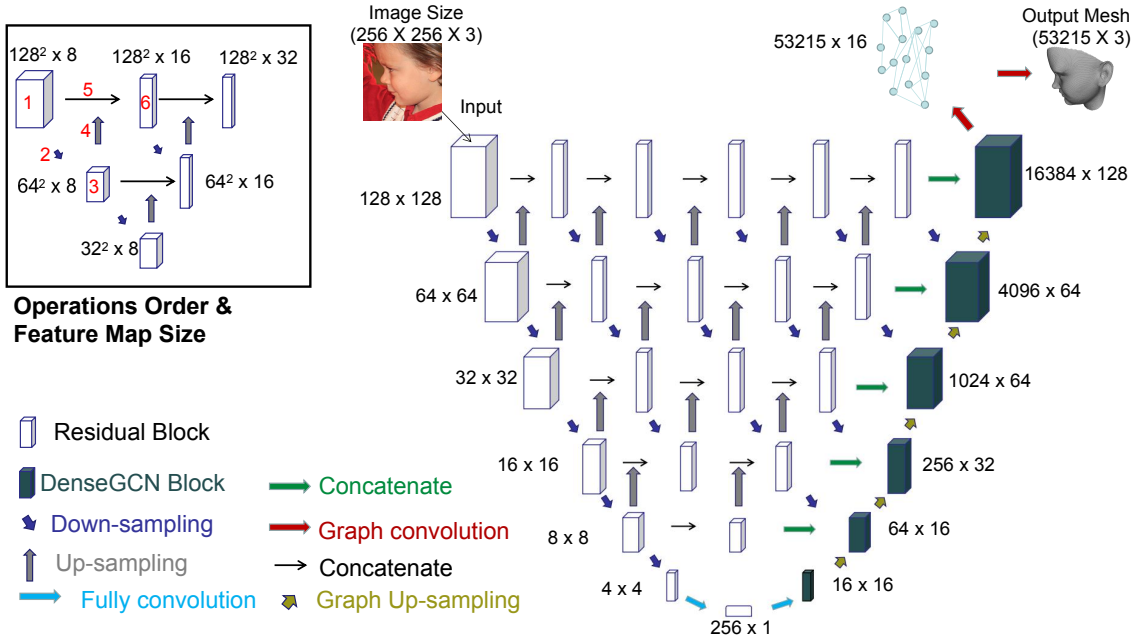


Figure 8.3: Overview of our proposed model. Down-sampling is conducted by setting stride size in the convolution layers as 2. Lower level features are bilinearly up-sampled by a factor 2. On the left branch, I show the feature map size after down-sampling, and on the right branch, I show the vertex feature map size with channels after up-sampling, because I use a vector to represent each vertex. For example, 16384×128 means that 16384 vertices are maintained, and each vertex is represented by a 128×1 vector. The order of operations and feature map size in a small level of aggregate circulation are illustrated in the left side, following the ascending order from 1 to 6 (in red color). As is shown, number 5 is the concatenation of number 1 and number 4's output, then as input to number 6. The green arrow concatenates the output from CNN Residual Block and DenseGCN Block at the same level. Graph down-sampling process is not shown because of the space limitation. More details can be found in Section 3.4.

8.3.4 Mesh Sampling

To achieve multi-scale graph convolutions on joint mesh vertices and 2D feature maps from CNNs, I follow [323] to form a new topology and neighbour relationships of vertices. More specifically, I use the permutation matrix $Q_d \in \{0,1\}^{m \times n}$ to down-sample a mesh with m vertices. $Q_d(i,j) = 1$ denotes the j^{th} vertex is kept, and $Q_d(i,j) = 0$ otherwise. Up-sampling is conducted with another transformation matrix $Q_u \in \mathbb{R}^{m \times n}$. In order to train the CNN and GCN hierarchically and iteratively, I specially design the number of vertices that are maintained in each up-sampling stage, and the feature map size in the down-sampling process, to enable CNNs and GCNs to cooperate in the same level. More details will be shown in Section 3.5.

Down-sampling in GCN is obtained by iteratively contracting vertex pairs, which uses a quadratic matrix to maintain surface error approximations [115]. The discarded vertices during down-sampling are recorded using barycentric coordinates. The up-sampling operates convolution transformations on retained vertices and map the discarded vertices into the down-sampled mesh surface using Barycentric coordinates. The up-sampled mesh with vertices V_u is obtained by a sparse matrix multiplication, i.e., $V_u = Q_u V_d$, where V_d are down-sampled vertices.

8.3.5 Proposed Aggregation Network

Our novel aggregation graph regression network is motivated by fusing features hierarchically and iteratively [471, 508], which is illustrated in Fig. 8.3. Our model can provide improvements in extracting the full spectrum of semantic and spatial information across stages and resolutions. Our network consists of an encoder and a decoder, which are connected by a series of nested residual convolution blocks (aggregation block). As I mentioned in section 3.4, I specially design the number of vertices that remain after each up-sampling

stage in the decoder, and the feature map size after each convolution block in the encoder to make them equal with each other. For example, after first Residual Block, I make the feature map size 128×128, which is equal to the total number of vertices remained after the last DenseGCN block 16384. Our experiments show that maintaining the same graph nodes on the same level helps to improve the performance in our model. I achieve direct end-to-end regression from 2D image to 3D mesh vertices through different feature levels, by making CNNs cooperate with GCNs directly.

The encoder takes input images of shape 256×256×3, and has six residual convolution blocks [143]. After each residual convolution block, the feature map size is decreased by half. This reduction continues until the dimension becomes 4×4×128. Then two fully connected layers are applied to construct a 256×1 dimension embedding.

An aggregation block contains a series of residual convolution blocks, in which there are three convolution layers with identity short-cut connection followed by a Batch Normalization layer [159] and Leaky Relu as the activation function. For each filter, the kernel size is three and the stride is one. Different from the network proposed by [471], our aggregation block achieves fully fused local and global information from encoder into the decoder. Up-sampling operations in the aggregation block from shallow to in-depth, further refining features when extracting 2D image features. Besides, I add down-sampling operations which can project high-resolution features from 2D images into low-resolution 3D mesh features. With up-sampling and down-sampling operations, the aggregation block can extract and reuse more features through different resolutions and scales, which can help to decrease information loss during the encoding process. In Section 5.3, our ablation study demonstrates that the combination of up-sampling and down-sampling helps to extract more useful information. Finally, the aggregation block iteratively and hierarchically aggregates these operations to learn a deep fusion of low and high-level feature information.

The decoder takes embeddings and multi-level outputs from the aggregation block, then decodes with six dense graph convolution blocks (DenseGCN), inspired by [204]. It has been shown that as layers go deeper, DenseGCN can prevent vanishing gradient problems. Our DenseGCN block consists of 4 graph convolution layers, and each graph convolution layer is followed by a Batch Normalization layer [159] and Leaky Relu. After 6 DenseGCN blocks and graph up-sampling operations, the number of vertices is up-sampled from 16 to 16384, and each vertex is represented by a vector of length 128. At last, two graph convolution layers are added to generate a 3D face mesh, which up-samples the number of vertices to 53215 and reduces the vertex feature map channels to 3, as each face mesh vertex has three dimensions: x, y, and z. On the right branch of the network structure in Fig. 8.3, I show the process of up-sampling vertices hierarchically with face meshes. Through an ablation study in Section 5.3, I demonstrate that our proposed method can perform better than non-aggregation or shallow aggregation network in the 3D face alignment task.

8.3.6 Loss Function

L2 and L1 loss have widely been used in facial landmark localization tasks by CNN based networks. It is commonly known that the L2 loss is sensitive to outliers, so in the early training stage, the training process can be unstable. With the L1 loss, it is difficult to continuously converge and find the global minimization in the late training stage without careful tuning of the learning rate. Most of the facial landmarks localization methods use a joint loss function to guide the training process. For example, PRN [105] uses a weighted L2 loss function to make the model pay more attention to the central region of the face. CMD [507] uses a joint loss function where the L2 loss for shape reconstruction, L1 for texture regression and L-render to minimize pixel-wise reconstruction error for facial pixels rendering.

Inspired by Wing-loss [106] and Smooth-L1 loss in [119], I propose a new loss function that can prevent the model from taking large update steps when approaching small range errors in the late training stage and can recover quickly when dealing with large errors during the early training stage. Our loss function is defined as:

$$L(x) = \begin{cases} W[e^{(|x|/\epsilon)} - 1] & if|x| < W \\ |x| - C & otherwise \end{cases} \tag{8.3}$$

Where W should be non-negative and limit the range of the non-linear part, ϵ decides the curvature between $(-W, W)$ and $C = W - W[e^{(|w|/\epsilon)} - 1]$ connects the linear and non-linear parts. After several evaluation experiments, the parameter W is set to 5 and ϵ to 4 in this work.

8.4 Experiments

8.4.1 Datasets

I train our model using semi-annotated in-the-wild data (300W-LP) [510]. The 300W-LP dataset contains 61225 large pose facial images with corresponding 3DMM parameters and pose coefficients, which are synthetically generated by the profiling method [510]. The dataset is produced by fitting a 3DMM model using the multi-feature fitting approach (MFF) [336]. Each image is rendered to 10-15 different poses resulting in a large scale dataset.

For the evaluation of the trained model, I perform extensive quantitative experiments on AFLW2000-3D [510] dataset. It contains 2000 large pose samples from the AFLW dataset [184], annotated with fitted 3DMM parameters and 68 3D landmarks. The sparse and dense face alignment evaluations are performed on this dataset.

AFLW-LFPA is another extension of AFLW dataset constructed by [171]. According to the poses, the dataset contains 1299 test images with a balanced distribution of yaw angles. Besides, each image is annotated with 13 additional landmarks as a expansion to the original 21 visible landmarks in AFLW. Same as [105], I use 34 visible landmarks as the ground truth to measure the accuracy of our results. This database is evaluated on the task of sparse 3D face alignment.

The Florence dataset is a 3D face dataset that contains high-resolution 3D scans of 53 samples which are acquired from a structure-light scanning system. I compare the performance of our method on face reconstruction against other recent state-of-the-art methods.

8.4.2 Implementation Details

I first fit the Basel Face Model (BFM) [32] model to generate and transform the 3D face mesh with corresponding pose coefficients to form the training data set. Specifically, I crop the images according to the ground truth bounding box and rescale them into size 256×256. To augment our dataset, similar to other methods [105], I perturb the input image by randomly rotating and translating. Specifically, the rotation ranges from -45 to 45 degree angles, translation changes is random from 10% of the input image size and has a scale range from 0.9 to 1.2. I use stochastic gradient descent with a momentum of 0.9 to optimize our loss function. I trained our model with a learning rate of 1e-3 and decay rate of 0.99 every epoch. The order of Chebyshev polynomial is set to 3 for all the graph convolution layers. The batch size is set as 48. All training processes are performed on a server with 8 TESLA V100, and all test experiments are conducted on a local machine Geforce RTX 2080Ti.

Table 8.1: Face alignment results on AFLW2000-3D benchmarks. The performance is reported as bounding box size normalized mean error (%). The best result in each category is highlighted in bold, the lower value is better. For any specific head pose, our model outperforms the other methods, and in particular, it defeats the other methods by a large margin for large pose yaw (60° to 90°).

Methods	AFLW2000-3D				AFLW-LFPA
	0° ∼ 30°	30° ∼ 60°	60° ∼ 90°	Mean	Mean
SDM [452]	3.67	4.94	9.67	6.12	-
3DDFA [510]	3.78	4.54	7.93	5.42	-
3DDFA + SDM [510]	3.43	4.24	7.17	4.94	-
N3DMM [390]	-	-	-	4.70	-
DeFA [243]	-	-	-	4.50	3.86
3DSTN [30]	3.15	4.33	5.98	4.49	-
CMD [507]	-	-	-	3.98	-
PRN [105]	2.75	3.51	4.61	3.62	2.93
Bulat *et al.* [44]	2.47	**3.01**	4.31	3.26	-
Jia *et al.* [128]	-	-	-	3.07	-
Ours	**2.38**	3.03	**3.54**	**2.98**	**2.86**

8.5 Results

In this section, I show our qualitative and quantitative results on AFLW2000-3D [510] and Florence [19] dataset in comparison with several other state-of-the-art methods. I then showed the results of an ablation study in order to demonstrate the effectiveness of the proposed aggregation block. The qualitative results of face alignment and 3D face reconstruction are shown in Fig. 8.2 and Fig. 8.5 (b) respectively.

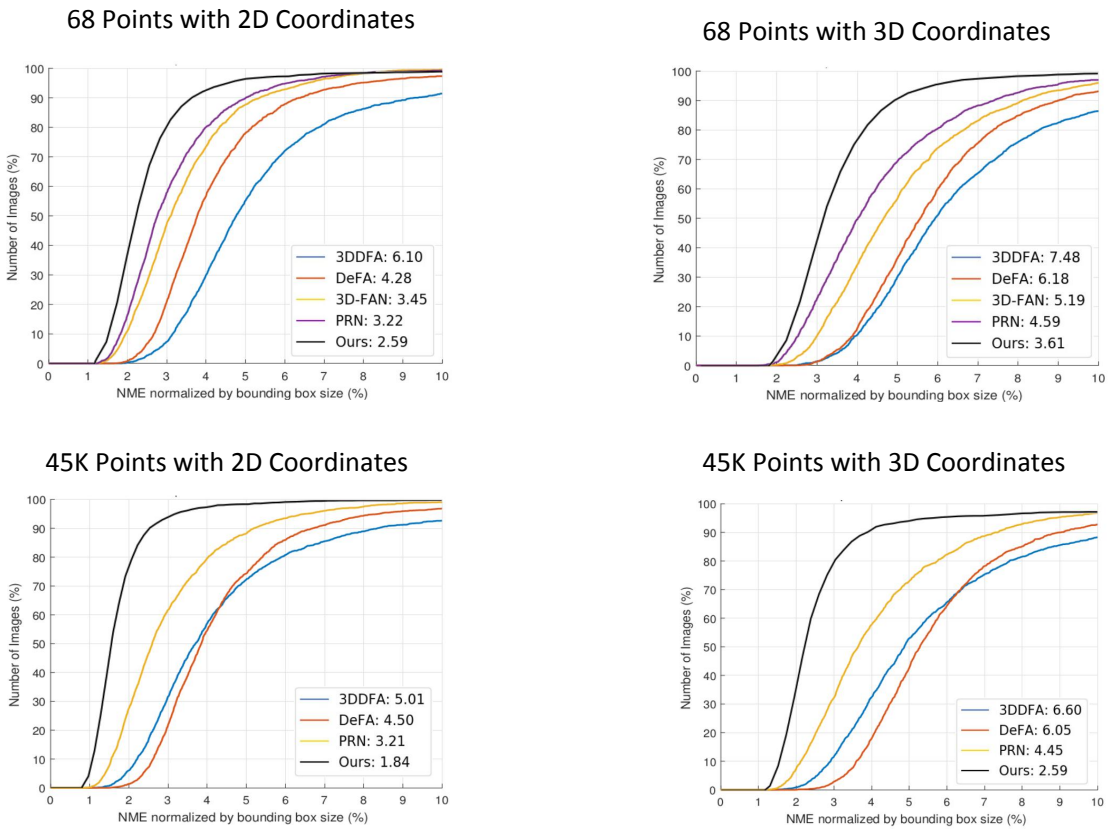


Figure 8.4: Errors Distribution (CED) curves for sparse and dense face alignment on AFLW2000-3D. Note that for dense face alignment, PRN [105] can only regress around 45K points, so I only select around 45K points for evaluation, even though our model can output all the 53215 vertices provided by the ground truth. Our model performs consistently better on both 2D and 3D problems when compared to other methods.

8.5.1 Face Alignment

I compare our model with other state-of-the-art methods, 3DDFA [510], DeFA [243], 3D-FAN [45], PRN [105], on sparse alignment tasks (68 landmarks). As suggested by 3DDFA [510], normalized mean error (NME) is used as the alignment accuracy metric. NME is the average of the landmarks error normalized by the size of the bounding box.

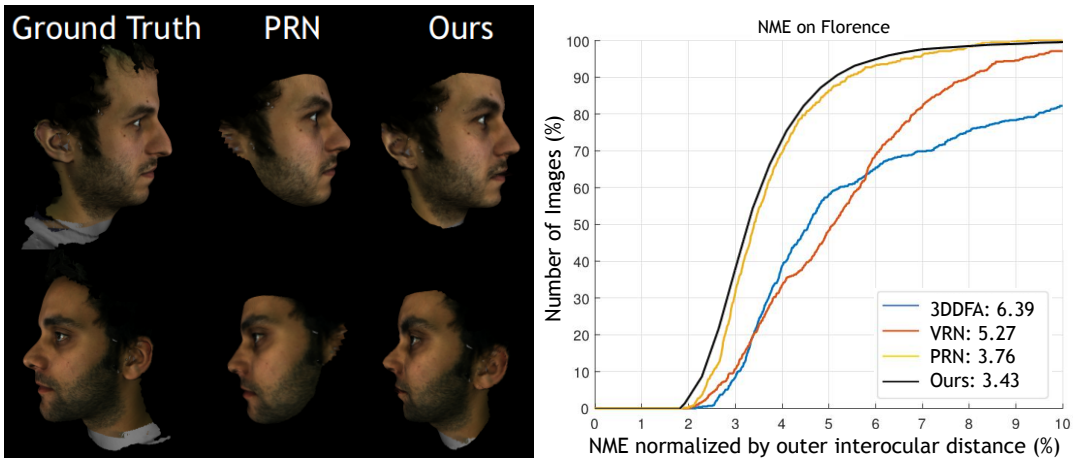


Figure 8.5: Example results on Florence dataset. (a): Qualitative results, First column are Ground truth [19]. The second column is Prediction by PRN [105]. The third column is Results from our model. Note that our model can faithfully reconstruct more regions such as ears. (b): Quantitative results, the normalized mean error of each method is showed in the legend.

The bounding box size is defined as the rectangle hull of all the 68 landmarks, which is $\sqrt{width * height}$. Fig. 8.4 (a) and (b) show the sparse face alignment with 68 landmarks on both 2D coordinate and 3D coordinate system. Our model exceeds other methods by a large margin on 3D face alignment. Specifically, more than 19 % relative higher performance is achieved compared with the best method on both 2D and 3D coordinates.

Our model also produces good performance in a dense face alignment task with 45K vertices. I compare with previous state-of-the-art methods, including 3DDFA [510], DeFA [243], PRN [105], and NME plots were shown in Fig. 8.4 (c) and (d), which demonstrate that our model gains more than 41% relative improvements compared to PRN [105], so our model can produce more accurate vertices localization results, with the help of an aggregation block to extract more useful information and GCNs to directly perform feature learning on 3D face mesh.

I further evaluate our model on sparse face alignment with different face poses in 2D

images in comparison with SDM [452], 3DDFA [510], 3DSTN [30], DeFA [243], PRN [105], N3DMM [390], CMD [507]. I randomly select 915 images from AFLW2000-3D to balance the distribution, whose absolute yaw angles with small, medium and large values are 1/3 each. Across three main classes with yaw values ($0° \sim 30°$, $30° \sim 60°$, $60° \sim 90°$) for the faces in different images, our model exceeds the other state-of-the-art methods. Especially for large pose face alignment ($60° \sim 90°$), as shown in Fig. 8.2 our model can handle large pose face well. Because of the invisible parts of the face due to occlusion, the other methods cannot capture enough semantic information to regress the landmarks. Our model, however, utilizes aggregate feature learned from the visible part of the face to infer the unseen part of the faces' landmarks, which fuse and reuse the 2D semantic information to regress the 3D geometric information. The results are shown in Tab. 8.1, where the numerical values of the other methods are cited from the original papers. As illustrated, our model achieves more than 25% relative improvement over the best method on AFLW2000-3D dataset.

8.5.2 3D Face Reconstruction

I illustrate our model's ability in a 3D face reconstruction task with experiments on the Florence dataset [19], compared with a state-of-the-art method, 3DDFA [510], PRN [105], VRN [161], following the experimental settings in PRN [105], and the metric which is the Mean Squared Error(MSE) normalized by outer interocular distance of 3D coordinates. I calculate the bounding box from the ground truth point cloud and crop the rendered image to 256×256, and I follow [161] to choose 19K points of face region for evaluation. Note that, during the training process, our model only considers the coordinates of vertices, but for better visualization, the colors of faces are rendered from the corresponding input 2D image. For our model, I render colours to each 3D face vertex from the corresponding 2D

input image pixels. Fig. 8.5 shows the qualitative and quantitative results, our model can handle large pose face well and accurately covers more regions in lateral face parts, such as ears and necks, but PRN remains blurry in the ear area, and for quantitative comparison, our model achieves superior performance to PRN and outperform the other two methods by a large margin.

8.6 Discussion and Conclusion

8.6.1 Ablation Study

Aggregation Block: In this section, I conduct several experiments to establish the effectiveness and compactness of our proposed aggregation block. I compare with no-aggregation block (encoder-decoder) and shallow-aggregation block structures (U-net). I change the decoder of those two networks into GCNs with graph up-sampling operations, but the encoder remains as CNNs. Apart from the aggregation part, the rest of the network maintain the same structure. Also, I remove the up-sampling and down-sampling operations, respectively, to further evaluate whether our aggregation block can help to better regress the face mesh vertices coordinates. Fig. 8.6 (a) and (b) show the quantitative results on a 3D face alignment task. As is illustrated, our aggregation model attains a superior performance over the other four methods, and non-aggregation network structure has the worst performance.

Parameters of Loss Function: Several experiments are conducted to evaluate the parameter setting of our proposed loss function. Fig. 8.6 (c) and (d) show the parameters setting results on sparse and dense alignments tasks, besides, our model is not sensitive to the two parameters, as no significant difference are found, and when $w = 5$, $\epsilon = 4$, our model achieve best results.

Loss Function: I compare with L1, L2, Smooth-L1 [119] loss functions, which are commonly used in the regression problem. Experiments are performed on sparse alignment (68 points) and dense alignment (45K points) in 3D coordinates. The performance is reported as average NME(%) of sparse and dense alignment tasks. Our proposed loss function (3.10 %) outperforms smooth-L1 loss (3.39 %) [119] by 9 % relatively better performance, L1 loss (3.61 %) by 14 % relatively better performance, and L2 Loss (4.02 %) by 23 % relatively better performance. Our proposed loss function attains a superior performance over the other three loss functions.

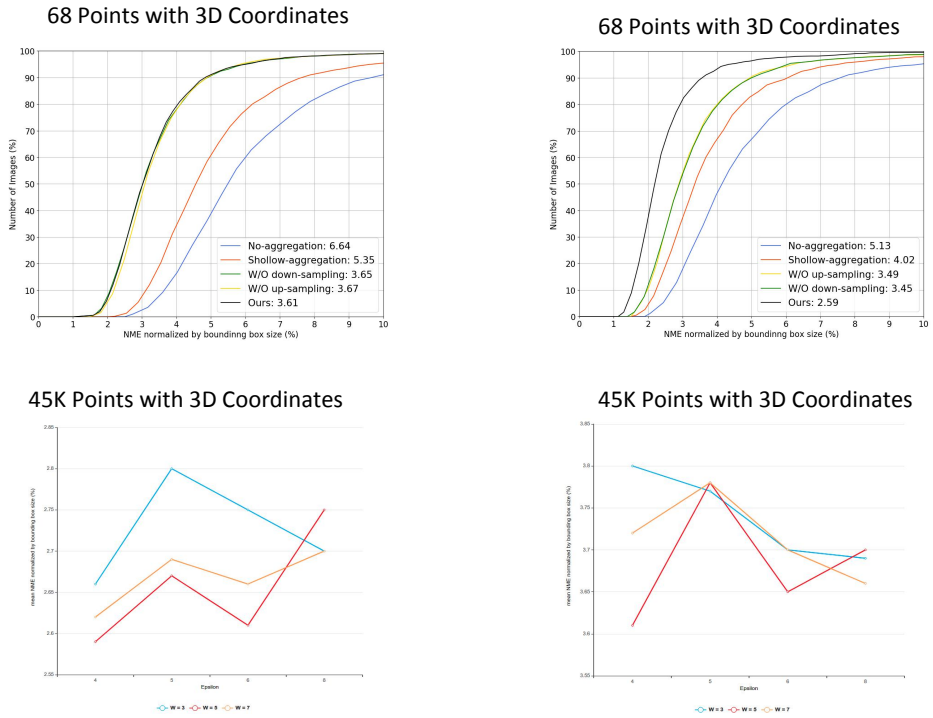


Figure 8.6: (a)&(b), Illustration of the influence of the aggregation block. (c)&(d), the parameter setting for the proposed loss function. Methods are evaluated on 3D face alignment with 68 landmarks 45K landmarks. Our aggregation model outperforms the other four methods, specifically more than 32% relative better performance is achieved over the non-aggregation method on both sparse and dense face alignment. And when W = 5, ϵ = 4, our model achieves best results.

8.6.2 Model Complexity and Running Speed

Even though our model structure looks complicated, in benefit from feature reuse, it is still relatively light-weight and efficient, taking up only 84.5MB compared to 1.5GB in VRN [161] and 153MB in PRN [105]. I use the same definition of running time, as suggested by PRN [105]. The running time of different models is reported in Tab. 8.2. Our model achieves comparable result with 16.0 milliseconds per image, and the hardware used for the evaluation is NVIDIA GeForce RTX 2080Ti GPU and Intel(R) Xeon(R) W-2104 CPU @ 3.20GHz. The results of 3DDFA [510], 3DSTN [30], CMD [507] are from their papers, while the running time of the other methods is obtained by running their publicly available source codes on the same machine as our model.

3DDFA [510]	3D-FAN [45]	PRN [105]	DeFA [243]	VRN [161]	CMD [507]	3DSTN [30]	Ours
75.7 ms	53.9 ms	9.7 ms	34.5 ms	68.5 ms	3.0 ms	19.0 ms	16.0 ms

Table 8.2: Running time per testing image

8.6.3 Conclusion

In this work, I propose a new end-to-end aggregation graph convolution network to improve the accuracy of dense face alignment and 3D face reconstruction simultaneously. Our network can regress the coordinates of 3D face mesh vertices directly by learning multi-level semantic and spatial features from a single 2D image. Qualitative and quantitative results confirm the effectiveness and efficiency of our model.

Chapter 9

Conclusion & Future Work

9.0.1 Summary

In this thesis, I have presented several works that explore the graph-structured representations on the task of biomedical and biometric image analysis. Apart from that, I addressed the several challenging questions that are overlooked or rarely discussed by previous studies, which promotes the development of the GNN in biomedical image analysis.

Specifically, in Chapter 3 and Chapter 4, I explore the geometric correlation and consistency between objects' region and boundary through implicit graph data representation learning; and propose different graph-based novel approaches to leverage complementary spatial relationships. I address the rarely discussed underlying relationship between the region and boundary characteristics in segmentation tasks and the semi-supervised learning paradigm. Such correlations of spatial consistency between the region and boundary features advanced the coherence of the network when tackling different tasks and mitigated the inevitable perturbations at the task level under semi-supervised learning mechanisms. My methods have achieved superior performance on biomedical image segmentation datasets such as five large-scale fundus image datasets for optic disc and cup segmentation in both

fully supervised and semi-supervised learning paradigms and five challenging datasets of colonoscopic endoscopy images for polys segmentation in the fully supervised mechanism.

In Chapter 5 and Chapter 6, I study the context pattern fusing of various forms of granularity information utilising inner-domain and cross-domain implicit graph data representation learning. I propose several graph-based novel methods for hybrid information fusion and address the contextual dependency difficulties of multi-granularity features during graph reasoning. The developed methods achieved excellent performance on the three largest COVID-19 diagnosis datasets and five challenging crowd counting datasets. Significantly, my proposed graph model can outperform other compared methods by a large margin in the generalisation ability evaluation experiments of the COVID-19 diagnosis task.

In Chapter 7, I propose to model the geometric structure of explicit graph data representations in terms of objects' boundaries. I introduce a novel graph-based segmentation paradigm and address the difficulties of direct features learning on objects' boundary locations by previous CNN based methods. I used the provided approaches to segment fundus-based optic discs and cups and ultrasound-based fetal heads. My model achieved comparable performance with other CNN-based methods but can directly indicate the boundary locations that show more interpretative prediction than dense-pixel wise classification methods. In Chapter 8, I research the explicit graph data representation learning of dense vertices regression task. I propose a multi-level aggregated GCN and address the challenges of loss of semantic and spatial information in classic GCN based methods. The proposed model was used in two sizeable 3D face reconstruction datasets, with outstanding results indicating its accuracy and capacity to handle many vertices. In conclusion, I have proposed several novel methods based on graph-based deep learning with explicit and implicit representations in different biomedical and biometric image analysis tasks. I have

demonstrated the robustness and generalisability of the aforementioned proposed methods in various biomedical and biometric image analysis tasks.

9.0.2 Future Work

The future work of this thesis can be extended in twofold.

Firstly, all of my approaches are anticipated to be widely applicable to real-world applications. However, all of the presented works in this thesis follow the same paradigm that exploits the benefit of explicit or implicit graph representations. At the same time, future works can combine the benefits of explicit and implicit graph representation learning and tackle more complicated problems in a graph structure, such as complex non-Euclidean geometry analysis tasks. Specifically, exploiting graph neural networks to study the task of protein analysis and drug discovery are attracting the researcher's attention recently, including target identification, hit identification, lead optimization, drug repositioning, *etc.* In general, the complete process for one authorized drug takes around 13.5 years, including 5.5 years before clinical trials (drug discovery) and eight years for the remainder of the procedure (drug development) [310]. Consequently, reducing the overall cost and time is a massive problem in industry and academia, and the modern medication R&D process may not be sustainable unless these obstacles are overcome. Due to the frequent elimination of medication candidates, the current pharmaceutical business expends enormous amounts of time and resources. According to recent statistics [397], 80% of the causes for attrition were attributed to poor pharmacokinetics (39%), lack of efficacy (30%), and animal toxicity (11%). Surprisingly, the aforementioned issues are strongly tied to the drug discovery process prior to clinical trials, indicating the possibility for improvement. In general, the overall process is determined by knowledge-based decisions, which can be highly biased, as it is virtually impossible to synthesize and evaluate all the possible compounds by

experiments. In this circumstance, AI-guided decision-making is a promising breakthrough [263, 463]. Notably, the graph-based network plays a vital role in such circumstances.

Secondly, validating the proposed models on a larger biomedical image dataset is appreciated and critical, especially for evaluating biomedical image analysis algorithms. There are, in total, six types of biomedical image analysis tasks with their corresponding data been included in this thesis, such as color fundus images of *OC & OD* segmentation, color fundus images of *vCDR* estimation, colonoscopy images of polyps segmentation, CT images of COVID-19 diagnosis, ultrasound images of fetal head segmentation. However, only 619 images are used as the test dataset for *OC & OD* segmentation task; 635 images as the test dataset for colonoscopy polyps segmentation task; and 94 images as the test dataset for fetal head segmentation task. Reporting results on such a relatively small size dataset will be affected by inevitable bias, and overfitting issues [334]. However, the limited data size of the AI-based biomedical image analysis task is still a challenge nowadays due to the difficulty of manual annotation efforts in clinics. It is unrealistic for clinicians to annotate a large number of images as it is costly, time-consuming, and labour-intensive. In the future, more efficient way of annotations, such as AI-assistant labeling processing, can address the difficulty and increase the data size.

Thirdly, the domain shift issues and the generalizability of the proposed algorithms need to be assessed using external test datasets and the associated experiments. The domain shift issue resulting from disparate distributions of source/reference data and target data causes the deep learning algorithms to perform poorly. Specifically, machine learning algorithms typically assume that the training dataset (source/reference domain) and test dataset (target domain) share the same data distribution [396]. Nonetheless, this assumption is overly strong and may not hold true in actual reality. Previous studies have revealed that the test error generally increases in proportion to the distribution difference between

training and test datasets [27, 388]. This is referred to as the well-known 'domain shift' problem [319]. Even in the deep learning era, deep neural networks trained on large-scale image datasets may still suffer from domain shift [93]. Thus, how to handle domain shift is a crucial issue to effectively apply the proposed methods to medical image analysis tasks.

Bibliography

[1] Andrea F Abate, Michele Nappi, Daniel Riccio, and Gabriele Sabatino. 2d and 3d face recognition: A survey. *Pattern recognition letters*, 28(14):1885–1906, 2007.

[2] Shahira Abousamra, Minh Hoai, Dimitris Samaras, and Chao Chen. Localization in the crowd with topological constraints. In *AAAI*, 2021.

[3] Shubhra Aich and Ian Stavness. Leaf counting with deep convolutional and deconvolutional networks. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 2080–2089, 2017.

[4] Oleg Alexander, Mike Rogers, William Lambeth, Jen-Yuan Chiang, Wan-Chun Ma, Chuan-Chang Wang, and Paul Debevec. The digital emily project: Achieving a photorealistic digital actor. *IEEE Computer Graphics and Applications*, 30(4):20–31, 2010.

[5] Paula A Alhadeff, C Gustavo De Moraes, Monica Chen, Ali S Raza, Robert Ritch, and Donald C Hood. The association between clinical features seen on fundus photographs and glaucomatous damage detected on visual fields and optical coherence tomography scans. *Journal of Glaucoma*, 26(5):498, 2017.

[6] Ahmed Almazroa, Sami Alodhayb, Essameldin Osman, Eslam Ramadan, Mohammed Hummadi, Mohammed Dlaim, Muhannad Alkatee, Kaamran Raahemifar, and Vasudevan Lakshminarayanan. Retinal fundus images for glaucoma analysis: the RIGA dataset. In *Medical Imaging 2018: Imaging Informatics for Healthcare, Research, and Applications*, volume 10579, page 105790B. International Society for Optics and Photonics, 2018.

[7] Riza Alp Guler, George Trigeorgis, Epameinondas Antonakos, Patrick Snape, Stefanos Zafeiriou, and Iasonas Kokkinos. Densereg: Fully convolutional dense shape regression in-the-wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6799–6808, 2017.

[8] Angela Ofeibea Amedo, Nana Yaa Koomson, Emmanuel Kobia Acquah, Tchiakpe Michel Pascal, Johnson Atuahene, Prince Kwaku Akowuah, Philip Tetteh Djeagbo, and Richard Baafi. Comparison of the clinical estimation of cup-to-disk ratio by direct ophthalmoscopy and optical coherence tomography. *Therapeutic Advances in Ophthalmology*, 11:2515841419827268, 2019.

[9] Jaume Amores. Multiple instance classification: Review, taxonomy and comparative study. *Artificial intelligence*, 201:81–105, 2013.

[10] Moussa Amrani, Abdelatif Bey, and Abdenour Amamra. New sar target recognition based on yolo and very deep multi-canonical correlation analysis. *International Journal of Remote Sensing*, pages 1–20, 2021.

[11] Moussa Amrani, Mohamed Hammad, Feng Jiang, Kuanquan Wang, and Amel Amrani. Very deep feature extraction and fusion for arrhythmias detection. *Neural Computing and Applications*, 30(7):2047–2057, 2018.

[12] Moussa Amrani and Feng Jiang. Deep feature extraction and combination for synthetic aperture radar target classification. *Journal of Applied Remote Sensing*, 11(4):042616, 2017.

[13] Moussa Amrani, Feng Jiang, Yunzhong Xu, Shaohui Liu, and Shengping Zhang. Sar-oriented visual saliency model and directed acyclic graph support vector metric based target classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(10):3794–3810, 2018.

[14] Moussa Amrani, Kai Yang, Dongyang Zhao, Xiaopeng Fan, and Feng Jiang. An efficient feature selection for sar target classification. In *Pacific Rim Conference on Multimedia*, pages 68–78. Springer, 2017.

[15] Anurag Arnab, Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Måns Larsson, Alexander Kirillov, Bogdan Savchynskyy, Carsten Rother, Fredrik Kahl, and Philip HS Torr. Conditional random fields meet deep neural networks for semantic segmentation: Combining probabilistic graphical models with deep learning for structured prediction. *IEEE Signal Processing Magazine*, 35(1):37–52, 2018.

[16] Carlos Arteta, Victor Lempitsky, and Andrew Zisserman. Counting in the wild. In *European conference on computer vision*, pages 483–498. Springer, 2016.

[17] James Atwood and Don Towsley. Diffusion-convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1993–2001, 2016.

[18] Angelica I Aviles-Rivero, Philip Sellars, Carola-Bibiane Schönlieb, and Nicolas Papadakis. Graphxcovid: explainable deep graph diffusion pseudo-labelling for identifying covid-19 on chest x-rays. *Pattern Recognition*, 122:108274, 2022.

[19] Andrew D Bagdanov, Alberto Del Bimbo, and Iacopo Masi. The florence 2d/3d hybrid face dataset. In *Proceedings of the 2011 joint ACM workshop on Human gesture and behavior understanding*, pages 79–80. ACM, 2011.

[20] Harrison X Bai, Ben Hsieh, Zeng Xiong, Kasey Halsey, Ji Whae Choi, Thi My Linh Tran, Ian Pan, Lin-Bo Shi, Dong-Cui Wang, Ji Mei, et al. Performance of radiologists in differentiating COVID-19 from viral pneumonia on chest CT. *Radiology*, 2020.

[21] Harrison X Bai, Robin Wang, Zeng Xiong, Ben Hsieh, Ken Chang, Kasey Halsey, Thi My Linh Tran, Ji Whae Choi, Dong-Cui Wang, Lin-Bo Shi, et al. Artificial intelligence augmentation of radiologist performance in distinguishing covid-19 from pneumonia of other origin at chest ct. *Radiology*, 296(3):E156–E165, 2020.

[22] Shuai Bai, Zhiqun He, Yu Qiao, Hanzhe Hu, Wei Wu, and Junjie Yan. Adaptive dilated network with self-correction supervision for counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4594–4603, 2020.

[23] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016.

[24] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[25] Thabo Beeler, Bernd Bickel, Paul Beardsley, Bob Sumner, and Markus Gross. High-quality single-shot capture of facial geometry. In *ACM SIGGRAPH 2010 papers*, pages 1–9. 2010.

[26] Thabo Beeler, Fabian Hahn, Derek Bradley, Bernd Bickel, Paul Beardsley, Craig Gotsman, Robert W Sumner, and Markus Gross. High-quality passive facial performance capture using anchor frames. In *ACM SIGGRAPH 2011 papers*, pages 1–10. 2011.

[27] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19, 2006.

[28] Jorge Bernal, F Javier Sánchez, Gloria Fernández-Esparrach, Debora Gil, Cristina Rodríguez, and Fernando Vilariño. WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized Medical Imaging and Graphics*, 43:99–111, 2015.

[29] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[30] Chandrasekhar Bhagavatula, Chenchen Zhu, Khoa Luu, and Marios Savvides. Faster than real-time facial alignment: A 3d spatial transformer network approach in unconstrained poses. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3980–3989, 2017.

[31] J Martin Bland and Douglas G Altman. Measuring agreement in method comparison studies. *Statistical Methods in Medical Research*, 8(2):135–160, 1999.

[32] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

[33] James Booth, Epameinondas Antonakos, Stylianos Ploumpis, George Trigeorgis, Yannis Panagakis, and Stefanos Zafeiriou. 3d face morphable models" in-the-wild". In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5464–5473. IEEE, 2017.

[34] James Booth, Anastasios Roussos, Allan Ponniah, David Dunaway, and Stefanos Zafeiriou. Large scale 3d morphable models. *International Journal of Computer Vision*, 126(2-4):233–254, 2018.

[35] James Booth, Anastasios Roussos, Evangelos Ververas, Epameinondas Antonakos, Stylianos Ploumpis, Yannis Panagakis, and Stefanos Zafeiriou. 3d reconstruction of "in-the-wild" faces in images and videos. *IEEE transactions on pattern analysis and machine intelligence*, 40(11):2638–2652, 2018.

[36] Christoph Börgers and Frank Natterer. *Computational radiology and imaging: therapy and diagnostics*, volume 110. Springer Science & Business Media, 1999.

[37] Giorgos Bouritsas, Sergiy Bokhnyak, Stylianos Ploumpis, Michael Bronstein, and Stefanos Zafeiriou. Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7213–7222, 2019.

[38] Joshua Bridge, Yanda Meng, Yitian Zhao, Yong Du, Mingfeng Zhao, Renrong Sun, and Yalin Zheng. Introducing the gev activation function for highly unbalanced

data to develop covid-19 diagnostic models. *IEEE journal of Biomedical and Health Informatics*, 24(10):2776–2786, 2020.

[39] Joshua Bridge, Yanda Meng, Wenyue Zhu, Thomas Fitzmaurice, Caroline McCann, Cliff Addison, Manhui Wang, Cristin Merritt, Stu Franks, Maria Mackey, et al. Development and external validation of a mixed-effects deep learning model to diagnose covid-19 from ct imaging. *medRxiv*, 2022.

[40] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[41] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[42] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*, pages http–openreview, 2014.

[43] Alan Brunton, Timo Bolkart, and Stefanie Wuhrer. Multilinear wavelets: A statistical shape space for human faces. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014.

[44] Adrian Bulat and Georgios Tzimiropoulos. Binarized convolutional landmark localizers for human pose estimation and face alignment with limited resources. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3706–3714, 2017.

[45] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In

Proceedings of the IEEE International Conference on Computer Vision, pages 1021–1030, 2017.

[46] A Mike Burton, Stephen Wilson, Michelle Cowan, and Vicki Bruce. Face recognition in poor-quality video: Evidence from security surveillance. *Psychological Science*, 10(3):243–248, 1999.

[47] Saul Calderon-Ramirez, Shengxiang Yang, Armaghan Moemeni, Simon Colreavy-Donnelly, David A Elizondo, Luis Oala, Jorge Rodríguez-Capitán, Manuel Jiménez-Navarro, Ezequiel López-Rubio, and Miguel A Molina-Cabello. Improving uncertainty estimation with semi-supervised deep learning for covid-19 detection using chest x-ray images. *IEEE Access*, 2021.

[48] Gabriele Campanella, Matthew G Hanna, Luke Geneslaw, Allen Miraflor, Vitor Werneck Krauss Silva, Klaus J Busam, Edi Brogi, Victor E Reuter, David S Klimstra, and Thomas J Fuchs. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature medicine*, 25(8):1301–1309, 2019.

[49] Chen Cao, Yanlin Weng, Stephen Lin, and Kun Zhou. 3d shape regression for real-time facial animation. *ACM Transactions on Graphics (TOG)*, 32(4):1–10, 2013.

[50] Xinkun Cao, Zhipeng Wang, Yanyun Zhao, and Fei Su. Scale aggregation network for accurate and efficient crowd counting. In *Proceedings of the European conference on computer vision (ECCV)*, pages 734–750, 2018.

[51] Juan J Cerrolaza, Antonio R Porras, Awais Mansoor, Qian Zhao, Marshall Summar, and Marius George Linguraru. Identification of dysmorphic syndromes using landmark-specific local texture descriptors. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 1080–1083. IEEE, 2016.

[52] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–7. IEEE, 2008.

[53] Antoni B Chan and Nuno Vasconcelos. Bayesian poisson regression for crowd counting. In *2009 IEEE 12th international conference on computer vision*, pages 545–551. IEEE, 2009.

[54] Hanqing Chao, Xi Fang, Jiajin Zhang, Fatemeh Homayounieh, Chiara D Arru, Subba R Digumarthy, Rosa Babaei, Hadi K Mobin, Iman Mohseni, Luca Saba, et al. Integrative analysis for covid-19 patient outcome prediction. *Medical Image Analysis*, 67:101844, 2021.

[55] Binghui Chen, Zhaoyi Yan, Ke Li, Pengyu Li, Biao Wang, Wangmeng Zuo, and Lei Zhang. Variational attention: Propagating domain-specific knowledge for multi-domain learning in crowd counting. *The IEEE International Conference on Computer Vision (ICCV)*, 2021.

[56] Bingzhi Chen, Jinxing Li, Guangming Lu, Hongbing Yu, and David Zhang. Label co-occurrence learning with graph convolutional networks for multi-label chest x-ray image classification. *IEEE journal of biomedical and health informatics*, 24(8):2292–2302, 2020.

[57] Chen Chen, Chen Qin, Huaqi Qiu, Giacomo Tarroni, Jinming Duan, Wenjia Bai, and Daniel Rueckert. Deep learning for cardiac image segmentation: a review. *Frontiers in Cardiovascular Medicine*, page 25, 2020.

[58] Ke Chen, Chen Change Loy, Shaogang Gong, and Tony Xiang. Feature mining for localised crowd counting. In *Bmvc*, volume 1, page 3, 2012.

[59] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.

[60] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[61] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018.

[62] Nanshan Chen, Min Zhou, Xuan Dong, Jieming Qu, Fengyun Gong, Yang Han, Yang Qiu, Jingli Wang, Ying Liu, Yuan Wei, et al. Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in wuhan, china: a descriptive study. *The lancet*, 395(10223):507–513, 2020.

[63] Wenan Chen, Rebecca Smith, Soo-Yeon Ji, Kevin R Ward, and Kayvan Najarian. Automated ventricular systems segmentation in brain ct images by combining low-level segmentation and high-level template matching. *BMC medical informatics and decision making*, 9(1):1–14, 2009.

[64] Xiang Chen, Nishant Ravikumar, Yan Xia, Rahman Attar, Andres Diaz-Pinto, Stefan K Piechnik, Stefan Neubauer, Steffen E Petersen, and Alejandro F Frangi. Shape registration with learned deformations for 3d shape reconstruction from sparse and incomplete point clouds. *Medical Image Analysis*, page 102228, 2021.

[65] Xinya Chen, Yanrui Bin, Nong Sang, and Changxin Gao. Scale pyramid network for crowd counting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1941–1950. IEEE, 2019.

[66] Xu Chen, Xiangde Luo, Guotai Wang, and Yalin Zheng. Deep elastica for image segmentation. In *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pages 706–710, 2021.

[67] Xu Chen, Yanda Meng, Yitian Zhao, Rachel Williams, Srinivasa R Vallabhaneni, and Yalin Zheng. Learning unsupervised parameter-specific affine transformation for medical images registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 24–34. Springer, 2021.

[68] Xu Chen, Bryan M Williams, Srinivasa R Vallabhaneni, Gabriela Czanner, Rachel Williams, and Yalin Zheng. Learning active contour models for medical image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11632–11640, 2019.

[69] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Multi-fiber networks for video recognition. In *Proceedings of the european conference on computer vision (ECCV)*, pages 352–367, 2018.

[70] Yunpeng Chen, Marcus Rohrbach, Zhicheng Yan, Yan Shuicheng, Jiashi Feng, and Yannis Kalantidis. Graph-based global reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 433–442, 2019.

[71] Zhengdao Chen, Soledad Villar, Lei Chen, and Joan Bruna. On the equivalence between graph isomorphism testing and function approximation with gnns. *Advances in neural information processing systems*, 32, 2019.

[72] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary IoU: Improving object-centric image segmentation evaluation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

[73] Dominic Cheng, Renjie Liao, Sanja Fidler, and Raquel Urtasun. DARNet: Deep active ray network for building segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7431–7439, 2019.

[74] Jian Cheng, Haipeng Xiong, Zhiguo Cao, and Hao Lu. Decoupled two-stage crowd counting and beyond. *IEEE Transactions on Image Processing*, 30:2862–2875, 2021.

[75] Shiyang Cheng, Michael Bronstein, Yuxiang Zhou, Irene Kotsia, Maja Pantic, and Stefanos Zafeiriou. Meshgan: Non-linear 3d morphable models of faces. *arXiv preprint arXiv:1903.10384*, 2019.

[76] Zhi-Qi Cheng, Qi Dai, Hong Li, Jingkuan Song, Xiao Wu, and Alexander G Hauptmann. Rethinking spatial invariance of convolutional networks for object counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19638–19648, 2022.

[77] Philip Chikontwe, Miguel Luna, Myeongkyun Kang, Kyung Soo Hong, June Hong Ahn, and Sang Hyun Park. Dual attention multiple instance learning with unsupervised complementary loss for covid-19 screening. *Medical Image Analysis*, page 102105, 2021.

[78] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[79] Fan RK Chung and Fan Chung Graham. *Spectral graph theory*. Number 92. American Mathematical Soc., 1997.

[80] Ciprian Adrian Corneanu, Marc Oliu Simón, Jeffrey F Cohn, and Sergio Escalera Guerrero. Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications. *IEEE transactions on pattern analysis and machine intelligence*, 38(8):1548–1568, 2016.

[81] Gabriela Csurka, Diane Larlus, Florent Perronnin, and France Meylan. What is a good evaluation measure for semantic segmentation?. In *BMVC*, volume 27, pages 10–5244, 2013.

[82] Dragos M Cvetkovic et al. Spectra of graphs. theory and application. 1980.

[83] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[84] Etienne Decencière, Xiwei Zhang, Guy Cazuguel, Bruno Lay, Béatrice Cochener, Caroline Trone, Philippe Gain, Richard Ordonez, Pascale Massin, Ali Erginay, et al. Feedback on a publicly distributed image database: the messidor database. *Image Analysis & Stereology*, 33(3):231–234, 2014.

[85] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[86] Elizabeth R DeLong, David M DeLong, and Daniel L Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, pages 837–845, 1988.

[87] Jiankang Deng, Shiyang Cheng, Niannan Xue, Yuxiang Zhou, and Stefanos Zafeiriou. Uv-gan: Adversarial facial uv map completion for pose-invariant face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7093–7102, 2018.

[88] Kaizhong Deng, Yanda Meng, Dongxu Gao, Joshua Bridge, Yaochun Shen, Gregory Lip, Yitian Zhao, and Yalin Zheng. Transbridge: A lightweight transformer for left ventricle segmentation in echocardiography. In *International Workshop on Advances in Simplifying Medical Ultrasound*, pages 63–72. Springer, 2021.

[89] Dmytro Derkach, Adria Ruiz, and Federico M Sukno. Head pose estimation based on 3-d facial landmarks localization and regression. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 820–827. IEEE, 2017.

[90] Donglin Di, Feng Shi, Fuhua Yan, Liming Xia, Zhanhao Mo, Zhongxiang Ding, Fei Shan, Bin Song, Shengrui Li, Ying Wei, et al. Hypergraph learning for identification of covid-19 with ct imaging. *Medical Image Analysis*, 68:101910, 2021.

[91] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1-2):31–71, 1997.

[92] Xinghao Ding, Fujin He, Zhirui Lin, Yu Wang, Huimin Guo, and Yue Huang. Crowd density estimation using fusion of multi-layer features. *IEEE Transactions on Intelligent Transportation Systems*, 22(8):4776–4787, 2020.

[93] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655. PMLR, 2014.

[94] Xingping Dong, Jianbing Shen, Ling Shao, and Luc Van Gool. Sub-Markov random walk for image segmentation. *IEEE Transactions on Image Processing*, 25(2):516–527, 2015.

[95] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

[96] Pengfei Dou, Shishir K Shah, and Ioannis A Kakadiaris. liu2018disentangling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5908–5917, 2017.

[97] Haoran Duan, Shidong Wang, and Yu Guan. Sofa-net: Second-order and first-order attention network for crowd counting. *BMVC*, 2020.

[98] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016.

[99] Joan Bruna Estrach, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and deep locally connected networks on graphs. In *2nd international conference on learning representations, ICLR*, volume 2014, 2014.

[100] Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Pranet: Parallel reverse attention network for polyp segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 263–273. Springer, 2020.

[101] Deng-Ping Fan, Tao Zhou, Ge-Peng Ji, Yi Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Inf-net: Automatic covid-19 lung infection segmentation from ct images. *IEEE Transactions on Medical Imaging*, 39(8):2626–2637, 2020.

[102] Yuqi Fan, Jiahao Liu, Ruixuan Yao, and Xiaohui Yuan. Covid-19 detection from x-ray images using multi-kernel-size spatial-channel attention network. *Pattern Recognition*, page 108055, 2021.

[103] Cong Fang, Song Bai, Qianlan Chen, Yu Zhou, Liming Xia, Lixin Qin, Shi Gong, Xudong Xie, Chunhua Zhou, Dandan Tu, et al. Deep learning for predicting covid-19 malignant progression. *Medical image analysis*, 72:102096, 2021.

[104] Yuqi Fang, Cheng Chen, Yixuan Yuan, and Kai-Yu Tong. Selective feature aggregation network with area-boundary constraints for polyp segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 302–310. Springer, 2019.

[105] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 534–551, 2018.

[106] Zhen-Hua Feng, Josef Kittler, Muhammad Awais, Patrik Huber, and Xiao-Jun Wu. Wing loss for robust facial landmark localisation with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2235–2245, 2018.

[107] Claudio Ferrari, Giuseppe Lisanti, Stefano Berretti, and Alberto Del Bimbo. A dictionary learning-based 3d morphable shape model. *IEEE Transactions on Multimedia*, 19(12):2666–2679, 2017.

[108] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. Pearson,, 2012.

[109] Huazhu Fu, Jun Cheng, Yanwu Xu, Damon Wing Kee Wong, Jiang Liu, and Xiaochun Cao. Joint optic disc and cup segmentation based on multi-label deep network and polar transformation. *IEEE Transactions on Medical Imaging*, 37(7):1597–1605, 2018.

[110] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[111] Francisco Fumero, Silvia Alayón, José L Sanchez, Jose Sigut, and M Gonzalez-Hernandez. RIM-ONE: An open retinal image database for optic nerve evaluation. In *24th International Symposium on Computer-based Medical Systems (CBMS)*, pages 1–6. IEEE, 2011.

[112] Guangshuai Gao, Junyu Gao, Qingjie Liu, Qi Wang, and Yunhong Wang. CNN-based density estimation and crowd counting: A survey. *arXiv preprint arXiv:2003.12783*, 2020.

[113] Kai Gao, Jianpo Su, Zhongbiao Jiang, Ling-Li Zeng, Zhichao Feng, Hui Shen, Pengfei Rong, Xin Xu, Jian Qin, Yuexiang Yang, et al. Dual-branch combination network (DCN): Towards accurate diagnosis and lesion segmentation of covid-19 using ct images. *Medical image analysis*, 67:101836, 2021.

[114] Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr. Res2net: A new multi-scale backbone architecture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[115] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pages 209–216. ACM Press/Addison-Wesley Publishing Co., 1997.

[116] Baris Gecer, Binod Bhattarai, Josef Kittler, and Tae-Kyun Kim. Semi-supervised adversarial learning to generate photorealistic face images of new identities from 3d morphable model. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–234, 2018.

[117] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1155–1164, 2019.

[118] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pages 1263–1272. PMLR, 2017.

[119] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[120] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.

[121] Chris Godsil and Gordon F Royle. *Algebraic graph theory*, volume 207. Springer Science & Business Media, 2001.

[122] Mikhail Goncharov, Maxim Pisov, Alexey Shevtsov, Boris Shirokikh, Anvar Kurmukov, Ivan Blokhin, Valeria Chernina, Alexander Solovev, Victor Gombolevskiy, Sergey Morozov, et al. Ct-based covid-19 triage: deep multitask learning improves joint identification and severity quantification. *Medical image analysis*, 71:102054, 2021.

[123] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE international joint conference on neural networks*, volume 2, pages 729–734, 2005.

[124] Hayit Greenspan, Raúl San José Estépar, Wiro J Niessen, Eliot Siegel, and Mads Nielsen. Position paper on covid-19 imaging and ai: From the clinical needs and technological challenges to initial ai solutions at the lab and national level towards a new era for ai in healthcare. *Medical image analysis*, 66:101800, 2020.

[125] Zaiwang Gu, Jun Cheng, Huazhu Fu, Kang Zhou, Huaying Hao, Yitian Zhao, Tianyang Zhang, Shenghua Gao, and Jiang Liu. CE-Net: Context encoder net-

work for 2d medical image segmentation. *IEEE Transactions on Medical Imaging*, 38(10):2281–2292, 2019.

[126] Valerio Guarrasi, Natascha Claudia D'Amico, Rosa Sicilia, Ermanno Cordelli, and Paolo Soda. Pareto optimization of deep networks for covid-19 diagnosis from chest x-rays. *Pattern Recognition*, 121:108242, 2022.

[127] Ricardo Guerrero-Gómez-Olmedo, Beatriz Torre-Jiménez, Roberto López-Sastre, Saturnino Maldonado-Bascón, and Daniel Onoro-Rubio. Extremely overlapping vehicle counting. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 423–431. Springer, 2015.

[128] Jia Guo, Jiankang Deng, Niannan Xue, and Stefanos Zafeiriou. Stacked dense u-nets with dual transformers for robust face alignment. *arXiv preprint arXiv:1812.01936*, 2018.

[129] Lei Guo, Li Tang, Tong Chen, Lei Zhu, Quoc Viet Hung Nguyen, and Hongzhi Yin. Da-gcn: A domain-aware attentive graph convolution network for shared-account cross-domain sequential recommendation. *The 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.

[130] Yudong Guo, Jianfei Cai, Boyi Jiang, Jianmin Zheng, et al. Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. *IEEE transactions on pattern analysis and machine intelligence*, 41(6):1294–1307, 2018.

[131] Yue Guo, Jason Stein, Guorong Wu, and Ashok Krishnamurthy. Sau-net: A universal deep network for cell counting. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 299–306, 2019.

[132] Shir Gur, Tal Shaharabany, and Lior Wolf. End to end trainable active contours via differentiable rendering. *arXiv preprint arXiv:1912.00367*, 2019.

[133] Shir Gur, Lior Wolf, Lior Golgher, and Pablo Blinder. Unsupervised microvascular image segmentation using an active contours mimicking neural network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10722–10731, 2019.

[134] Muhammad Salman Haleem, Liangxiu Han, Jano Van Hemert, and Baihua Li. Automatic extraction of retinal features from colour retinal images for glaucoma diagnosis: a review. *Computerized Medical Imaging and Graphics*, 37(7-8):581–596, 2013.

[135] Zhongyi Han, Benzheng Wei, Yanfei Hong, Tianyang Li, Jinyu Cong, Xue Zhu, Haifeng Wei, and Wei Zhang. Accurate screening of covid-19 using attention-based deep 3d multiple instance learning. *IEEE Transactions on Medical Imaging*, 39(8):2584–2594, 2020.

[136] Chahinez Hani, Nghi HOANG Trieu, Inès Saab, Séverine Dangeard, Souhail Bennani, Guillaume Chassagnon, and M-P Revel. COVID-19 pneumonia: a review of typical CT findings and differential diagnosis. *Diagnostic and interventional imaging*, 101(5):263–268, 2020.

[137] Dan Witzner Hansen and Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):478–500, 2009.

[138] Jinkui Hao, Jiang Liu, Ella Pereira, Ri Liu, Jiong Zhang, Yangfan Zhang, Kun Yan, Yan Gong, Jianjun Zheng, Jingfeng Zhang, et al. Uncertainty-guided graph attention

network for parapneumonic effusion diagnosis. *Medical Image Analysis*, page 102217, 2021.

[139] Robert M Haralick, Karthikeyan Shanmugam, and Its' Hak Dinstein. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[140] Hassan Hashemi, Reza Pakzad, Mehdi Khabazkhoob, Mohammad Hassan Emamian, Abbasali Yekta, and Akbar Fotouhi. The distribution of vertical cup-to-disc ratio and its determinants in the iranian adult population. *Journal of Current Ophthalmology*, 32(3):226, 2020.

[141] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[142] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, 2015.

[143] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[144] Kelei He, Wei Zhao, Xingzhi Xie, Wen Ji, Mingxia Liu, Zhenyu Tang, Yinghuan Shi, Feng Shi, Yang Gao, Jun Liu, et al. Synergistic learning of lung lobe segmentation and hierarchical multi-instance classification for automated severity assessment of covid-19 in ct images. *Pattern recognition*, 113:107828, 2021.

[145] Xin He, Shihao Wang, Xiaowen Chu, Shaohuai Shi, Jiangping Tang, Xin Liu, Chenggang Yan, Jiyong Zhang, and Guiguang Ding. Automated model design and benchmarking of deep learning models for covid-19 detection with chest ct scans. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4821–4829, 2021.

[146] Karsten Held, E Rota Kops, Bernd J Krause, William M Wells, Ron Kikinis, and H-W Muller-Gartner. Markov random field segmentation of brain mr images. *IEEE transactions on medical imaging*, 16(6):878–886, 1997.

[147] Mikael Henaff, Joan Bruna, and Yann LeCun. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*, 2015.

[148] Robin J Hennessy, Patrizia A Baldwin, David J Browne, Anthony Kinsella, and John L Waddington. Three-dimensional laser surface imaging and geometric morphometrics resolve frontonasal dysmorphology in schizophrenia. *Biological psychiatry*, 61(10):1187–1194, 2007.

[149] Junlin Hou, Jilan Xu, Longquan Jiang, Shanshan Du, Rui Feng, Yuejie Zhang, Fei Shan, and Xiangyang Xue. Periphery-aware covid-19 diagnosis with contrastive representation enhancement. *Pattern Recognition*, 118:108005, 2021.

[150] Le Hou, Dimitris Samaras, Tahsin M Kurc, Yi Gao, James E Davis, and Joel H Saltz. Patch-based convolutional neural network for whole slide tissue image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2424–2433, 2016.

[151] Ying-Xiang Hu, Rui-Sheng Jia, Yan-Bo Liu, Yong-Chao Li, and Hong-Mei Sun. Wsnet: A local–global consistent traffic density estimation method based on weakly supervised learning. *Knowledge-Based Systems*, page 109727, 2022.

[152] Chaolin Huang, Yeming Wang, Xingwang Li, Lili Ren, Jianping Zhao, Yi Hu, Li Zhang, Guohui Fan, Jiuyang Xu, Xiaoying Gu, et al. Clinical features of patients infected with 2019 novel coronavirus in wuhan, china. *The lancet*, 395(10223):497–506, 2020.

[153] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[154] Huimin Huang, Lanfen Lin, Yue Zhang, Yingying Xu, Jing Zheng, XiongWei Mao, Xiaohan Qian, Zhiyi Peng, Jianying Zhou, Yen-Wei Chen, et al. Graph-bas3net: Boundary-aware semi-supervised segmentation network with bilateral graph convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7386–7395, 2021.

[155] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2547–2554, 2013.

[156] Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah. Composition loss for counting, density map estimation and localization in dense crowds. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 532–546, 2018.

[157] Yoko Ikeda, Kazuhiko Mori, Morio Ueno, Yuko Maruyama, Kengo Yoshii, Junji Hamuro, Chie Sotozono, and Shigeru Kinoshita. Ten-year of glaucoma transition rate on the basis of optic nerve morphology in normal japanese subjects. *Investigative Ophthalmology & Visual Science*, 60(9):1968–1968, 2019.

[158] Maximilian Ilse, Jakub Tomczak, and Max Welling. Attention-based deep multiple instance learning. In *International conference on machine learning*, pages 2127–2136. PMLR, 2018.

[159] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[160] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 448–456, 2015.

[161] Aaron S Jackson, Adrian Bulat, Vasileios Argyriou, and Georgios Tzimiropoulos. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1031–1039, 2017.

[162] László A Jeni, András Lőrincz, Zoltán Szabó, Jeffrey F Cohn, and Takeo Kanade. Spatio-temporal event classification using time-series kernel based structured sparsity. In *European Conference on Computer Vision*, pages 135–150. Springer, 2014.

[163] Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *International Conference on Multimedia Modeling*, pages 451–462. Springer, 2020.

[164] Ge-Peng Ji, Yu-Cheng Chou, Deng-Ping Fan, Geng Chen, Huazhu Fu, Debesh Jha, and Ling Shao. Progressively normalized self-attention network for video polyp segmentation. *arXiv preprint arXiv:2105.08468*, 2021.

[165] Wei Ji, Shuang Yu, Junde Wu, Kai Ma, Cheng Bian, Qi Bi, Jingjing Li, Hanruo Liu, Li Cheng, and Yefeng Zheng. Learning calibrated medical image segmentation via multi-rater agreement modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12341–12351, 2021.

[166] Xiaoheng Jiang, Li Zhang, Mingliang Xu, Tianzhu Zhang, Pei Lv, Bing Zhou, Xin Yang, and Yanwei Pang. Attention scaling for crowd counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4706–4715, 2020.

[167] Xiaoheng Jiang, Li Zhang, Tianzhu Zhang, Pei Lv, Bing Zhou, Yanwei Pang, Mingliang Xu, and Changsheng Xu. Density-aware multi-task learning for crowd counting. *IEEE Transactions on Multimedia*, 23:443–453, 2020.

[168] Zi-Hang Jiang, Qianyi Wu, Keyu Chen, and Juyong Zhang. Disentangled representation learning for 3d face shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11957–11966, 2019.

[169] Hai Jin, Xun Wang, Zichun Zhong, and Jing Hua. Robust 3d face modeling and reconstruction from frontal and side images. *Computer Aided Geometric Design*, 50:1–13, 2017.

[170] Amin Jourabloo and Xiaoming Liu. Pose-invariant 3d face alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3694–3702, 2015.

[171] Amin Jourabloo and Xiaoming Liu. Large-pose face alignment via cnn-based dense 3d model fitting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4188–4196, 2016.

[172] Sinan Kaplan, Mehmet Amac Guvensan, Ali Gokhan Yavuz, and Yasin Karalurt. Driver behavior analysis for safe driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3017–3032, 2015.

[173] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[174] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems*, pages 5574–5584, 2017.

[175] Hoel Kervadec, Jihene Bouchtiba, Christian Desrosiers, Eric Granger, Jose Dolz, and Ismail Ben Ayed. Boundary loss for highly unbalanced segmentation. In *International Conference on Medical Imaging with Deep Learning*, pages 285–296. PMLR, 2019.

[176] Salman Khan, Munawar Hayat, Syed Waqas Zamir, Jianbing Shen, and Ling Shao. Striking the right balance with uncertainty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 103–112, 2019.

[177] Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *sensors*, 12(2):1437–1454, 2012.

[178] Aleksandr Kim, Aljoša Ošep, and Laura Leal-Taixé. Eagermot: 3d multi-object tracking via sensor fusion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11315–11321. IEEE, 2021.

[179] Eun A Kim, Kyung Soo Lee, Steven L Primack, Hye Kyung Yoon, Hong Sik Byun, Tae Sung Kim, Gee Young Suh, O Jung Kwon, and Joungho Han. Viral pneumonias in adults: radiologic and pathologic findings. *Radiographics*, 22(suppl_1):S137–S149, 2002.

[180] Si-Ho Kim, Yu Mi Wi, Sujin Lim, Kil-Tae Han, and In-Gyu Bae. Differences in clinical characteristics and chest images between coronavirus disease 2019 and influenza-associated pneumonia. *Diagnostics*, 11(2):261, 2021.

[181] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[182] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ICLR*, 2017.

[183] KJ Kiser, S Ahmed, SM Stieb, ASR Mohamed, H Elhalawani, PYS Park, NS Doyle, BJ Wang, A Barman, CD Fuller, et al. Data from the thoracic volume and pleural effusion segmentations in diseased lungs for benchmarking chest ct processing pipelines [dataset]. *The Cancer Imaging Archive*, 10, 2020.

[184] Martin Koestinger, Paul Wohlhart, Peter M Roth, and Horst Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 2144–2151. IEEE, 2011.

[185] Pushmeet Kohli, Philip HS Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009.

[186] Fanwei Kong, Nathan Wilson, and Shawn C Shadden. A deep-learning approach for direct whole-heart mesh reconstruction. *Medical Image Analysis*, 2021.

[187] Hyun Jung Koo, Soyeoun Lim, Jooae Choe, Sang-Ho Choi, Heungsup Sung, Kyung-Hyun Do, et al. Radiographic and CT features of viral pneumonia. *Radiographics*, 38(3):719–739, 2018.

[188] Oren Z Kraus, Jimmy Lei Ba, and Brendan J Frey. Classifying and segmenting microscopy images with deep multiple instance learning. *Bioinformatics*, 32(12):i52–i59, 2016.

[189] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[190] Aayush Kumar, Ayush R Tripathi, Suresh Chandra Satapathy, and Yu-Dong Zhang. Sars-net: Covid-19 detection from chest x-rays by combining graph convolutional network and convolutional neural network. *Pattern Recognition*, 122:108255, 2022.

[191] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017.

[192] Marc Lalonde, Mario Beaulieu, and Langis Gagnon. Fast and robust optic disc detection using pyramidal decomposition and hausdorff-based template matching. *IEEE transactions on medical imaging*, 20(11):1193–1200, 2001.

[193] Issam Laradji, Pau Rodriguez, Oscar Manas, Keegan Lensink, Marco Law, Lironne Kurzman, William Parker, David Vazquez, and Derek Nowrouzezahrai. A weakly supervised consistency-based learning method for covid-19 segmentation in ct images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2453–2462, 2021.

[194] Michael Truong Le, Frederik Diehl, Thomas Brunner, and Alois Knol. Uncertainty estimation for deep neural object detectors in safety-critical applications. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3873–3878. IEEE, 2018.

[195] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[196] Jisoo Lee and Sae-Young Chung. Robust training with ensemble consensus. 2020.

[197] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5267–5276, 2019.

[198] Yuencheng Lee, Demetri Terzopoulos, and Keith Waters. Realistic modeling for facial animation. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 55–62, 1995.

[199] Yinjie Lei, Yan Liu, Pingping Zhang, and Lingqiao Liu. Towards using count-level weak supervision for crowd counting. *Pattern Recognition*, 109:107616, 2021.

[200] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 878–885. IEEE, 2005.

[201] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. *Advances in neural information processing systems*, 23, 2010.

[202] Bin Li, Yin Li, and Kevin W Eliceiri. Dual-stream multiple instance learning network for whole slide image classification with self-supervised contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2021.

[203] Changyang Li, Xiuying Wang, Stefan Eberl, Michael Fulham, Yong Yin, Jinhu Chen, and David Dagan Feng. A likelihood and local constraint level set model for liver tumor segmentation from ct volumes. *IEEE Transactions on Biomedical Engineering*, 60(10):2967–2977, 2013.

[204] Guohao Li, Matthias Müller, Ali Thabet, and Bernard Ghanem. Can GCNs go as deep as CNNs? *arXiv preprint arXiv:1904.03751*, 2019.

[205] Hanchao Li, Pengfei Xiong, Haoqiang Fan, and Jian Sun. Dfanet: Deep feature aggregation for real-time semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9522–9531, 2019.

[206] Kunhua Li, Jiong Wu, Faqi Wu, Dajing Guo, Linli Chen, Zheng Fang, and Chuanming Li. The clinical and chest CT features associated with severe and critical COVID-19 pneumonia. *Investigative radiology*, 2020.

[207] Lin Li, Lixin Qin, Zeguo Xu, Youbing Yin, Xin Wang, Bin Kong, Junjie Bai, Yi Lu, Zhenghan Fang, Qi Song, et al. Using artificial intelligence to detect covid-19 and community-acquired pneumonia based on pulmonary ct: evaluation of the diagnostic accuracy. *Radiology*, 296(2):E65–E71, 2020.

[208] Min Li, Zhaoxiang Zhang, Kaiqi Huang, and Tieniu Tan. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder

detection. In *2008 19th international conference on pattern recognition*, pages 1–4. IEEE, 2008.

[209] Qun Li, Xuhua Guan, Peng Wu, Xiaoye Wang, Lei Zhou, Yeqing Tong, Ruiqi Ren, Kathy SM Leung, Eric HY Lau, Jessica Y Wong, et al. Early transmission dynamics in wuhan, china, of novel coronavirus–infected pneumonia. *New England journal of medicine*, 2020.

[210] Shaohua Li, Xiuchao Sui, Xiangde Luo, Xinxing Xu, Liu Yong, and Rick Siow Mong Goh. Medical image segmentation using squeeze-and-expansion transformers. In *The 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.

[211] Shuailin Li, Chuyu Zhang, and Xuming He. Shape-aware semi-supervised 3d semantic segmentation for medical images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 552–561. Springer, 2020.

[212] Shuo Li, Thomas Fevens, and A Krzyżak. A svm-based framework for autonomous volumetric medical image segmentation using hierarchical and coupled level sets. In *International Congress Series*, volume 1268, pages 207–212. Elsevier, 2004.

[213] Tao Li, Wang Bo, Chunyu Hu, Hong Kang, Hanruo Liu, Kai Wang, and Huazhu Fu. Applications of deep learning in fundus images: A review. *Medical Image Analysis*, page 101971, 2021.

[214] Wen Li et al. Automatic segmentation of liver tumor in ct images with deep convolutional neural networks. *Journal of Computer and Communications*, 3(11):146, 2015.

[215] Xia Li, Yibo Yang, Qijie Zhao, Tiancheng Shen, Zhouchen Lin, and Hong Liu. Spatial pyramid based graph reasoning for semantic segmentation. In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8950–8959, 2020.

[216] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE transactions on medical imaging*, 37(12):2663–2674, 2018.

[217] Yin Li and Abhinav Gupta. Beyond grids: Learning graph representations for visual recognition. In *Advances in Neural Information Processing Systems*, pages 9225–9235, 2018.

[218] Yuexiang Li, Dong Wei, Jiawei Chen, Shilei Cao, Hongyu Zhou, Yanchun Zhu, Jianrong Wu, Lan Lan, Wenbo Sun, Tianyi Qian, et al. Efficient and effective training of covid-19 classification networks with self-supervised dual-track learning to rank. *IEEE Journal of Biomedical and Health Informatics*, 24(10):2787–2797, 2020.

[219] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1091–1100, 2018.

[220] Yujia Li, Richard Zemel, Marc Brockschmidt, and Daniel Tarlow. Gated graph sequence neural networks. In *Proceedings of ICLR'16*, 2016.

[221] Zekun Li, Wei Zhao, Feng Shi, Lei Qi, Xingzhi Xie, Ying Wei, Zhongxiang Ding, Yang Gao, Shangjie Wu, Jun Liu, et al. A novel multiple instance learning framework for covid-19 severity assessment via data augmentation and self-supervised learning. *Medical Image Analysis*, 69:101978, 2021.

[222] Dongze Lian, Jing Li, Jia Zheng, Weixin Luo, and Shenghua Gao. Density map regression guided detection network for rgb-d crowd counting and localization. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1821–1830, 2019.

[223] Xiaodan Liang, Zhiting Hu, Hao Zhang, Liang Lin, and Eric P Xing. Symbolic graph reasoning meets convolutions. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 1858–1868, 2018.

[224] Fangzhou Liao, Ming Liang, Zhe Li, Xiaolin Hu, and Sen Song. Evaluate the malignancy of pulmonary nodules using the 3-d deep leaky noisy-or network. *IEEE transactions on neural networks and learning systems*, 30(11):3484–3495, 2019.

[225] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1925–1934, 2017.

[226] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[227] Chen Liu, Jinze Cui, Dailin Gan, and Guosheng Yin. Beyond covid-19 diagnosis: Prognosis with hierarchical graph representation learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 283–292. Springer, 2021.

[228] Chenchen Liu, Xinyu Weng, and Yadong Mu. Recurrent attentive zooming for joint crowd counting and precise localization. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1217–1226. IEEE, 2019.

[229] Jiang Liu, Chenqiang Gao, Deyu Meng, and Alexander G Hauptmann. Decidenet: Counting varying density crowds through attention guided detection and density estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2018.

[230] Jiannan Liu, Bo Dong, Shuai Wang, Hui Cui, Deng-Ping Fan, Jiquan Ma, and Geng Chen. Covid-19 lung infection segmentation with a novel two-stage cross-domain transfer learning framework. *Medical Image Analysis*, 74:102205, 2021.

[231] Lingbo Liu, Jiaqi Chen, Hefeng Wu, Tianshui Chen, Guanbin Li, and Liang Lin. Efficient crowd counting via structured knowledge transfer. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2645–2654, 2020.

[232] Lingbo Liu, Jiaqi Chen, Hefeng Wu, Guanbin Li, Chenglong Li, and Liang Lin. Cross-modal collaborative representation learning and a large-scale rgbt benchmark for crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4823–4833, 2021.

[233] Lingbo Liu, Zhilin Qiu, Guanbin Li, Shufan Liu, Wanli Ouyang, and Liang Lin. Crowd counting with deep structured scale integration network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1774–1783, 2019.

[234] Lingbo Liu, Hongjun Wang, Guanbin Li, Wanli Ouyang, and Liang Lin. Crowd counting using deep recurrent spatial-aware network. In *IJCAI*, 2018.

[235] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[236] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5099–5108, 2019.

[237] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Estimating people flows to better count them in crowded scenes. In *European Conference on Computer Vision*, pages 723–740. Springer, 2020.

[238] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Counting people by estimating people flows. *IEEE transactions on pattern analysis and machine intelligence*, 2021.

[239] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. Leveraging unlabeled data for crowd counting by learning to rank. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7661–7669, 2018.

[240] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. Exploiting unlabeled data in cnns by self-supervised learning to rank. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1862–1878, 2019.

[241] Xiaoming Liu, Quan Yuan, Yaozong Gao, Kelei He, Shuo Wang, Xiao Tang, Jinshan Tang, and Dinggang Shen. Weakly supervised segmentation of covid19 infection with scribble annotation on ct images. *Pattern recognition*, 122:108341, 2022.

[242] Xiyang Liu, Jie Yang, Wenrui Ding, Tieqiang Wang, Zhijin Wang, and Junjun Xiong. Adaptive mixture regression network with local counting map for crowd counting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 241–257. Springer, 2020.

[243] Yaojie Liu, Amin Jourabloo, William Ren, and Xiaoming Liu. Dense face alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1619–1628, 2017.

[244] Yun Liu, Krishna Gadepalli, Mohammad Norouzi, George E Dahl, Timo Kohlberger, Aleksey Boyko, Subhashini Venugopalan, Aleksei Timofeev, Philip Q Nelson, Greg S Corrado, et al. Detecting cancer metastases on gigapixel pathology images. *arXiv preprint arXiv:1703.02442*, 2017.

[245] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1377–1385, 2015.

[246] EH Lockwood. Length of ellipse. *The Mathematical Gazette*, 16(220):269–270, 1932.

[247] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[248] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. *ICLR*, 2017.

[249] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[250] Yao Lu, Soren Pirk, Jan Dlabal, Anthony Brohan, Ankita Pasad, Zhao Chen, Vincent Casser, Anelia Angelova, and Ariel Gordon. Taskology: Utilizing task relations at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8700–8709, 2021.

[251] Ao Luo, Xin Li, Fan Yang, Zhicheng Jiao, Hong Cheng, and Siwei Lyu. Cascade graph neural networks for rgb-d salient object detection. In *European Conference on Computer Vision*, pages 346–364. Springer, 2020.

[252] Ao Luo, Fan Yang, Xin Li, Dong Nie, Zhicheng Jiao, Shangchen Zhou, and Hong Cheng. Hybrid graph neural networks for crowd counting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11693–11700, 2020.

[253] Luyang Luo, Lequan Yu, Hao Chen, Quande Liu, Xi Wang, Jiaqi Xu, and Pheng-Ann Heng. Deep mining external imperfect data for chest x-ray disease screening. *IEEE transactions on medical imaging*, 39(11):3583–3594, 2020.

[254] Xiangde Luo. Ssl4mis. https://github.com/hilab-git/ssl4mis, 2020.

[255] Xiangde Luo, Jieneng Chen, Tao Song, and Guotai Wang. Semi-supervised medical image segmentation through dual-task consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8801–8809, 2021.

[256] Xiangde Luo, Wenjun Liao, Jieneng Chen, Tao Song, Yinan Chen, Shichuan Zhang, Nianyong Chen, Guotai Wang, and Shaoting Zhang. Efficient semi-supervised gross target volume of nasopharyngeal carcinoma segmentation via uncertainty rectified pyramid consistency. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 318–329. Springer, 2021.

[257] Marcel Lüthi, Thomas Gerig, Christoph Jud, and Thomas Vetter. Gaussian process morphable models. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1860–1873, 2017.

[258] Jun Ma, Zhan Wei, Yiwen Zhang, Yixin Wang, Rongfei Lv, Cheng Zhu, Chen Gaoxiang, Jianan Liu, Chao Peng, Lei Wang, et al. How distance transform maps boost

segmentation cnns: an empirical study. In *Medical Imaging with Deep Learning*, pages 479–492. PMLR, 2020.

[259] Yu-Jen Ma, Hong-Han Shuai, and Wen-Huang Cheng. Spatiotemporal dilated convolution with uncertain matching for video-based crowd estimation. *IEEE Transactions on Multimedia*, 2021.

[260] Yuhui Ma, Huaying Hao, Jianyang Xie, Huazhu Fu, Jiong Zhang, Jianlong Yang, Zhen Wang, Jiang Liu, Yalin Zheng, and Yitian Zhao. ROSE: A retinal OCT-angiography vessel segmentation dataset and new model. *IEEE Transactions on Medical Imaging*, 40(3):928–939, 2020.

[261] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6142–6151, 2019.

[262] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Learning scales from points: A scale-aware probabilistic model for crowd counting. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 220–228, 2020.

[263] Kit-Kay Mak and Mallikarjuna Rao Pichika. Artificial intelligence in drug development: present status and future prospects. *Drug discovery today*, 24(3):773–780, 2019.

[264] Aakarsh Malhotra, Surbhi Mittal, Puspita Majumdar, Saheb Chhabra, Kartik Thakral, Mayank Vatsa, Richa Singh, Santanu Chaudhury, Ashwin Pudrod, and Anjali Agrawal. Multi-task driven explainable diagnosis of covid-19 using chest x-ray images. *Pattern Recognition*, page 108243, 2021.

[265] Ankur Mallick, Chaitanya Dwivedi, Bhavya Kailkhura, Gauri Joshi, and T Yong-Jin Han. Can your ai differentiate cats from covid-19? sample efficient uncertainty estimation for deep learning safety. In *ICML 2020 Workshop on Uncertainty & Robustness in Deep Learning*, 2020.

[266] Diego Marcos, Devis Tuia, Benjamin Kellenberger, Lisa Zhang, Min Bai, Renjie Liao, and Raquel Urtasun. Learning deep structured active contours end-to-end. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8877–8885, 2018.

[267] Haggai Maron, Ethan Fetaya, Nimrod Segol, and Yaron Lipman. On the universality of invariant networks. In *International conference on machine learning*, pages 4363–4371. PMLR, 2019.

[268] Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. *Advances in neural information processing systems*, pages 570–576, 1998.

[269] Mark Marsden, Kevin McGuinness, Suzanne Little, Ciara E Keogh, and Noel E O'Connor. People, penguins and petri dishes: Adapting object counting models to new visual domains and object types without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8070–8079, 2018.

[270] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, June 1947.

[271] Yanda Meng, Joshua Bridge, Cliff Addison, Manhui Wang, Cristin Merritt, Stu Franks, Maria Mackey, Steve Messenger, Renrong Sun, Yitian Zhao, and Yalin Zheng. Bilateral adaptive graph convolutional network on ct based covid-19 diagnosis with uncertainty-aware consensus-assisted multiple instance learning. *Under Review*, 2022.

[272] Yanda Meng, Joshua Bridge, Meng Wei, Yitian Zhao, Yihong Qiao, Xiaoyun Yang, Xiaowei Huang, and Yalin Zheng. Counting with adaptive auxiliary learning. *arXiv preprint arXiv:2203.04061*, 2022.

[273] Yanda Meng, Xu Chen, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Yihong Qiao, Xiaowei Huang, and Yalin Zheng. 3d dense face alignment with fused features by aggregating cnns and gcns. *arXiv preprint arXiv:2203.04643*, 2022.

[274] Yanda Meng, Xu Chen, Hongrun Zhang, Yitian Zhao, Dongxu Gao, Barbra Hamill, Godhuli Patri, Tunde Peto, Savita Madhusudhan, and Yalin Zheng. Shape-aware weakly/semi-supervised optic disc and cup segmentation with regional/marginal consistency. *Under Review*, 2022.

[275] Yanda Meng, Xu Chen, Hongrun Zhang, Yitian Zhao, Dongxu Gao, Barbra Hamill, Godhuli Patri, Tunde Peto, Savita Madhusudhan, and Yalin Zheng. Shape-aware weakly/semi-supervised optic disc and cup segmentation with regional/marginal consistency. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2022.

[276] Yanda Meng, Wei Meng, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Xiaowei Huang, and Yalin Zheng. Regression of instance boundary by aggregated cnn and gcn. In *European Conference on Computer Vision*, pages 190–207. Springer, 2020.

[277] Yanda Meng, Wei Meng, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Xiaowei Huang, and Yalin Zheng. Regression of instance boundary by aggregated cnn and gcn. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[278] Yanda Meng, Meng Wei, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Xiaowei Huang, and Yalin Zheng. Cnn-gcn aggregation enabled boundary regression for biomedical

image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 352–362. Springer, 2020.

[279] Yanda Meng, Meng Wei, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Xiaowei Huang, and Yalin Zheng. CNN-GCN aggregation enabled boundary regression for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, page in press, 2020.

[280] Yanda Meng, Hongrun Zhang, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Xuesheng Qian, Xiaowei Huang, and Yalin Zheng. Bi-gcn: Boundary-aware input-dependent graph convolution network for biomedical image segmentation. In *32nd British Machine Vision Conference: BMVC 2021*. British Machine Vision Association, 2021.

[281] Yanda Meng, Hongrun Zhang, Dongxu Gao, Yitian Zhao, Xiaoyun Yang, Xuesheng Qian, Xiaowei Huang, and Yalin Zheng. Bi-gconv: boundary-aware input-dependent graph convolution for biomedical image segmentation. In *The British Machine Vision Conference (BMVC)*, 2021.

[282] Yanda Meng, Hongrun Zhang, Yitian Zhao, Dongxu Gao, Barbra Hamill, Godhuli Patri, Tunde Peto, Savita Madhusudhan, and Yalin Zheng. Dual consistency enabled weakly and semi-supervised optic disc and cup segmentation with dual adaptive graph convolutional networks. *Under Review*, 2022.

[283] Yanda Meng, Hongrun Zhang, Yitian Zhao, Dongxu Gao, Barbra Hamill, Godhuli Patri, Tunde Peto, Savita Madhusudhan, and Yalin Zheng. Dual consistency enabled weakly and semi-supervised optic disc and cup segmentation with dual adaptive graph convolutional networks. *IEEE Transactions on Medical Imaging*, page in press, 2022.

[284] Yanda Meng, Hongrun Zhang, Yitian Zhao, Xiaoyun Yang, Xuesheng Qian, Xiaowei Huang, and Yalin Zheng. Spatial uncertainty-aware semi-supervised crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15549–15559, 2021.

[285] Yanda Meng, Hongrun Zhang, Yitian Zhao, Xiaoyun Yang, Yihong Qiao, J. C. MacCormick Ian, Xiaowei Huang, and Yalin Zheng. Graph-based region and boundary aggregation for biomedical image segmentation. In *IEEE Transactions on Medical Imaging*, 2021.

[286] Yanda Meng, Hongrun Zhang, Yitian Zhao, Xiaoyun Yang, Yihong Qiao, Ian JC MacCormick, Xiaowei Huang, and Yalin Zheng. Graph-based region and boundary aggregation for biomedical image segmentation. *IEEE Transactions on Medical Imaging*, 2021.

[287] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.

[288] Yunqi Miao, Zijia Lin, Guiguang Ding, and Jungong Han. Shallow feature based dense attention network for crowd counting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11765–11772, 2020.

[289] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016.

[290] Shervin Minaee, Rahele Kafieh, Milan Sonka, Shakib Yazdani, and Ghazaleh Jamalipour Soufi. Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning. *Medical image analysis*, 65:101794, 2020.

[291] Hong Mo, Wenqi Ren, Yuan Xiong, Xiaoqi Pan, Zhong Zhou, Xiaochun Cao, and Wei Wu. Background noise filtering and distribution dividing for crowd counting. *IEEE Transactions on Image Processing*, 29:8199–8212, 2020.

[292] Davide Modolo, Bing Shuai, Rahul Rama Varior, and Joseph Tighe. Understanding the impact of mistakes on background regions in crowd counting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1650–1659, 2021.

[293] Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5115–5124, 2017.

[294] Sergey P Morozov, Anna E Andreychenko, Ivan A Blokhin, Pavel B Gelezhe, Anna P Gonchar, Alexander E Nikolaev, Nikolay A Pavlov, Valeria Yu Chernina, and Victor A Gombolevskiy. Mosmeddata: data set of 1110 chest ct scans performed during the covid-19 epidemic. *Digital Diagnostics*, 1(1):49–59, 2020.

[295] Lei Mou, Yitian Zhao, Li Chen, Jun Cheng, Zaiwang Gu, Huaying Hao, Hong Qi, Yalin Zheng, Alejandro Frangi, and Jiang Liu. CS-Net: Channel and spatial attention network for curvilinear structure segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 721–730. Springer, 2019.

[296] Lei Mou, Yitian Zhao, Huazhu Fu, Yonghuai Liu, Jun Cheng, Yalin Zheng, Pan Su, Jianlong Yang, Li Chen, Alejandro F Frangi, et al. CS2-Net: Deep learning segmentation of curvilinear structures in medical imaging. *Medical Image Analysis*, 67:101874, 2021.

[297] Mavuto M Mukaka. A guide to appropriate use of correlation coefficient in medical research. *Malawi medical journal*, 24(3):69–71, 2012.

[298] Mircea Paul Muresan, Sergiu Nedevschi, and Radu Danescu. Robust data association using fusion of data-driven and engineered features for real-time pedestrian tracking in thermal images. *Sensors*, 21(23):8005, 2021.

[299] Balamurali Murugesan, Kaushik Sarveswaran, Sharath M Shankaranarayana, Keerthi Ram, Jayaraj Joseph, and Mohanasankar Sivaprakasam. Psi-Net: Shape and boundary aware joint multi-task deep network for medical image segmentation. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 7223–7226. IEEE, 2019.

[300] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Icml*, 2010.

[301] Thomas Neumann, Kiran Varanasi, Stephan Wenger, Markus Wacker, Marcus Magnor, and Christian Theobalt. Sparse localized deformation components. *ACM Transactions on Graphics (TOG)*, 32(6):1–10, 2013.

[302] Guang-Yu Nie, Ming-Ming Cheng, Yun Liu, Zhengfa Liang, Deng-Ping Fan, Yue Liu, and Yongtian Wang. Multi-level context ultra-aggregation for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3283–3291, 2019.

[303] Kyoung Jin Noh, Sang Jun Park, and Soochahn Lee. Combining fundus images and fluorescein angiography for artery/vein classification using the hierarchical vessel graph network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 595–605. Springer, 2020.

[304] Yujin Oh, Sangjoon Park, and Jong Chul Ye. Deep learning covid-19 features on cxr using limited training data sets. *IEEE Transactions on Medical Imaging*, 39(8):2688–2700, 2020.

[305] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Gray scale and rotation invariant texture classification with local binary patterns. In *European conference on computer vision*, pages 404–420. Springer, 2000.

[306] Daniel Onoro-Rubio and Roberto J López-Sastre. Towards perspective-free object counting with deep learning. In *European Conference on Computer Vision*, pages 615–629. Springer, 2016.

[307] José Ignacio Orlando, Huazhu Fu, João Barbossa Breda, Karel van Keer, Deepti R Bathula, Andrés Diaz-Pinto, Ruogu Fang, Pheng-Ann Heng, Jeyoung Kim, JoonHo Lee, et al. REFUGE challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs. *Medical Image Analysis*, 59:101570, 2020.

[308] Xi Ouyang, Jiayu Huo, Liming Xia, Fei Shan, Jun Liu, Zhanhao Mo, Fuhua Yan, Zhongxiang Ding, Qi Yang, Bin Song, et al. Dual-sampling attention network for diagnosis of covid-19 from community acquired pneumonia. *IEEE Transactions on Medical Imaging*, 39(8):2595–2605, 2020.

[309] Nikos Paragios and Visvanathan Ramesh. A mrf-based approach for real-time subway monitoring. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[310] Steven M Paul, Daniel S Mytelka, Christopher T Dunwiddie, Charles C Persinger, Bernard H Munos, Stacy R Lindborg, and Aaron L Schacht. How to improve r&d productivity: the pharmaceutical industry's grand challenge. *Nature reviews Drug discovery*, 9(3):203–214, 2010.

[311] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 724–732, 2016.

[312] Oscar Perdomo, Sebastian Otálora, Fabio A González, Fabrice Meriaudeau, and Henning Müller. Oct-net: A convolutional network for automatic classification of normal and diabetic macular edema using sd-oct volumes. In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 1423–1426. IEEE, 2018.

[313] Sérgio Pereira, Adriano Pinto, Victor Alves, and Carlos A Silva. Brain tumor segmentation using convolutional neural networks in mri images. *IEEE transactions on medical imaging*, 35(5):1240–1251, 2016.

[314] Viet-Quoc Pham, Tatsuo Kozakaya, Osamu Yamaguchi, and Ryuzo Okada. Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 3253–3261, 2015.

[315] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1713–1721, 2015.

[316] Frank G Preston, Yanda Meng, Jamie Burgess, Maryam Ferdousi, Shazli Azmi, Ioannis N Petropoulos, Stephen Kaye, Rayaz A Malik, Yalin Zheng, and Uazman Alam. Artificial intelligence utilising corneal confocal microscopy for the diagnosis of peripheral neuropathy in diabetes mellitus and prediabetes. *Diabetologia*, 65(3):457–466, 2022.

[317] Xuelin Qian, Huazhu Fu, Weiya Shi, Tao Chen, Yanwei Fu, Fei Shan, and Xiangyang Xue. M ̂3 lung-sys: A deep learning system for multi-class lung pneumonia screening from ct imaging. *IEEE journal of biomedical and health informatics*, 24(12):3539–3550, 2020.

[318] Jiaming Qiu and Yankui Sun. Self-supervised iterative refinement learning for macular oct volumetric data classification. *Computers in biology and medicine*, 111:103327, 2019.

[319] Joaquin Quinonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. Mit Press, 2008.

[320] Mohammad Rahimzadeh, Abolfazl Attar, and Seyed Mohammad Sakhaei. A fully automated deep learning-based network for detecting covid-19 from a new and large lung ct scan dataset. *Biomedical Signal Processing and Control*, page 102588, 2021.

[321] Marwa Chendeb EL Rai, Naoufel Werghi, Hassan Al Muhairi, and Habiba Alsafar. Using facial images for the diagnosis of genetic syndromes: a survey. In *2015 Inter-*

national Conference on Communications, Signal Processing, and their Applications (ICCSPA'15), pages 1–6. IEEE, 2015.

[322] Martin Rajchl, Matthew CH Lee, Ozan Oktay, Konstantinos Kamnitsas, Jonathan Passerat-Palmbach, Wenjia Bai, Mellisa Damodaram, Mary A Rutherford, Joseph V Hajnal, Bernhard Kainz, et al. Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE transactions on medical imaging*, 36(2):674–683, 2016.

[323] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. Generating 3d faces using convolutional mesh autoencoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 704–720, 2018.

[324] Marc'Aurelio Ranzato, Fu Jie Huang, Y-Lan Boureau, and Yann LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.

[325] Michael P Recht, Marc Dewey, Keith Dreyer, Curtis Langlotz, Wiro Niessen, Barbara Prainsack, and John J Smith. Integrating artificial intelligence into the clinical practice of radiology: challenges and recommendations. *European radiology*, 30(6):3576–3584, 2020.

[326] Mahesh Kumar Krishna Reddy, Mrigank Rochan, Yiwei Lu, and Yang Wang. Adacrowd: Unlabeled scene adaptation for crowd counting. *IEEE Transactions on Multimedia*, 2020.

[327] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[328] Pia Reittner, Suzanne Ward, Laura Heyneman, Takeshi Johkoh, and Nestor L Müller. Pneumonia: high-resolution CT findings in 114 patients. *European radiology*, 13(3):515–521, 2003.

[329] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[330] Weihong Ren, Xinchao Wang, Jiandong Tian, Yandong Tang, and Antoni B Chan. Tracking-by-counting: Using network flows on crowd density maps for tracking multiple targets. *IEEE Transactions on Image Processing*, 30:1439–1452, 2020.

[331] Sungmin Rhee, Seokjun Seo, and Sun Kim. Hybrid approach of relation network and localized graph convolutional filtering for breast cancer subtype classification. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3527–3534, 2018.

[332] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 460–469. IEEE, 2016.

[333] Elad Richardson, Matan Sela, Roy Or-El, and Ron Kimmel. Learning detailed face reconstruction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1259–1268, 2017.

[334] Michael Roberts, Derek Driggs, Matthew Thorpe, Julian Gilbey, Michael Yeung, Stephan Ursprung, Angelica I Aviles-Rivero, Christian Etmann, Cathal McCague, Lucian Beer, et al. Common pitfalls and recommendations for using machine learning to detect and prognosticate for covid-19 using chest radiographs and ct scans. *Nature Machine Intelligence*, 3(3):199–217, 2021.

[335] Xavier Robin, Natacha Turck, Alexandre Hainard, Natalia Tiberti, Frédérique Lisacek, Jean-Charles Sanchez, and Markus Müller. proc: an open-source package for r and s+ to analyze and compare roc curves. *BMC Bioinformatics*, 12(1):1–8, 2011.

[336] Sami Romdhani and Thomas Vetter. Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 986–993. IEEE, 2005.

[337] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015.

[338] Subhankar Roy, Willi Menapace, Sebastiaan Oei, Ben Luijten, Enrico Fini, Cristiano Saltori, Iris Huijben, Nishith Chennakeshava, Federico Mento, Alessandro Sentelli, et al. Deep learning for classification and localization of covid-19 markers in point-of-care lung ultrasound. *IEEE Transactions on Medical Imaging*, 39(8):2676–2687, 2020.

[339] Daniel L Rubin. Informatics in radiology: measuring and improving quality in radiology: meeting the challenge with informatics. *Radiographics*, 31(6):1511–1527, 2011.

[340] David Ryan, Simon Denman, Clinton Fookes, and Sridha Sridharan. Crowd counting using multiple local features. In *2009 Digital Image Computing: Techniques and Applications*, pages 81–88. IEEE, 2009.

[341] Deepak Babu Sam, Skand Vishwanath Peri, Mukuntha Narayanan Sundararaman, Amogh Kamath, and Venkatesh Babu Radhakrishnan. Locate, size and count: Accurately resolving people in dense crowds via detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[342] Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin Riedmiller, Raia Hadsell, and Peter Battaglia. Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning*, pages 4470–4479. PMLR, 2018.

[343] Aliaksei Sandryhaila and José MF Moura. Discrete signal processing on graphs. *IEEE transactions on signal processing*, 61(7):1644–1656, 2013.

[344] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[345] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 265–273. Springer, 2018.

[346] Santiago Segarra, Antonio G Marques, and Alejandro Ribeiro. Optimal graph-filter design and applications to distributed linear network operators. *IEEE Transactions on Signal Processing*, 65(15):4117–4131, 2017.

[347] Matan Sela, Elad Richardson, and Ron Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1576–1585, 2017.

[348] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[349] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[350] Afshar Shamsi, Hamzeh Asgharnezhad, Shirin Shamsi Jokandan, Abbas Khosravi, Parham M Kebria, Darius Nahavandi, Saeid Nahavandi, and Dipti Srinivasan. An uncertainty-aware transfer learning-based framework for covid-19 diagnosis. *IEEE transactions on neural networks and learning systems*, 32(4):1408–1417, 2021.

[351] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[352] Claude Elwood Shannon and Warren Weaver. The mathematical theory of communication. *Illinois press, Urbana, Illinois*, 11:117, 1949.

[353] Jing Shao, Kai Kang, Chen Change Loy, and Xiaogang Wang. Deeply learned attributes for crowded scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4657–4666, 2015.

[354] Zhuchen Shao, Hao Bian, Yang Chen, Yifeng Wang, Jian Zhang, Xiangyang Ji, et al. Transmil: Transformer based correlated multiple instance learning for whole slide image classification. *Advances in Neural Information Processing Systems*, 34:2136–2147, 2021.

[355] Jianbing Shen, Xingping Dong, Jianteng Peng, Xiaogang Jin, Ling Shao, and Fatih Porikli. Submodular function optimization for motion clustering and image segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 30(9):2637–2649, 2019.

[356] Jianbing Shen, Yunfan Du, Wenguan Wang, and Xuelong Li. Lazy random walks for superpixel segmentation. *IEEE Transactions on Image Processing*, 23(4):1451–1462, 2014.

[357] Jianbing Shen, Jianteng Peng, Xingping Dong, Ling Shao, and Fatih Porikli. Higher order energies for image segmentation. *IEEE Transactions on Image Processing*, 26(10):4911–4922, 2017.

[358] Heshui Shi, Xiaoyu Han, Nanchuan Jiang, Yukun Cao, Osamah Alwalid, Jin Gu, Yanqing Fan, and Chuansheng Zheng. Radiological findings from 81 patients with COVID-19 pneumonia in wuhan, china: a descriptive study. *The Lancet infectious diseases*, 20(4):425–434, 2020.

[359] Zenglin Shi, Pascal Mettes, and Cees GM Snoek. Counting with focus for free. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4200–4209, 2019.

[360] Seung Yeon Shin, Soochahn Lee, Il Dong Yun, and Kyoung Mu Lee. Deep vessel segmentation by learning graphical connectivity. *Medical Image Analysis*, 58:101556, 2019.

[361] Mohammad Shorfuzzaman and M Shamim Hossain. Metacovid: A siamese neural network framework with contrastive loss for n-shot diagnosis of covid-19 patients. *Pattern recognition*, 113:107700, 2021.

[362] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

[363] Alberto Signoroni, Mattia Savardi, Sergio Benini, Nicola Adami, Riccardo Leonardi, Paolo Gibellini, Filippo Vaccher, Marco Ravanelli, Andrea Borghesi, Roberto Maroldi, et al. Bs-net: Learning covid-19 pneumonia severity on a large chest x-ray dataset. *Medical Image Analysis*, 71:102046, 2021.

[364] Juan Silva, Aymeric Histace, Olivier Romain, Xavier Dray, and Bertrand Granado. Toward embedded detection of polyps in WCE images for early diagnosis of colorectal cancer. *International Journal of Computer Assisted Radiology and Surgery*, 9(2):283–293, 2014.

[365] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[366] Vishwanath Sindagi and Vishal Patel. Ha-ccn: Hierarchical attention-based crowd counting network. *IEEE Transactions on Image Processing*, 29:323–335, 2019.

[367] Vishwanath A Sindagi and Vishal M Patel. Generating high-quality crowd density maps using contextual pyramid cnns. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1861–1870, 2017.

[368] Vishwanath A Sindagi and Vishal M Patel. Multi-level bottom-top and top-bottom feature fusion for crowd counting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1002–1012, 2019.

[369] Vishwanath A Sindagi, Rajeev Yasarla, and Vishal M Patel. Pushing the frontiers of unconstrained crowd counting: New dataset and benchmark method. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1221–1231, 2019.

[370] Jayanthi Sivaswamy, SR Krishnadas, Gopal Datt Joshi, Madhulika Jain, and A Ujjwaft Syed Tabish. Drishti-GS: Retinal image dataset for optic nerve head (ONH) segmentation. In *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, pages 53–56. IEEE, 2014.

[371] Paolo Soda, Natascha Claudia D'Amico, Jacopo Tessadori, Giovanni Valbusa, Valerio Guarrasi, Chandra Bortolotto, Muhammad Usman Akbar, Rosa Sicilia, Ermanno Cordelli, Deborah Fazzini, et al. Aiforcovid: predicting the clinical outcomes in patients with covid-19 applying ai to chest-x-rays. an italian multicentre study. *Medical image analysis*, 74:102216, 2021.

[372] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu. Rethinking counting and localization in crowds: A purely point-based framework. *The IEEE International Conference on Computer Vision (ICCV)*, 2021.

[373] Qingyu Song, Changan Wang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Jian Wu, and Jiayi Ma. To choose or to fuse? scale selection for crowd counting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2576–2583, 2021.

[374] Vardhani Shivapuja Sravya, Pradeep Khamka Mansi, Bajaj Divij, Ramakrishnan Ganesh, and Kiran Sarvadevabhatla Ravi. Wisdom of (binned) crowds: A bayesian stratification paradigm for crowd counting. In *Proceedings of the 2021 ACM Conference on Multimedia*, China, 2021. ACM.

[375] Ziyu Su, Thomas E Tavolara, Gabriel Carreno-Galeano, Sang Jin Lee, Metin N Gurcan, and MKK Niazi. Attention2majority: Weak multiple instance learning for regenerative kidney grading on whole slide images. *Medical Image Analysis*, 79:102462, 2022.

[376] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. *arXiv preprint arXiv:1904.04514*, 2019.

[377] Michael Suttie, Tatiana Foroud, Leah Wetherill, Joseph L Jacobson, Christopher D Molteno, Ernesta M Meintjes, H Eugene Hoyme, Nathaniel Khaole, Luther K Robinson, Edward P Riley, et al. Facial dysmorphism across the fetal alcohol spectrum. *Pediatrics*, 131(3):e779–e788, 2013.

[378] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[379] Nima Tajbakhsh, Suryakanth R Gurudu, and Jianming Liang. Automated polyp detection in colonoscopy videos using shape and context information. *IEEE Transactions on Medical Imaging*, 35(2):630–644, 2015.

[380] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.

[381] Weijun Tan and Jingfeng Liu. A 3d cnn network with bert for automatic covid-19 diagnosis from ct-scan images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 439–445, 2021.

[382] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, pages 1195–1204, 2017.

[383] National Lung Screening Trial Research Team. Reduced lung-cancer mortality with low-dose computed tomographic screening. *New England Journal of Medicine*, 365(5):395–409, 2011.

[384] Yih-Chung Tham, Xiang Li, Tien Y Wong, Harry A Quigley, Tin Aung, and Ching-Yu Cheng. Global prevalence of glaucoma and projections of glaucoma burden through 2040: a systematic review and meta-analysis. *Ophthalmology*, 121(11):2081–2090, 2014.

[385] Yan Tian, Leonid Sigal, Hernán Badino, Fernando De la Torre, and Yong Liu. Latent gaussian mixture regression for human pose estimation. In *Asian Conference on Computer Vision*, pages 679–690. Springer, 2010.

[386] Yukun Tian, Yiming Lei, Junping Zhang, and James Z Wang. Padnet: Pan-density crowd counting. *IEEE Transactions on Image Processing*, 29:2714–2727, 2019.

[387] Ibrahim Saygin Topkaya, Hakan Erdogan, and Fatih Porikli. Counting people by clustering person detector outputs. In *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 313–318. IEEE, 2014.

[388] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

[389] Alexander Toshev and Christian Szegedy. DeepPose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660, 2014.

[390] Luan Tran and Xiaoming Liu. Nonlinear 3d face morphable model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7346–7355, 2018.

[391] William T Tran, Ali Sadeghi-Naini, Fang-I Lu, Sonal Gandhi, Nicholas Meti, Muriel Brackstone, Eileen Rakovitch, and Belinda Curpen. Computational radiology in breast cancer screening and diagnosis using artificial intelligence. *Canadian Association of Radiologists Journal*, 72(1):98–108, 2021.

[392] Anh Tun Trn, Tal Hassner, Iacopo Masi, Eran Paz, Yuval Nirkin, and Gérard Medioni. Extreme 3d face reconstruction: Seeing through occlusions. In *Proceed-*

ings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3935–3944, 2018.

[393] Andy Tsai, Anthony Yezzi, William Wells, Clare Tempany, Dewey Tucker, Ayres Fan, W Eric Grimson, and Alan Willsky. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE transactions on medical imaging*, 22(2):137–154, 2003.

[394] Anh Tuan Tran, Tal Hassner, Iacopo Masi, and Gérard Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5163–5172, 2017.

[395] Tomoki Uemura, Janne J Näppi, Chinatsu Watari, Toru Hironaka, Tohru Kamiya, and Hiroyuki Yoshida. Weakly unsupervised conditional generative adversarial network for image-based prognostic prediction for covid-19 patients based on chest ct. *Medical Image Analysis*, 73:102159, 2021.

[396] Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[397] Han Van De Waterbeemd and Eric Gifford. Admet in silico modelling: towards prediction paradise? *Nature reviews Drug discovery*, 2(3):192–204, 2003.

[398] Thomas LA van den Heuvel, Dagmar de Bruijn, Chris L de Korte, and Bram van Ginneken. Automated measurement of fetal head circumference using 2D ultrasound images. *PloS One*, 13(8), 2018.

[399] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[400] David Vázquez, Jorge Bernal, F Javier Sánchez, Gloria Fernández-Esparrach, Antonio M López, Adriana Romero, Michal Drozdzal, and Aaron Courville. A benchmark for endoluminal scene segmentation of colonoscopy images. *Journal of Healthcare Engineering*, 2017, 2017.

[401] Pablo Vieira, Orrana Sousa, Deborah Magalhes, Ricardo Rablo, and Romuere Silva. Detecting pulmonary diseases using deep features in x-ray images. *Pattern Recognition*, page 108081, 2021.

[402] José Vilar, Maria Luisa Domingo, Cristina Soto, and Jonathan Cogollos. Radiology of bacterial pneumonia. *European journal of radiology*, 51(2):102–113, 2004.

[403] Kentaro Wada. labelme: Image Polygonal Annotation with Python. https://github.com/wkentaro/labelme, 2016.

[404] Andrew Wagner, John Wright, Arvind Ganesh, Zihan Zhou, Hossein Mobahi, and Yi Ma. Toward a practical face recognition system: Robust alignment and illumination by sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(2):372–386, 2011.

[405] Elad Walach and Lior Wolf. Learning to count with cnn boosting. In *European conference on computer vision*, pages 660–676. Springer, 2016.

[406] Jia Wan and Antoni Chan. Adaptive density map generation for crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1130–1139, 2019.

[407] Jia Wan and Antoni Chan. Modeling noisy annotations for crowd counting. *Advances in Neural Information Processing Systems*, 33, 2020.

[408] Jia Wan, Nikil Senthil Kumar, and Antoni B Chan. Fine-grained crowd counting. *IEEE transactions on image processing*, 30:2114–2126, 2021.

[409] Jia Wan, Ziquan Liu, and Antoni B Chan. A generalized loss function for crowd counting and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1974–1983, 2021.

[410] Jia Wan, Qingzhong Wang, and Antoni B Chan. Kernel-based density map generation for dense object counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[411] Bo Wang, Wei Wei, Shuang Qiu, Shengpei Wang, Dan Li, and Huiguang He. Boundary aware U-Net for retinal layers segmentation in optical coherence tomography images. *IEEE Journal of Biomedical and Health Informatics*, 2021.

[412] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai Nguyen. Distribution matching for crowd counting. *Advances in Neural Information Processing Systems*, 33, 2020.

[413] Changan Wang, Qingyu Song, Boshen Zhang, Yabiao Wang, Ying Tai, Xuyi Hu, Chengjie Wang, Jilin Li, Jiayi Ma, and Yang Wu. Uniformity in heterogeneity: Diving deep into count interval partition for crowd counting. *The IEEE International Conference on Computer Vision (ICCV)*, 2021.

[414] Chen Wang, Peter W Horby, Frederick G Hayden, and George F Gao. A novel coronavirus outbreak of global health concern. *The lancet*, 395(10223):470–473, 2020.

[415] Guotai Wang, Xinglong Liu, Chaoping Li, Zhiyong Xu, Jiugen Ruan, Haifeng Zhu, Tao Meng, Kang Li, Ning Huang, and Shaoting Zhang. A noise-robust framework for automatic segmentation of covid-19 pneumonia lesions from ct images. *IEEE Transactions on Medical Imaging*, 39(8):2653–2663, 2020.

[416] Jun Wang, Yiming Bao, Yaofeng Wen, Hongbing Lu, Hu Luo, Yunfei Xiang, Xiaoming Li, Chen Liu, and Dahong Qian. Prior-attention residual learning for more discriminative covid-19 screening in ct images. *IEEE Transactions on Medical Imaging*, 39(8):2572–2583, 2020.

[417] Jun Wang, Jiawei Wang, Yaofeng Wen, Hongbing Lu, Tianye Niu, Jiangfeng Pan, and Dahong Qian. Pulmonary nodule detection in volumetric chest ct scans using cnns-based nodule-size-adaptive detection and classification. *IEEE Access*, 7:46033–46044, 2019.

[418] Mingjie Wang, Hao Cai, Xianfeng Han, Jun Zhou, and Minglun Gong. Stnet: Scale tree network with multi-level auxiliator for crowd counting. *IEEE Transactions on Multimedia*, 2022.

[419] Qi Wang, Junyu Gao, Wei Lin, and Xuelong Li. NWPU-Crowd: A large-scale benchmark for crowd counting and localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[420] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Learning from synthetic data for crowd counting in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8198–8207, 2019.

[421] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Pixel-wise crowd understanding via synthetic data. *International Journal of Computer Vision*, 129(1):225–245, 2021.

[422] Qi Wang, Wei Lin, Junyu Gao, and Xuelong Li. Density-aware curriculum learning for crowd counting. *IEEE Transactions on Cybernetics*, 2020.

[423] Qian Wang and Toby P Breckon. Crowd counting via segmentation guided attention networks and curriculum loss. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[424] Shuai Wang, Mingxia Liu, Jun Lian, and Dinggang Shen. Boundary coding representation for organ segmentation in prostate cancer radiotherapy. *IEEE Transactions on Medical Imaging*, 40(1):310–320, 2020.

[425] Shujun Wang, Lequan Yu, Kang Li, Xin Yang, Chi-Wing Fu, and Pheng-Ann Heng. Boundary and entropy-driven adversarial learning for fundus image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 102–110. Springer, 2019.

[426] Shuo Wang, Mu Zhou, Zaiyi Liu, Zhenyu Liu, Dongsheng Gu, Yali Zang, Di Dong, Olivier Gevaert, and Jie Tian. Central focused convolutional neural networks: Developing a data-driven model for lung nodule segmentation. *Medical image analysis*, 40:172–183, 2017.

[427] Xi Wang, Fangyao Tang, Hao Chen, Luyang Luo, Ziqi Tang, An-Ran Ran, Carol Y Cheung, and Pheng-Ann Heng. Ud-mil: uncertainty-driven deep multiple instance learning for oct image classification. *IEEE journal of biomedical and health informatics*, 24(12):3431–3442, 2020.

[428] Xiaofei Wang, Lai Jiang, Liu Li, Mai Xu, Xin Deng, Lisong Dai, Xiangyang Xu, Tianyi Li, Yichen Guo, Zulin Wang, et al. Joint learning of 3d lesion segmentation

and classification for explainable covid-19 diagnosis. *IEEE Transactions on Medical Imaging*, 2021.

[429] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[430] Xinggang Wang, Xianbo Deng, Qing Fu, Qiang Zhou, Jiapei Feng, Hui Ma, Wenyu Liu, and Chuansheng Zheng. A weakly-supervised framework for covid-19 classification and lesion localization from chest ct. *IEEE Transactions on Medical Imaging*, 39(8):2615–2625, 2020.

[431] Xinggang Wang, Yongluan Yan, Peng Tang, Xiang Bai, and Wenyu Liu. Revisiting multiple instance neural networks. *Pattern Recognition*, 74:15–24, 2018.

[432] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics*, 38(5):1–12, 2019.

[433] Zhao Wang, Quande Liu, and Qi Dou. Contrastive cross-site learning with redesigned net for covid-19 ct classification. *IEEE Journal of Biomedical and Health Informatics*, 24(10):2806–2813, 2020.

[434] Zheng Wang, Ying Xiao, Yong Li, Jie Zhang, Fanggen Lu, Muzhou Hou, and Xiaowei Liu. Automatically discriminating and localizing covid-19 from community-acquired pneumonia on chest x-rays. *Pattern recognition*, 110:107613, 2021.

[435] Thibaut Weise, Sofien Bouaziz, Hao Li, and Mark Pauly. Realtime performance-based facial animation. *ACM transactions on graphics (TOG)*, 30(4):1–10, 2011.

[436] Thibaut Weise, Hao Li, Luc Van Gool, and Mark Pauly. Face/off: Live facial puppetry. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer animation*, pages 7–16, 2009.

[437] Longyin Wen, Dawei Du, Pengfei Zhu, Qinghua Hu, Qilong Wang, Liefeng Bo, and Siwei Lyu. Detection, tracking, and counting meets drones in crowds: A benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7812–7821, 2021.

[438] Paul J Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[439] Udaranga Wickramasinghe, Edoardo Remelli, Graham Knott, and Pascal Fua. Voxel2mesh: 3d mesh model generation from volumetric data. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 299–308. Springer, 2020.

[440] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[441] Fuping Wu and Xiahai Zhuang. Cf distance: A new domain discrepancy metric and application to explicit domain adaptation for cross-modality cardiac image segmentation. *IEEE Transactions on Medical Imaging*, 39(12):4274–4285, 2020.

[442] Hai Wu, Wenkai Han, Chenglu Wen, Xin Li, and Cheng Wang. 3d multi-object tracking in point clouds based on prediction confidence-guided data association. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[443] Hai Wu, Qing Li, Chenglu Wen, Xin Li, Xiaoliang Fan, and Cheng Wang. Tracklet proposal network for multi-object tracking on point clouds. In *IJCAI*, pages 1165–1171, 2021.

[444] Jiajun Wu, Yinan Yu, Chang Huang, and Kai Yu. Deep multiple instance learning for image classification and auto-annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3460–3469, 2015.

[445] Jun Wu, Kaiwei Wang, Zongjiang Shang, Jie Xu, Dayong Ding, Xirong Li, and Gang Yang. Oval shape constraint based optic disc and cup segmentation in fundus photographs. In *BMVC*, page 265, 2019.

[446] Xing Wu, Cheng Chen, Mingyu Zhong, Jianjia Wang, and Jun Shi. Covid-al: The diagnosis of covid-19 with deep active learning. *Medical Image Analysis*, 68:101913, 2021.

[447] Yu-Huan Wu, Shang-Hua Gao, Jie Mei, Jun Xu, Deng-Ping Fan, Rong-Guo Zhang, and Ming-Ming Cheng. Jcs: An explainable covid-19 diagnosis system by joint classification and segmentation. *IEEE Transactions on Image Processing*, 30:3113–3126, 2021.

[448] Enze Xie, Peize Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. *arXiv preprint arXiv:1909.13226*, 2019.

[449] Enze Xie, Peize Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12193–12202, 2020.

[450] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[451] Weiyi Xie, Colin Jacobs, Jean-Paul Charbonnier, and Bram Van Ginneken. Relational modeling for robust and efficient pulmonary lobe segmentation in ct scans. *IEEE Transactions on Medical Imaging*, 39(8):2664–2675, 2020.

[452] Xuehan Xiong and Fernando De la Torre. Global supervised descent method. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2664–2673, 2015.

[453] Chenfeng Xu, Dingkang Liang, Yongchao Xu, Song Bai, Wei Zhan, Xiang Bai, and Masayoshi Tomizuka. Autoscale: Learning to scale for crowd counting. *International Journal of Computer Vision*, 130(2):405–434, 2022.

[454] Geng-Xin Xu, Chen Liu, Jun Liu, Zhongxiang Ding, Feng Shi, Man Guo, Wei Zhao, Xiaoming Li, Ying Wei, Yaozong Gao, et al. Cross-site severity assessment of covid-19 from ct images via domain adaptation. *IEEE Transactions on Medical Imaging*, 2021.

[455] Wenqiang Xu, Haiyang Wang, Fubo Qi, and Cewu Lu. Explicit shape encoding for real-time instance segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5168–5177, 2019.

[456] Wufeng Xue, Chunyan Cao, Jie Liu, Yilian Duan, Haiyan Cao, Jian Wang, Xumin Tao, Zejian Chen, Meng Wu, Jinxiang Zhang, et al. Modality alignment contrastive learning for severity assessment of covid-19 from lung ultrasound and clinical information. *Medical Image Analysis*, 69:101975, 2021.

[457] Yuan Xue, Hui Tang, Zhi Qiao, Guanzhong Gong, Yong Yin, Zhen Qian, Chao Huang, Wei Fan, and Xiaolei Huang. Shape-aware organ segmentation by predicting signed distance maps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12565–12572, 2020.

[458] Zhaoyi Yan, Yuchen Yuan, Wangmeng Zuo, Xiao Tan, Yezhen Wang, Shilei Wen, and Errui Ding. Perspective-guided convolution networks for crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 952–961, 2019.

[459] Zhaoyi Yan, Ruimao Zhang, Hongzhi Zhang, Qingfu Zhang, and Wangmeng Zuo. Crowd counting via perspective-guided fractional-dilation convolution. *IEEE Transactions on Multimedia*, 2021.

[460] Dong Yang, Ziyue Xu, Wenqi Li, Andriy Myronenko, Holger R Roth, Stephanie Harmon, Sheng Xu, Baris Turkbey, Evrim Turkbey, Xiaosong Wang, et al. Federated semi-supervised learning for covid region segmentation in chest ct using multi-national data from china, italy, japan. *Medical image analysis*, 70:101992, 2021.

[461] Jianxing Yang, Yuan Zhou, and Sun-Yuan Kung. Multi-scale generative adversarial networks for crowd counting. In *2018 24th international conference on pattern recognition (ICPR)*, pages 3244–3249. IEEE, 2018.

[462] Lin Yang, Longyu Zhang, Haiwei Dong, Abdulhameed Alelaiwi, and Abdulmotaleb El Saddik. Evaluating and improving the depth accuracy of kinect for windows v2. *IEEE Sensors Journal*, 15(8):4275–4285, 2015.

[463] Xin Yang, Yifei Wang, Ryan Byrne, Gisbert Schneider, and Shengyong Yang. Concepts of artificial intelligence for computer-assisted drug discovery. *Chemical reviews*, 119(18):10520–10594, 2019.

[464] Yifan Yang, Guorong Li, Dawei Du, Qingming Huang, and Nicu Sebe. Embedding perspective analysis into multi-column convolutional neural network for crowd counting. *IEEE Transactions on Image Processing*, 30:1395–1407, 2020.

[465] Ziduo Yang, Lu Zhao, Shuyu Wu, and Calvin Yu-Chian Chen. Lung lesion localization of covid-19 from chest ct image: A novel weakly supervised learning method. *IEEE Journal of Biomedical and Health Informatics*, 25(6):1864–1872, 2021.

[466] Jiawen Yao, Jinzheng Cai, Dong Yang, Daguang Xu, and Junzhou Huang. Integrating 3d geometry of organ for improving medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 318–326. Springer, 2019.

[467] Qingsong Yao, Li Xiao, Peihang Liu, and S Kevin Zhou. Label-free segmentation of covid-19 lesions in lung ct. *IEEE Transactions on Medical Imaging*, 2021.

[468] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018.

[469] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[470] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016.

[471] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2403–2412, 2018.

[472] Lequan Yu, Shujun Wang, Xiaomeng Li, Chi-Wing Fu, and Pheng-Ann Heng. Uncertainty-aware self-ensembling model for semi-supervised 3d left atrium segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 605–613. Springer, 2019.

[473] Zhao Yu-Qian, Gui Wei-Hua, Chen Zhen-Cheng, Tang Jing-Tian, and Li Ling-Yun. Medical images edge detection based on mathematical morphology. In *2005 IEEE engineering in medicine and biology 27th annual conference*, pages 6492–6495. IEEE, 2006.

[474] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.

[475] Amir R Zamir, Alexander Sax, Nikhil Cheerla, Rohan Suri, Zhangjie Cao, Jitendra Malik, and Leonidas J Guibas. Robust learning through cross-task consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11197–11206, 2020.

[476] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on computer vision and pattern recognition*, pages 2528–2535. IEEE, 2010.

[477] Anran Zhang, Lei Yue, Jiayi Shen, Fan Zhu, Xiantong Zhen, Xianbin Cao, and Ling Shao. Attentional neural fields for crowd counting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5714–5723, 2019.

[478] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5532–5540, 2017.

[479] Hongrun Zhang, Yanda Meng, Yitian Zhao, Yihong Qiao, Xiaoyun Yang, Sarah E Coupland, and Yalin Zheng. Dtfd-mil: Double-tier feature distillation multiple instance learning for histopathology whole slide image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[480] Jing Zhang, Deng-Ping Fan, Yuchao Dai, Saeed Anwar, Fatemeh Saleh, Sadegh Aliakbarian, and Nick Barnes. Uncertainty inspired rgb-d saliency detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[481] Kang Zhang, Xiaohong Liu, Jun Shen, Zhihuan Li, Ye Sang, Xingwang Wu, Yunfei Zha, Wenhua Liang, Chengdi Wang, Ke Wang, et al. Clinically applicable ai system for accurate diagnosis, quantitative measurements, and prognosis of covid-19 pneumonia using computed tomography. *Cell*, 181(6):1423–1433, 2020.

[482] Li Zhang, Ming Jiang, Dewan Farid, and M Alamgir Hossain. Intelligent facial emotion recognition and semantic-based topic detection for a humanoid robot. *Expert Systems with Applications*, 40(13):5160–5168, 2013.

[483] Li(*) Zhang, Xiangtai(*) Li, Anurag Arnab, Kuiyuan Yang, Yunhai Tong, and Philip HS Torr. Dual graph convolutional network for semantic segmentation. In *BMVC2019*.

[484] Li Zhang, Xiangtai Li, Anurag Arnab, Kuiyuan Yang, Yunhai Tong, and Philip HS Torr. Dual graph convolutional network for semantic segmentation. *arXiv preprint arXiv:1909.06121*, 2019.

[485] Qi Zhang, Wei Lin, and Antoni B Chan. Cross-view cross-scene multi-view crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 557–567, 2021.

[486] Ruifei Zhang, Guanbin Li, Zhen Li, Shuguang Cui, Dahong Qian, and Yizhou Yu. Adaptive context selection for polyp segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 253–262. Springer, 2020.

[487] Shanghang Zhang, Guanhang Wu, Joao P Costeira, and José MF Moura. Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras. In *Proceedings of the IEEE international conference on computer vision*, pages 3667–3676, 2017.

[488] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 589–597, 2016.

[489] Zhijie Zhang, Huazhu Fu, Hang Dai, Jianbing Shen, Yanwei Pang, and Ling Shao. ET-Net: A generic edge-attention guidance network for medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 442–450. Springer, 2019.

[490] Zhuo Zhang, Feng Shou Yin, Jiang Liu, Wing Kee Wong, Ngan Meng Tan, Beng Hai Lee, Jun Cheng, and Tien Yin Wong. ORIGA-light: An online retinal fundus image database for glaucoma analysis and research. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 3065–3068. IEEE, 2010.

[491] Chen Zhao, Yan Xu, Zhuo He, Jinshan Tang, Yijun Zhang, Jungang Han, Yuxin Shi, and Weihua Zhou. Lung segmentation and automatic detection of covid-19 using radiomic features from chest ct images. *Pattern Recognition*, page 108071, 2021.

[492] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2881–2890, 2017.

[493] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 267–283, 2018.

[494] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3D human pose regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3425–3435, 2019.

[495] Muming Zhao, Chongyang Zhang, Jian Zhang, Fatih Porikli, Bingbing Ni, and Wenjun Zhang. Scale-aware crowd counting via depth-embedded convolutional neural networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(10):3651–3662, 2019.

[496] Muming Zhao, Jian Zhang, Chongyang Zhang, and Wenjun Zhang. Leveraging heterogeneous auxiliary tasks to assist crowd counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12736–12745, 2019.

[497] Rongchang Zhao, Xuanlin Chen, Xiyao Liu, Zailiang Chen, Fan Guo, and Shuo Li. Direct cup-to-disc ratio estimation for glaucoma screening via semi-supervised learning. *IEEE journal of biomedical and health informatics*, 24(4):1104–1113, 2019.

[498] Tianyi Zhao, Kai Cao, Jiawen Yao, Isabella Nogues, Le Lu, Lingyun Huang, Jing Xiao, Zhaozheng Yin, and Ling Zhang. 3d graph anatomy geometry-integrated network for pancreatic mass segmentation, diagnosis, and quantitative patient management. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13743–13752, 2021.

[499] Yitian Zhao, Lavdie Rada, Ke Chen, Simon P Harding, and Yalin Zheng. Automated vessel segmentation using infinite perimeter active contour model with hybrid region information with application to retinal images. *IEEE Transactions on Medical Imaging*, 34(9):1797–1807, 2015.

[500] Yu Zhao, Fan Yang, Yuqi Fang, Hailing Liu, Niyun Zhou, Jun Zhang, Jiarui Sun, Sen Yang, Bjoern Menze, Xinjuan Fan, et al. Predicting lymph node metastasis using histopathological images based on multiple instance learning with deep graph convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4837–4846, 2020.

[501] Aoxiao Zhong, Xiang Li, Dufan Wu, Hui Ren, Kyungsang Kim, Younggon Kim, Varun Buch, Nir Neumark, Bernardo Bizzo, Won Young Tak, et al. Deep metric learning-based image retrieval system for chest radiograph and its clinical applications in covid-19. *Medical Image Analysis*, 70:101993, 2021.

[502] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[503] Bolei Zhou, Xiaogang Wang, and Xiaoou Tang. Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2871–2878. IEEE, 2012.

[504] Longxi Zhou, Zhongxiao Li, Juexiao Zhou, Haoyang Li, Yupeng Chen, Yuxin Huang, Dexuan Xie, Lintao Zhao, Ming Fan, Shahrukh Hashmi, et al. A rapid, accurate and machine-agnostic segmentation and quantification method for ct-based covid-19 diagnosis. *IEEE Transactions on Medical Imaging*, 39(8):2638–2652, 2020.

[505] Tianyi Zhou, Le Zhang, Du Jiawei, Xi Peng, Zhiwen Fang, Zhe Xiao, and Hongyuan Zhu. Locality-aware crowd counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[506] Yuan Zhou, Jianxing Yang, Hongru Li, Tao Cao, and Sun-Yuan Kung. Adversarial learning for multiscale crowd counting under complex scenes. *IEEE transactions on cybernetics*, 2020.

[507] Yuxiang Zhou, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Dense 3d face decoding over 2500fps: Joint texture & shape convolutional mesh decoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1097–1106, 2019.

[508] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11. Springer, 2018.

[509] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. *IEEE Transactions on Medical Imaging*, 2019.

[510] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.

[511] Xiaofeng Zhu, Bin Song, Feng Shi, Yanbo Chen, Rongyao Hu, Jiangzhang Gan, Wenhai Zhang, Man Li, Liye Wang, Yaozong Gao, et al. Joint prediction and time estimation of covid-19 developing severe symptoms using chest ct scan. *Medical image analysis*, 67:101824, 2021.