Attention Guided Multi-Task Learning for Surface defect identification

Vignesh Sampath[*], Iñaki Maurtua, Juan José Aguilar Martín, Andoni Rivera, Jorge Molina and Aitor Gutierrez

Abstract—**Surface defect identification is an essential task in the industrial quality control process, in which visual checks are conducted on a manufactured product to ensure that it meets quality standards. Convolutional Neural Network (CNN) based surface defect identification method has proven to outperform traditional image processing techniques. However, the real-world surface defect datasets are limited in size due to the expensive data generation process and the rare occurrence of defects. To address this issue, this paper presents a method for exploiting auxiliary information beyond the primary labels to improve the generalization ability of surface defect identification tasks. Considering the correlation between pixel level segmentation masks, object level bounding boxes and global image level classification labels, we argue that jointly learning features of the related tasks can improve the performance of surface defect identification tasks. This paper proposes a framework named Defect-Aux-Net, based on multi-task learning with attention mechanisms that exploit the rich additional information from related tasks with the goal of simultaneously improving robustness and accuracy of the CNN based surface defect identification. We conducted a series of experiments with the proposed framework. The experimental results showed that the proposed method can significantly improve the performance of state-of-the-art models while achieving an overall accuracy of 97.1%, Dice score of 0.926 and mAP of 0.762 on defect classification, segmentation and detection tasks.**

Index Terms—**Deep learning, defect classification, defect detection, defect segmentation, machine vision, multi-task-learning, quality control, surface defect detection.**

I. INTRODUCTION

AUTOMATED visual inspection plays an important role in industrial informatics based decision-making systems in

S. Vignesh, M. Iñaki, R. Andoni, M. Jorge and G. Aitor are with Tekniker, Autonomous and Intelligent Systems unit, Gipuzkoa 20600, Spain (e-mail: vignesh.sampath@tekniker.es; inaki.maurtua@tekniker.es; andoni.rivera@tekniker.es; jmolina@tekniker.es; aitor.gutierrez@tekniker.es).
A.M. Juan José is with the Department of Design and Manufacturing Engineering, University of Zaragoza, Zaragoza 50009, Spain (e-mail: jaguilar@unizar.es).

various industries, including steel manufacturing companies, automotive industries, electronic manufacturing, and pharmaceutical companies. The correct, consistent, and early detection of surface defects can make it possible to detect defective products early in the manufacturing process, which leads to time and cost savings. Inspection procedures for detecting such defects are usually performed using non-destructive testing (NDT) methods. NDT procedure is a combination of various inspection steps used to identify discontinuities or defects in a product without causing damage to its usability. The most frequently used industrial NDT methods are Visual optic testing, Radiography, X-ray vision, Ultrasonic imaging, Dye penetrant testing, Magnetic particle testing, and Infrared thermal imaging. The testing procedure for each of these methods involves several steps, all of which can be easily automated. However, the final step of visual inspection is more complex in terms of automation and remains primarily a manual process performed by operators.

The traditional machine-vision system relies on a hand-crafted features such as color, contrast, texture, edges, foreground background statistics, etc. followed by machine learning classifiers such as support vector machines, decision tree or K-Nearest Neighbors. Consequently, hand-crafted features extraction plays an important role in classical approaches. However, these features are not robust and suited for different tasks, which lead to long development cycles. Deep learning methods, on the other hand, learn the relevant features directly from the raw data, without the need for handcrafted feature representations. In recent years, Convolutional Neural Network (CNN) has achieved and even surpassed human-level performance on computer vision tasks such as image classification. The key difference between CNN and traditional machine-vision algorithms is that CNN automatically detects significant features without any human supervision which made it the most widely used. A fascinating feature of CNN is its ability to take advantage of the spatial or temporal correlation of image data. There are three main problem categories for image recognition tasks using CNN: classification, segmentation, and object detection. Classification task aims to classify an image into a certain category. Starting with the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winning architecture of AlexNet [1], a series of increasingly complex architectures including ResNet [2], Inception [3], Densenet [4], and EfficientNet [5] have been proposed in the literature for the classification task. Object detection is a task that localizes an

object using a bounding box. Some of the notable object detection algorithms include Fast R-CNN [6], Faster R-CNN, Mask R-CNN [7], Single Shot Detection (SSD) [8], You Only Look Once (YOLO) [9], etc. Segmentation is the task of performing pixel-by-pixel classification. Several segmentation algorithms have been proposed in the literature including fully convolutional networks, encoder-decoder based approaches [10], multi-scale and pyramid architectures [11], etc.
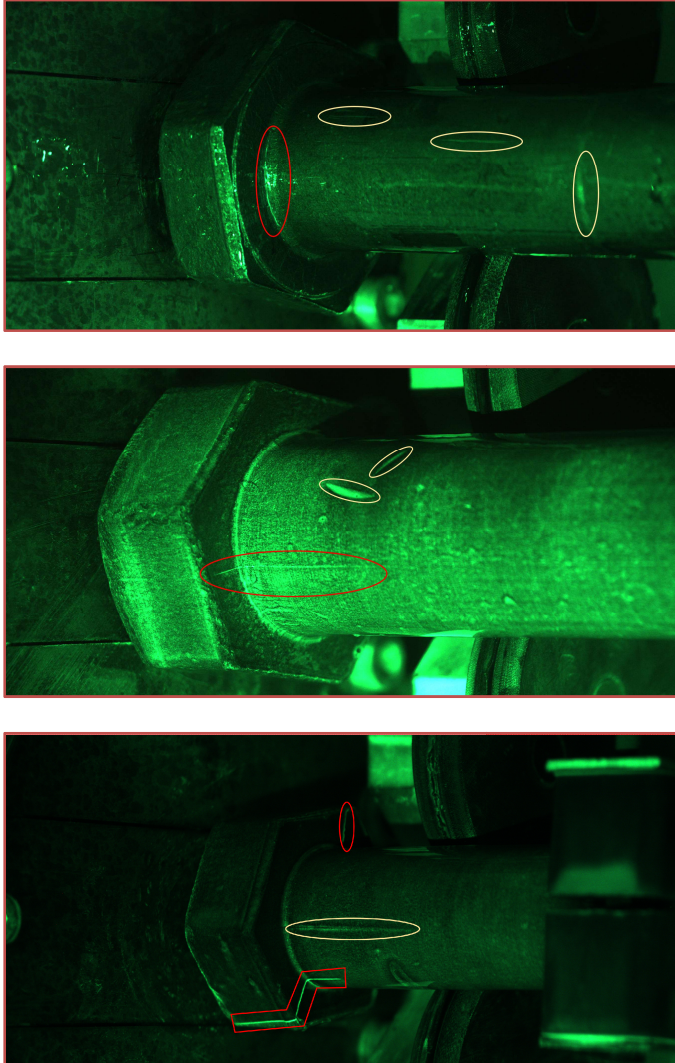

Fig. 1. Magnetic particle inspection on threaded fasteners of different surface finish (TekErreka dataset). Surface defects are marked by red circles and noise due to magnetic particle depositions are marked in yellow.

However, industrial visual inspection systems barely utilized the potential of those complex architectures due to several reasons [12]. One of the main reasons is that the continuous improvement in industrial processes has resulted in fewer and fewer defective samples, or the number of defective samples is very limited [13]. This problem of learning from a limited number of samples is usually referred to as the small sample problem, which can easily lead to poor generalization ability of the trained model [14]. In addition, the target surface defects have different scales, making the deep learning models even more challenging to identify the small sized defects. On the one hand, visual appearance of the real-world surfaces defects varies with type of materials, imaging conditions, and camera position. On the other hand, it is challenging to distinguish tiny defects from the noise or non-defect components within an image (as shown in Fig. 1). Hence the appearance of false positives in a defect free image is an inevitable circumstance. Furthermore, real time applications of complex CNN models are extremely limited due to the long inference time and the resulting higher computational resource and power consumption.

To address these limitations, we present a novel universal architecture that integrates classification, segmentation, and detection of surface defects in a single network. Our architecture, Defect-Aux-Net, is primarily motivated by a multi-task learning (MTL) scheme that exploits useful information from related learning tasks to help mitigate the problem of data scarcity. The proposed architecture is based on FPN-semantic-segmentation [11] with the additional tasks of defect classification and detection to improve the generalization ability by utilizing the image level information as an inductive bias. Specifically, we developed a new multi-task learning network based on FPN, where the classification task is carried out in the bottom-up pathway of the network and segmentation is performed in the top-down pathway of the network. To create a bounding box we employ two sub-networks in the top-down pathway, where one subnet determines the class associated with bounding box and the other performs the regression to adjust the bounding box position.

The FPN-based Feature Extractor in the proposed network allows surface defects to be recognized at vastly different scales by efficiently sharing features between image regions. We further introduce the positional and the channel attention mechanisms that focus on learning the features of small surface defects to improve the robustness of detecting small defects surrounded by complex background.

We evaluate our model on TekErreka, and Severstal [15] surface defect datasets, with defect classification, segmentation, and detection tasks. Experimental results demonstrate that jointly learning features of related tasks can improve the performance of all tasks.

Overall, the contributions of our work are as follows:
1) Firstly, we propose a Defect-Aux-Net model architecture, which can perform classification, segmentation, and detection of surface defects in a single network. Compared with the existing state-of-the-art CNN models, this architecture is lightweight and compact in terms of model parameters. From the model training point of view employing fewer parameters in the architecture enables model to efficiently learn potential surface defects from a smaller number of labelled examples.
2) In contrast to existing single task learning, our proposed multi-task learning in surface defect detection facilitates the model to learn useful representations of the data by exploiting shared information from related tasks.

3) Considering surface defect detection with complex background, the positional and the channel attention mechanisms are incorporated to amplify target features and to reduce the influence of background noise.

4) The proposed model is compact and efficient with state-of-the-art performance that meets the computational resource requirements of the real-time inference speed.

II. RELATED WORK

A large and growing body of literature has explored the use of CNN for surface defect identification. Kim et al. [16] adopted few-shot learning technique with Siamese Neural Network using CNN, which aims to classify surface defects with a limited number of training images. Lin et al. [17] employed class activation mapping technique in CNN to simultaneously achieve defect classification and localization tasks in LED chip defect inspection process. Tao et al. [18] designed cascaded autoencoder (CASAE) architecture to segment and localize defect region. The proposed architecture transforms the input image into a mask prediction and then defect regions of segmented mask is classified to their specific classes. Jing et al. [19] combined autoencoder with fully connected network (FCN) to detect keyboard light leakage defect from mere dust. Jian et al. [20] leveraged Generative adversarial network (GAN) to exaggerate the tiny defects within the images to improve the accuracy of different classifiers. Zheng et al. [21] proposed a 3-stage model for rail surface and fastener defect detection. At the first stage, YOLOV5 framework is employed to localize the rail and fasteners. Then, an object detection model based on Mask-RCNN is used to detect the surface defect of the rail surface. At the final stage, the Resnet architecture is utilized to classify defects of the fasteners. To detect defects at different scale, Xu et al. [22] used a pre-trained ResNet model to extract the multi-scale features and fuse them using a multilevel feature fusion network (MFN). In [23], U-Net and residual U-Net architectures were used for the fine-grained segmentation of surface defects on a steel sheet. The main drawback of these methods is that the model needs a large amount of annotated data and hence the localization of defect is very coarse in the real-time scenario.

III. PROPOSED METHOD

A. Network architecture

Our proposed network is inspired by two deep learning architectures that are widely used: Feature pyramid Network (FPN) and ResNet-50. Recognizing surface defects at vastly different scales is a fundamental challenge in industrial machine vision system. For this reason, we use FPN that uses a pyramidal hierarchy of convolutional filters to extract feature pyramids at different scales. FPN consists of two pathways: bottom-up and top-down. The bottom-up pathway also known as encoder, is the typical convolutional neural network, which can be any image classifier for feature extraction. As we go up, the encoder gradually decreases the spatial resolution, while building high level feature maps. The

top-down pathway is connected to the bottom-up pathway through lateral connections for efficient multi-scale feature fusion. It is designed to enhance the feature maps from the bottom-up pathway and build semantically strong feature maps at multiple scales by double upscaling. As a result, the feature pyramid has rich semantics at all levels because the lower semantic features are interconnected to the higher semantics.

1) Bottom-up pathway

We tested several standard image classification architectures to select the core model, and finally chose ResNet-50 as the backbone. ResNet-50 has shown great performance for surface defect classification, segmentation and detection tasks. ResNet-50 architecture has the advantage of using a stride of two for each scale reduction, which makes it easier to incorporate ResNet-50 into FPNs when we need to upscale feature maps in top-down pathway. Furthermore, Resnet-50 is a relatively small network based on modern standards; therefore, it is suitable for our limited labeled data problem. However, existing ResNet-50 feature pyramids have two problems in the way they apply convolution operations to the input features. Firstly, the receptive field of the encoder has the information only about the local region, so the global information is lost. Secondly, the feature maps constructed from the learned weights are given equal magnitude of importance, but some feature maps are more important for the next layers than others. For instance, a feature map that contains edge information of the defects might be more important than another feature map that has background texture information (as shown in Fig. 3.). Thus, to incorporate channel attention we adopt Squeeze-and-Excitation (SE) module [24] in the encoder. SE module consists of three components 1. Squeeze, 2. Excite and 3.Scale components.
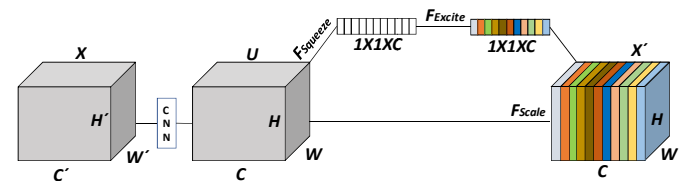

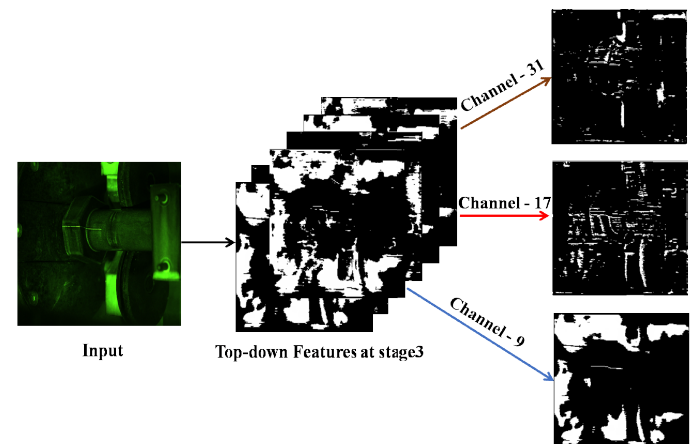Fig. 2. Structure of Squeeze and Excite module.

Fig. 3. Sample features in different channels of top-down pathway at stage 3.
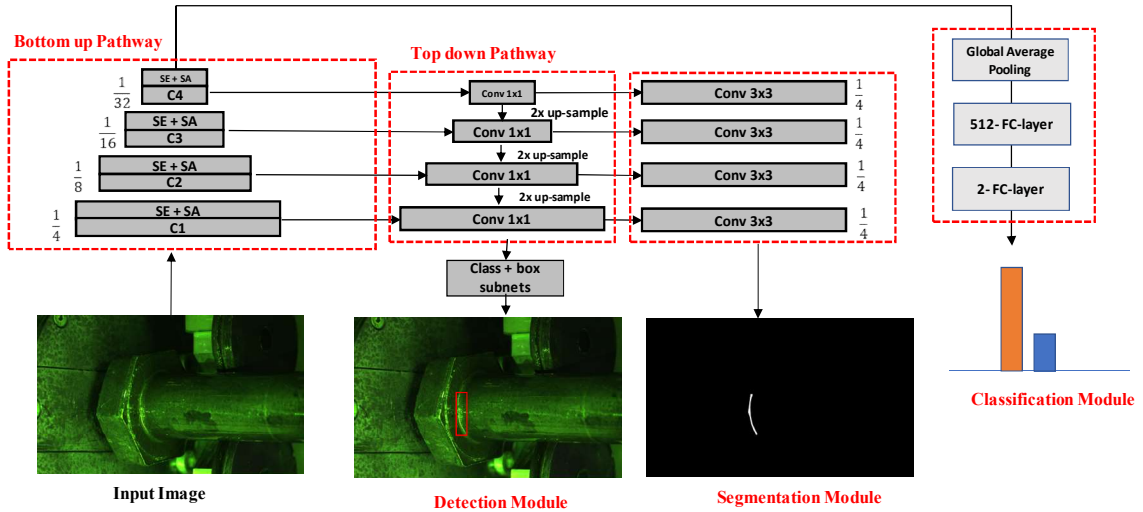
Fig. 4. An overview of proposed Defect-Aux-Net architecture. It mainly composed of classification, segmentation and detection module that incorporates multi-task loss function.

The main goal of the squeeze component is to extract global information from each of the channels c in a feature block U. The global information is acquired by applying a global average pooling operation across their spatial dimensions ($H \times W$) for each channel U_c of U to obtain global statistics ($1 \times 1 \times C$). Mathematically, squeeze operation can be represented as:

$$z_c = F_{squeeze}(U_c) = \frac{1}{H \times W}\sum_{m=1}^{H}\sum_{n=1}^{W} U_c(m,n) \qquad (1)$$

After obtaining global information from the squeeze component, the excite component generate a set of weights for each channel. It uses a fully connected Multi-Layer Perceptron (MLP) bottleneck structure to dynamically calibrate the weights. This MLP bottleneck has two fully connected layers with sigmoid activation as the output layer. Output of the excitation component can formally be represented by the following equation:

$$s = F_{excite}(z,W) = \sigma(g(z,W)) = \sigma(W_2\rho(W_1,z)) \qquad (2)$$

Where σ is a Sigmoid operation, ρ is ReLU operation, z is the output from the squeeze component, W_1 and W_2 refers to weights of the two fully connected layers. Subsequently each channel in the feature map is scaled by a simple element-wise multiplication of the input feature map and weights obtained from the excite component (as shown in Fig. 2).

Surface defects only appear in some parts of the image but not the whole image. Unlike the conventional Resnet-50 architecture, which gives equal importance to each region in an image, the spatial attention reduces background interferences by assigning a weight to each pixel in the feature map.

The spatial attention focuses on the most relevant parts of the feature maps in the spatial dimension. The working principle of our spatial attention mechanism is as follows. Given feature block U, we use average and max-pooling

operations along the channel axis and concatenate them to generate an efficient feature map summary M. A convolutional layer followed by sigmoid operation is then performed on the feature M to produce spatial attention map (as shown in Fig. 5).
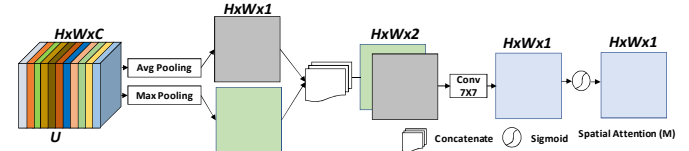

Fig. 5. Structure of Spatial Attention module.

Resnet uses four modules consisting of residual blocks, each of which uses two blocks, Identity (ID) blocks and convolution blocks, depending on whether the input / output dimensions are the same or different. We arrange SE and SA module in series and integrate into residual block (as shown in Fig. 6)
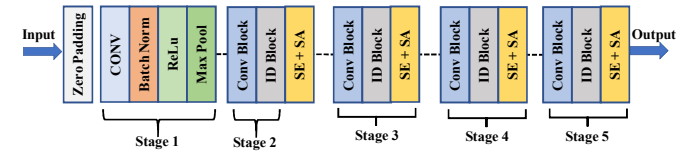

Fig. 6. FPN Bottom-Up structure with attention module

2) Top-down pathway

Deep features from bottom-up pathway are upsampled by convolutions and bilinear up-sampling operations until all the feature maps reach ¼ scale. Attention module outputs from bottom-up pathway $\{C_2, C_3, C_4, C_5\}$ are fused to top-down pathway through lateral connections for an efficient multi-scale feature fusion. Firstly, 1 x 1 convolutional filter is applied to the feature maps $\{C_2, C_3, C_4, C_5\}$ to get a fixed number of channels and then merged with the corresponding top-down feature map by element-wise addition. Finally, the outputs are summed and then transformed into a pixel-wise output (as shown in Fig. 4).

3) Segmentation branch

The segmentation branch from top-down pathway aims at classifying pixels into a set of pre-defined classes. The pixels corresponding to background are far numerous than pixels of surface defects in the real-world dataset, which causes the model to be biased toward the background element. To address the pixel wise class imbalance, we employ Dice loss, which uses Dice coefficient to calculate overlapping of the pixels of the predicted mask with the ground truth label. Mathematically Dice loss function is defined as:

$$L_{seg} = 1 - \frac{2y\hat{y}+1}{y+\hat{y}+1} \tag{3}$$

Where, y_i is the ground truth label, $\hat{y}_i$ is the predicted label. The value of Dice coefficient ranges from 0 to 1, where 1 indicates the perfect and complete overlap of pixels.

4) Classification branch

The output of the bottom-up pathway encodes the rich abstract feature representations of the input image. Hence, we utilize the spatial average of the feature maps from the bottom-up pathway via a global average pooling layer and then the resulting feature vector is fed into the sigmoid or softmax layer depending on classification type. We employ binary cross-entropy (BCE) as classification loss function. Mathematically our classification loss is defined as:

$$L_{class} = \frac{1}{k}\sum_1^k CE(y_i, \hat{y}_i) \tag{4}$$

Where, y_i is the ground truth label, $\hat{y}_i$ is the predicted label of ith sample, k is the total number of samples. CE is the binary cross entropy function.

5) Object Detection branch

We extract bounding boxes and its associated classes by employing box regression and classification subnets at each level of top-down pathway. The classification subnet predicts the probability of defect presence at each spatial location of an input image. The box regression subnet is attached to top-down pathway in parallel to classification subnet for the purpose of regressing offset from each anchor box to the ground truth bounding boxes. To handle class imbalance problem, we adopt focal loss [25], an improved version of cross entropy to focus learning on hard negative examples. It is defined as:

$$L_{detection} = -\alpha_t(1-p_t)^\gamma \log(p_t) \tag{5}$$

Where, α_t is the weight parameter per class and γ is the hyper parameter focuses on hard negative samples. We choose $\alpha_t=0.25$ and $\gamma= 4$ as suggested in [26].

B. Loss Function

Our proposed method combines three loss functions from the classification, segmentation and detection tasks which provide mutual sources of inductive bias for each task. Specifically, the segmentation and detection loss functions

signal back to the entire model (bottom-up and top-down pathway), while the classification loss signals back only to bottom-up pathway. We combine and weight the three losses into a multi-task loss L_M to leverage the heterogeneous annotations and jointly optimize multiple tasks as follows:

$$L_M = \beta L_{class} + \beta_1 L_{seg} + \beta_2 L_{detection} \tag{6}$$

Where, β, β_1, and β_2 are weight parameters. We tested with different combinations of weight parameters and found that $\beta = \beta_1 = \beta_2 = 1$ yields the best result for all the tasks.

IV. EXPERIMENTS

A. Datasets

In this paper, we evaluate our framework on real-world surface defect identification problems. We use two challenging datasets with increasing resolutions and complexities, Severstal steel sheet [15] and TekErreka steel fastener defect datasets. Severstal, the largest steel and steel-related mining company, has recently published the largest industrial steel sheet surface defect dataset, which contains pixel-wise masks annotated by their technical experts. The dataset contains 12568 grayscale images of size of 1600×256. Each image in the dataset has the possibility of having either no defects, a single defect, or multiple defects divided into four classes. Fig. 7 show the example of steel defect images on Severstal datasets. We randomly select 10% and 20% of the 12,568 original images as the validation and test data. The main challenge with this dataset is that the inter-class similarities between defective and defect-free examples are very high.
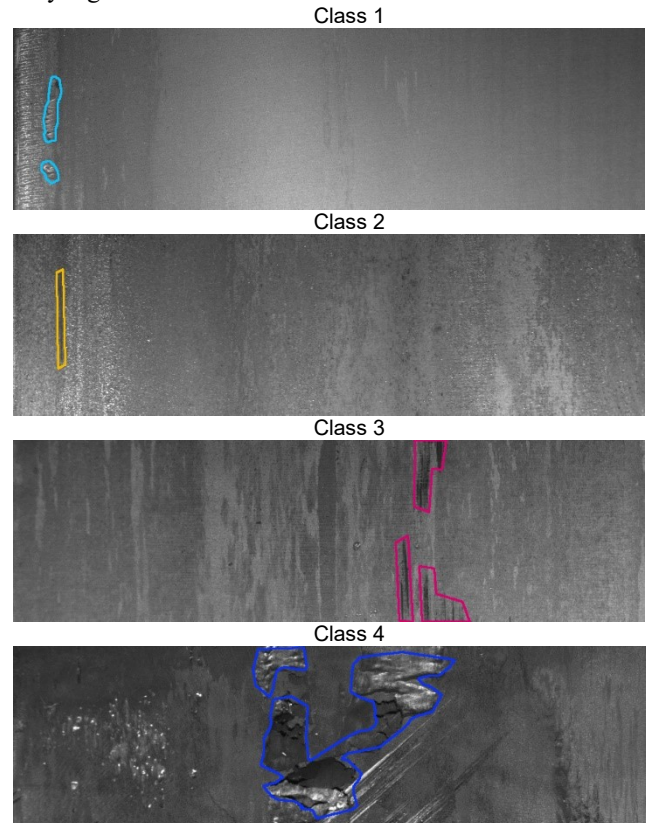

Fig. 7. Sample images of Severstal steel with 4 classes of defect.

The TekErreka dataset is a self-collected steel fastener surface defect dataset based on magnetic particle inspection procedure. The magnetic particle inspection is an excellent method to investigate near surface defects in steel fasteners. The basic principle is to magnetize a steel fastener parallel to its surface. If the fastener is free from defects the magnetic field lines run within the fastener and parallel to its surface. In case of magnetic inhomogeneity, for instance, near cracks, the magnetic field lines will locally leave the surface and a leakage field occurs. When a suspension of ferromagnetic particles is applied onto the test piece surface the magnetic particles will run off at defect free areas. In the places of leakage fields the magnetic particles are attracted and clustered together thus indicating the location of the defect. The surface defects can be visible under ultra violet light. We acquired TekErreka dataset from a magnetic particle inspection apparatus located at the Erreka Fastening solutions. The defects in the TekErreka dataset differ in their size, shape, location and materials type and thus cover several scenarios in real time defect detection. The difficulty in this dataset lies in the similarity of defects and noise due to magnetic particles deposition on defect free surface of the fasteners. There are many factors responsible for the noise component, which include magnetic particle size, the amount of magnetic particle used, ultra-violet light present, etc. The original examples are directly stored in a database as RGB images of size 2464 x 2056. It has 450 positive and 1200 negative examples. We split TekErreka dataset into training and testing sets: 80% for training and 20% for evaluation of the model performance.

B. Preprocessing

We resized the images of Severstal dataset to 128x800 and TekErreka dataset to 600x600. To keep the pixel values in same scale, we normalized the images using min-max standardization. It rescales raw pixel values to range of 0 and 1. This helps the optimizer not get stuck taking steps that are too large in one dimension, or too small in another.

C. Data Augmentation

To improve the diversity of the training set we apply random but realistic data augmentation such as rotation, vertical/horizontal flips, zoom, shear and channel shifts.

D. Training details

The Defect-Aux-Net is implemented using the Tensorflow framework. All the experiments are run on Google-cloud TPU V2 infrastructure which contains 8 cores with 64 GB memory. The network is optimized with the Adam optimizer and trained with a batch size of 128 for 50 epochs. We adopt one cycle policy [27] to find an optimal learning rate.

E. Evaluation Metrics

The classification results are evaluated using precision, recall, F1-score and binary accuracy.

$$Recall \; = \frac{TP}{TP+FN} \qquad (7)$$

$$Precision = \frac{TP}{TP+FP} \qquad (8)$$

$$F1 \; Score \; = \frac{2.(Precision.Recall)}{(Precision+Recall)} \qquad (9)$$

$$Accuracy = \frac{TP+TN}{TP+FP+TN+F} \qquad (10)$$

Where TP, TN, FP and FN denote true positive (correctly identified surface defects), true negative (correctly identified non defect images), false positive (erroneously classified images as surface defect) and false negative (erroneously classified images as non defect). Precision measures the percentage of images with surface defect that are correctly classified, while recall is the ratio of correctly classified images with surface defect to all images with surface defect. F1- score can be interpreted as harmonic mean of precision and recall. The overall performance of the classification task is measured by its accuracy.

The segmentation results are evaluated using Dice score and Intersection-over-Union (IoU), which quantify the percentage overlap between the predicted and target binary masks. To evaluate defect detection results, we used the mean average precision (mAP) that compares the detected bounding box to the ground truth bounding box and returns a score.

F. Experiments on Defect Segmentation

We performed series of experiments on TekErreka dataset to test the effectiveness of different loss functions. First, we trained Defect-Aux-Net using BCE, and Dice loss alone as the segmentation loss. Then it was trained using a combination of loss functions. The results are shown in TABLE I.

TABLE I
PERFORMANCE OF THE PROPOSED APPROACH ON LOSS VARIANTS FOR THE DEFECT SEGMENTATION TASK

Loss Function	IoU	Dice
BCE	0.892	0.911
Dice	**0.903**	**0.926**
Jaccard	0.900	0.913
Dice + BCE	0.901	0.920
Jaccard + BCE	0.899	0.912

Using Dice loss alone yielded more accurate results than using combination of losses. Additionally, Dice loss function assisted our model to converge faster. We use Dice loss function throughout rest of the experiments.

To verify the effectiveness of segmentation task using multi-task learning strategy, we compared the proposed multi-task learning network (Defect-Aux-Net) against the following network with same bottom-up backbone (Resnet50 + SE + SA attention module):

1. FPN [11]: This is the original FPN architecture without multi-task learning strategy and serves as our baseline.
2. UNet [10]: This network uses an encoder for multi-level feature extraction and a decoder that scales them up and combines multi-level feature through stacking.
3. LinkNet [28]: This is similar to UNet with the difference of replacing stacking operation with addition in skip connections.

4. PSPNet [28]: Pyramid scene parsing Network uses pyramid pooling module for multi-scale feature extraction
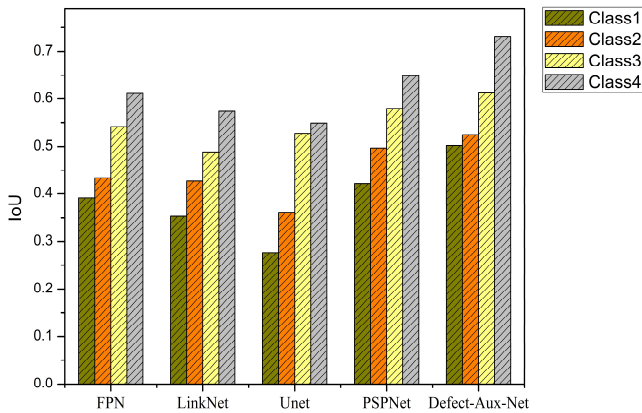

Fig. 8. IOU comparison between the state-of-the-art segmentation methods and the proposed approach on each type of defect classification.
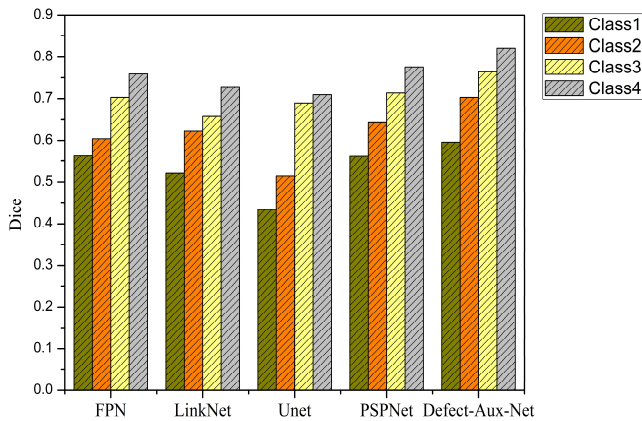

Fig. 9. Dice score comparison between the state-of-the-art segmentation methods and the proposed approach on each type of defect classification.

Based on the experimental results, we observed that the proposed multi task learning strategy achieves better segmentation performance as compared to the state-of-the-art segmentation models. The Dice and IoU scores of the various segmentation models on Severstal dataset are depicted in Fig. 8 and Fig. 9.

TABLE II
PERFORMANCE OF THE COMPETING MODELS ON THE TEKERREKA DATASET

Model	Iou	Dice
FPN [11]	0.881	0.902
LinkNet [28]	0.876	0.895
Unet [10]	0.832	0.856
PSPNet [29]	0.885	0.917
Defect-Aux-Net	**0.903**	**0.926**

We observe that Defect-Aux-Net is able to achieve higher scores for all classes as compare to the other segmentation

models. TABLE II shows the performance of the various networks on TekErreka dataset. Experimental results from TABLE II showed that the proposed multi-task-learning can improve the performance of its corresponding single task model. Taking advantages of the classification-guidance module, Defect-Aux-Net avoids the over-segmentation of defects in complex background.

G. Experiments on Defect Classification

We evaluated and compared the classification task performance of proposed approach with the state-of-the-art deep learning architectures. While evaluating classification task, other two modules: segmentation and detection are removed from the network. Results of the experiments are summarized in

TABLE III. It can be noted that the most errors are due to false positives. The visual similarity between defects and surface noise leads to false positive errors. Notably, Defect-Aux-Net obtains overall accuracy of at least 92.9% and at most 99.4% across all defect types on Severstal dataset. Based on the experimental results, we observe that the proposed multi-task learning approach achieves a surpassing performance over the other models. Also, it is evident that incorporating segmentation task improves the performance of classification task and vice-versa.
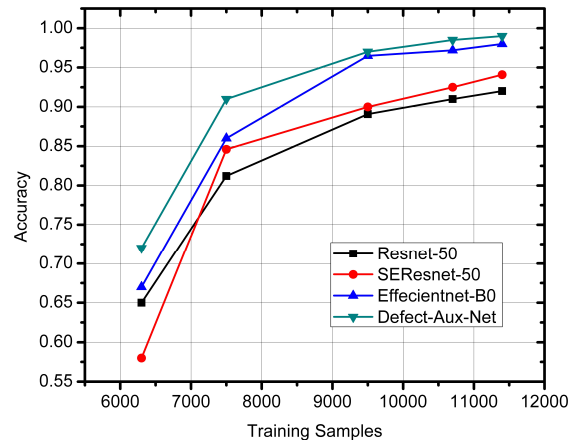

Fig. 10. Training data size vs. classification accuracy of Severstal dataset.

To assess the effectiveness of the proposed approach against limited data problem, we removed part of the training data and conducted series of experiments leaving 90%, 75%, and 50% from the training data. The effect of training data size on its accuracy is shown in Fig. 10. The proposed Defect-Aux-Net showed a consistent performance even when only 50% of the original training data is used in training. As seen, the proposed multi-task loss function greatly improves performance of the classification task by talking image, pixel, and map level optimization into the consideration.

To verify the importance of the attention mechanisms in Defect-Aux-Net, we compared accuracy the network with and without spatial and channel attention mechanism (squeeze and excite) on TekErreka dataset, as shown in TABLE IV. Further, we experimented with inserting combination of both spatial

and channel attention mechanisms.

TABLE III
COMPARISION OF PERFORMANCE OF DEFECT-AUX-NET AND
STATE-OF-THE-ART CLASSIFICATION MODELS

Model	Dataset	Class	Recall	Precision	F1-Score	Accuracy
Resnet-50 [2]	Severstal	Class1	0.454	0.403	0.427	0.831
		Class2	0.591	0.533	0.561	0.958
		Class3	0.918	0.847	0.881	0.811
		Class4	0.857	0.852	0.854	0.963
	TekErreka	Class1	0.759	0.979	0.855	0.949
SEResnet-50 [24]	Severstal	Class1	0.508	0.556	0.531	0.875
		Class2	0.617	0.580	0.598	0.970
		Class3	0.980	0.816	0.891	0.817
		Class4	0.559	0.940	0.701	0.940
	TekErreka	Class1	0.803	0.968	0.878	0.955
Effecient net-B0 [5]	Severstal	Class1	0.891	0.859	0.875	0.964
		Class2	0.872	0.732	0.796	0.984
		Class3	0.943	0.963	0.953	0.929
		Class4	0.946	0.924	0.935	0.983
	TekErreka	Class1	0.858	0.928	0.892	0.958
Defect-Aux-Net (ours)	Severstal	Class1	0.891	0.926	0.908	0.975
		Class2	0.957	0.900	0.928	0.994
		Class3	0.982	0.929	0.955	0.929
		Class4	0.946	0.940	0.943	0.985
	TekErreka	Class1	0.887	0.939	0.912	0.971

TABLE IV
EFFECT OF USING ATTENTION MECHANISMS ON TEKERREKA
DATASET

Model	Accuracy	Parameters (M)
Defect-Aux-Net (without attentions)	0.962	33.2
Defect-Aux-Net (with SE attention)	0.968	35.7
Defect-Aux-Net (Spatial attention)	0.963	33.5
Defect-Aux-Net (with SE + Spatial attention)	**0.971**	**36.2**

H. Experiments on Defect Detection

The proposed is compared with other object detection algorithms on the TekErreka dataset. The comparative models include SSD [8], RetinaNet [25], and cascade R-CNN [30]. Fig. 11 shows the mAP scores of the various detection models for the TekErreka dataset. We observe that Defect-Aux-Net is able to achieve higher mAP score as compared to the alternative networks. The mAP of the proposed algorithm is 17.95%, 43.77%, and 26.03% higher than that of RetinaNet, SSD and Cascade RCNN.
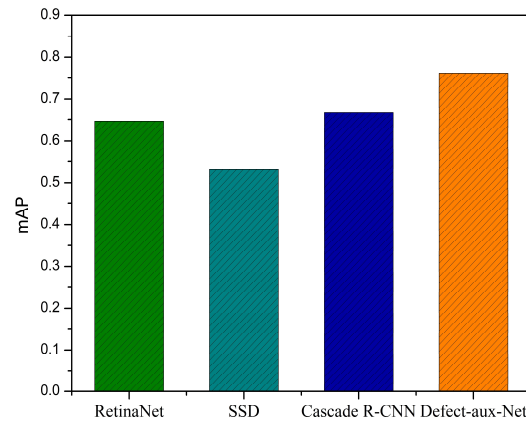

Fig. 11. mAP comparison between the state-of-the-art detection models and the proposed.

I. Inference Time

In addition to the model performance, we attempt to determine the effectiveness of multi-task learning framework on the inference time. We compared inference time of the proposed approach with conventional single task network where each task requires a separate pass through the network during inference. All the inference time was measured using a computer with an Intel Core processor. The CPU specification is summarized in TABLE V.

TABLE V
SYSTEM SPECIFICATION

CPU Specification	
CPU Processor type	Intel(R) Xeon(R)
Processor Base Frequency	2.20 GHz
Total Cores	1

From the TABLE VI, we can see that our proposed framework allows for a 57.1% reduction in the model size by solving different tasks jointly rather than independently. Compared to the single task network, the inference time of our proposed network reduce by 45.5%.

TABLE VI
COMPARISION OF INFERENCE TIME OF DEFECT-AUX-NET AND
BASELINE MODEL

Model	Task	Task Name	Inference time CPU (s)	Parameters (M)
Single Task Networks	Task 1	Classification (ResNet-50)	0.0654	23.5
	Task 2	Segmentation (ResNet-50 FPN)	0.1106	26.9
	Task 3	Detection (ResNet-50 RetinaNet)	0.1780	34.0
	Total	Classification + Segmentation + Detection	0.3540	84.4
Multitask Network	Multitask	Classification + Segmentation + Detection (Defect- Aux-Net)	**0.1927**	**36.2**

V. Discussion

By incorporating multi-task learning strategy, our proposed Defect-Aux-Net improves the performance of defect classification, segmentation and detection tasks. Intuitively multi-task deep learning system can provide regularization effects to the multi-scale feature learning and thus improve the performance as opposed to the single task algorithms. Also, the multi-task learning framework can save computational inference time as only single network needs to be evaluated for three different tasks. The experimental results show that our proposed algorithm greatly improves the performance of the surface defect identification tasks compared to other state-of-the-art deep learning algorithms.

VI. Conclusion

In this work, we described an attention guided multi-task learning scheme which combines classification, segmentation and defection for automated surface defect detection. Specifically, we proposed an extended FPN architecture with Resnet-50 incorporated as the encoder section of the model. The hybrid loss function is introduced to enhance the performance of the model. An overall accuracy of 97.1%, Dice score of 0.926 and mAP of 0.762 on classification, segmentation and detection tasks of TekErreka dataset were achieved with Defect-Aux-Net.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[3] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.

[4] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.

[5] M. Tan and Q. Le, "{E}fficient{N}et: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, vol. 97, pp. 6105–6114.

[6] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[7] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020, doi: 10.1109/TPAMI.2018.2844175.

[8] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[10] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," 2015, pp. 234–241.

[11] S. Seferbekov, V. Iglovikov, A. Buslaev, and A. Shvets, "Feature Pyramid Network for Multi-class Land Segmentation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 272–2723, doi: 10.1109/CVPRW.2018.00051.

[12] X. Ni, Z. Ma, J. Liu, B. Shi, and H. Liu, "Attention Network for Rail Surface Defect Detection via Consistency of Intersection-over-Union(IoU)-Guided Center-Point Estimation," *IEEE Trans. Ind. Informatics*, vol. 18, no. 3, pp. 1694–1705, Mar. 2022, doi: 10.1109/TII.2021.3085848.

[13] D. Zhang, K. Song, Q. Wang, Y. He, X. Wen, and Y. Yan, "Two Deep Learning Networks for Rail Surface Defect Inspection of Limited Samples With Line-Level Label," *IEEE Trans. Ind. Informatics*, vol. 17, no. 10, pp. 6731–6741, Oct. 2021, doi: 10.1109/TII.2020.3045196.

[14] L. Wen, Y. Wang, and X. Li, "A New Cycle-consistent Adversarial Networks with Attention Mechanism for Surface Defect Classification with Small Samples," *IEEE Trans. Ind. Informatics*, pp. 1–1, 2022, doi: 10.1109/TII.2022.3168432.

[15] "Kaggle. Severstal: Steel Defect Detection. Can You Detect and Classify Defects in Steel?," 2019. .

[16] M. S. Kim, T. Park, and P. Park, "Classification of Steel Surface Defect Using Convolutional Neural Network with Few Images," in *2019 12th Asian Control Conference (ASCC)*, 2019, pp. 1398–1401.

[17] H. Lin, B. Li, X. Wang, Y. Shu, and S. Niu, "Automated defect inspection of LED chip using deep convolutional neural network," *J. Intell. Manuf.*, vol. 30, no. 6, pp. 2525–2534, Aug. 2019, doi: 10.1007/s10845-018-1415-x.

[18] X. Tao, D. Zhang, W. Ma, X. Liu, and De Xu, "Automatic metallic surface defect detection and recognition with convolutional neural networks," *Appl. Sci.*, vol. 8, no. 9, pp. 1–15, 2018, doi: 10.3390/app8091575.

[19] J. Ren and X. Huang, "Defect Detection Using Combined Deep Autoencoder and Classifier for Small Sample Size," in *2020 IEEE 6th International Conference on Control Science and Systems Engineering (ICCSSE)*, 2020, pp. 32–35, doi: 10.1109/ICCSSE50399.2020.9171953.

[20] J. Lian *et al.*, "Deep-Learning-Based Small Surface Defect Detection via an Exaggerated Local Variation-Based Generative Adversarial Network," *IEEE Trans. Ind. Informatics*, vol. 16, no. 2, pp. 1343–1351, Feb. 2020, doi: 10.1109/TII.2019.2945403.

[21] D. Zheng *et al.*, "A Defect Detection Method for Rail Surface and Fasteners Based on Deep Convolutional Neural Network," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–15, Jul. 2021, doi: 10.1155/2021/2565500.

[22] P. Xu, Z. Guo, L. Liang, and X. Xu, "MSF-Net: Multi-Scale Feature Learning Network for Classification of Surface Defects of Multifarious Sizes," *Sensors*, vol. 21, no. 15, p. 5125, Jul. 2021, doi: 10.3390/s21155125.

[23] D. Amin and S. Akhter, "Deep Learning-Based Defect Detection System in Steel Sheet Surfaces," in *2020 IEEE Region 10 Symposium (TENSYMP)*, 2020, pp. 444–448, doi: 10.1109/TENSYMP50017.2020.9230863.

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141, doi: 10.1109/CVPR.2018.00745.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.

[26] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2999–3007, doi: 10.1109/ICCV.2017.324.

[27] L. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 -- learning rate, batch size, momentum, and weight decay," 2018.

[28] A. Chaurasia and E. Culurciello, "LinkNet: Exploiting encoder representations for efficient semantic segmentation," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017, pp. 1–4, doi: 10.1109/VCIP.2017.8305148.

[29] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6230–6239, doi: 10.1109/CVPR.2017.660.

[30] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving Into High Quality Object Detection," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6154–6162, doi: 10.1109/CVPR.2018.00644.