Feature Aggregation Decoder for Segmenting Laparoscopic Scenes

Abdolrahim Kadkhodamohammadi[1], Imanol Luengo[1], Santiago Barbarisi[1], Hinde Taleb[1], Evangello Flouty[1], and Danail Stoyanov[1,2]

[1] Digital Surgery Ltd, 230 City Road, EC1V 2QY, London, UK
[2] Wellcome/EPSRC Centre for Interventional and Surgical Sciences, University College London, UK

Abstract. Laparoscopic scene segmentation is one of the key building blocks required for developing advanced computer assisted interventions and robotic automation. Scene segmentation approaches often rely on encoder-decoder architectures that encode a representation of the input to be decoded to semantic pixel labels. In this paper, we propose to use the deep *Xception* model for the encoder and a simple yet effective decoder that relies on a feature aggregation module. Our feature aggregation module constructs a mapping function that reuses and transfers encoder features and combines information across all feature scales to build a richer representation that keeps both high-level context and low-level boundary information. We argue that this aggregation module enables us to simplify the decoder and reduce the number of parameters in the decoder. We have evaluated our approach on two datasets and our experimental results show that our model outperforms state-of-the-art models on the same experimental setup and significantly improves the previous results, 98.44% vs 89.00%, on the EndoVis'15 dataset.

Keywords: semantic segmentation, minimally invasive surgery, surgical vision.

1 Introduction

Laparoscopic techniques have become a paradigm in modern interventions due to the numerous benefits over laparotomy such as shorter hospital stay, less scars, reduced postsurgical pain and faster recovery. Visualising the anatomy in high definition with bright illumination through the laparoscope also provides a magnified, detailed view of the surgical site that can be seen in 3D. However, minimally invasive surgery comes at the cost of restricting surgeon's range of motion and imposing altered hand-eye coordination [16]. As a result, significant efforts in computer assisted interventions (CAI) have been directed at tools to enhance surgeons' capabilities through robotics, image guidance and surgical data science [5, 10, 13, 14]. Laparoscopic scene segmentation is an essential building block in vision based CAI and is required to enable applications needing full surgical scene understanding [6, 2].

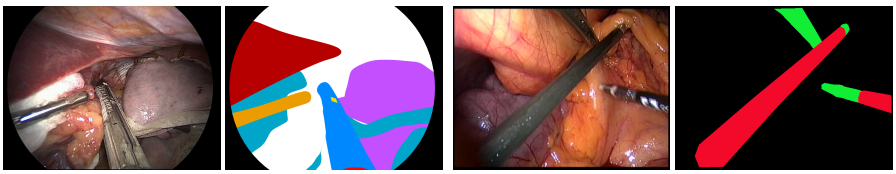

Fig. 1. Sample color and label images from LapSleeve and EndoVis'15 datasets, respectively.

Scene segmentation is a fundamental vision problem that is now tackled using deep Convolutional Neural Networks (CNNs). Features driving segmentation learned using deep CNN outperform handcrafted features like SIFT and HOG [8]. Fully Convolutional Networks (FCNs) can construct segmentation models that are learned in an end-to-end manner [12] using AlexNet [11] as the feature encoder and relying on transposed convolutions as the decoder to predict pixel-level labels. The FCN model can be extended by improving either the encoder or the decoder to achieve better performance [3, 7]. *U-Net* is one of the popular architectures adopting FCN for segmenting biomedical images [15]. The U-net encoder consists of a sequence of convolutional blocks that map and downsample the input by a factor of two and the decoder applies a sequence of similar blocks, but upsamples the output at the end of each block. *ToolNet* [7] follows a similar architecture, but simplifies the decoder to reduce the computation burden. The decoder concatenates the output of each encoder blocks and computes a segmentation loss on the output of each block to provide stage-wise supervision. While powerful, these architectures are relatively shallow and have a limited feature receptive field, which limits performance in complex surgical scenes.

In this paper, we introduce a novel decoder architecture that reuses the rich representations extracted by the *Xception* model [4]. This builds deep, rich representations while it reduces the number of parameters by using depthwise separable convolutions as shown by DeepLabv3+ [3], the top performer on Pascal segmentation challenge at the moment [1]. Our decoder relies on a feature aggregation module to incorporate information across all feature channels and construct a mapping function that selects and combines the most informative channels. This aggregation module allows reuse of the multi-scale features extracted at different Xception modules and construction of a representation that preserves semantic information along with detailed object boundaries. Previous works [3, 7, 12] have also explored the idea of reusing multi-scale features computed by the encoder but only with a decoder that reuses features in-between a series of convolutions and upsampling blocks. This introduces more parameters to the decoder and hence requires more training data. Instead, our feature aggregation decoder constructs a channel-wise mixing function and removes the need for multi-layer convolutions. We evaluate our approach on two datasets: *EndoVis'15* and Laparoscopic Sleeve gastrectomy, hereafter called *LapSleeve*. Fig. 1 shows sample images. Our experimental results show that the proposed decoder outperforms the more complex segmentation network of [3] on the same

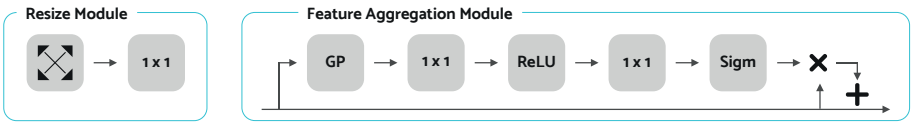

Fig. 2. The core modules of our decoder: left, a block to resize the output of Xception modules to the same size; right, feature aggregation module to learn a mapping function for transferring and combining feature channels.

experimental setup. Our model also significantly advances the state of the art results on EndoVis'15 dataset.

2 Method

Most recent scene segmentation approaches are based on FCN [3, 12]. These approaches are following the encoder-decoder design where sequences of convolutional blocks are used as both encoder and decoder. We argue that deep CNN encoders can encode both low-level and high-level information and a decoder can reuse this information without the need for deep multi-stage decoders. We therefore propose to use a deep CNN encoder and propose a simple feature aggregation encoder to perform scene segmentation, which are explained next.

2.1 Xception Encoder

The Xception network has been originally proposed for image classification and has achieved promising results on ImageNet [4]. This network benefits from depthwise separable convolutions to reduce the number parameters. Chollet in [4] shows that separable convolutions also allow using the model parameters more efficiently. The Xception architecture consists of entry, middle and exit flows, which are built by using sequences of Xception modules with different numbers of output channels, stride sizes and residual connection types. In this paper, we use the modified aligned Xception model of [3], which was adapted for image segmentation. The modifications are: (1) doubling the number modules in the middle flow; (2) replacing max pooling operations with separable convolution with stride; (3) adding batch normalisation and ReLU activation after each 3×3 convolution; (4) extracting multi-resolution feature maps using *atrous convolution*. From the modified Xception module, we do not use atrous convolution. We instead build a multi-scale feature map by reusing the features computed by Xception modules at different scales. More specifically, we reuse the output of all Xception modules in the entry flow and the last module in the middle as well as exit flows. The entry flow modules have narrow receptive fields and are therefore more likely to capture low-level features such as texture and boundary information [15]. Meanwhile modules close to the output of the network benefit from larger receptive fields, hence wider context, that can theoretically

enable constructing high-level representations for discriminating semantic categories [3]. We use our feature aggregation module to predict image pixel labels by assembling this low and high-level information.

2.2 Feature Aggregation Decoder

Our decoder utilises two modules to map representations into image pixel labels, shown in Fig. 2. We use the resize module at the output of the selected Xception modules for first resizing all the feature channels to be 1/16 of the input size and for fixing the number of output channels to 256. Bilinear interpolation is used to scale feature channels. The second module is the feature aggregation module. This module is designed to first capture global information and second construct a mapping function across all scales.

We aggregate information per channel by using global average pooling as a way of summarising global image information captured by each feature channel. We use these concise channel representations to learn a function for mapping information across channels. A similar idea has also been explored in [9] to model interdependency between channels inside a module. However, we argue that this operation can be used to learn dependency among features coming from different modules and recalibrate them to build a better representation. In our case, the benefit is not only aggregating information across scales, but also reducing the number of parameters and the computation burden at the decoder by effectively reusing extracted features. More formally, we can define the output of global average pooling as X and write the aggregation function as:

$$f(X) = \sigma(\rho(X * W_1 + \beta_1) * W_2 + \beta_2), \tag{1}$$

where σ is the standard logistic sigmoid function, ρ is the ReLU function, W_i and β_i are representing weight and bias vectors, respectively. This allows us to learn a nonlinear function, which incorporates channel-wise dependencies and relationships. This function can therefore put more emphases on some channels and learn a mapping function to calibrate feature channels. As function parameters are learned by optimising a segmentation loss, it learns to assemble the multi-scale features extracted at different parts of the encoder, which enables combining context and boundaries information. Finally, we apply a 1×1 convolution layer to the output of the feature aggregation module to refine and reduce the number of feature channels to 256. Our experiments show that this extra layer makes the training easier.

3 Experimental Results and Discussions

We implemented our approach using TensorFlow and perform all experiments on a Linux machine equipped with two NVIDIA GTX 1080 Ti GPUs. We optimise our networks using stochastic gradient decent. We use *poly* policy as learning rate scheduler [3] with the start learning rate of 0.0005 and finetune batch normalisation parameters. For the Xception backbone, we initialise the weight from

	[2]	DeepLabv3+ [3]			Feature Aggregation Decoder		
Metric	DSC	DSC	mean DSC	mean IOU	DSC	mean DSC	mean IOU
OP1	-	98.41	95.15	91.02	98.83	95.85	92.22
OP2	-	98.36	94.89	90.58	98.01	95.03	90.78
OP3	-	98.42	95.22	91.15	98.76	96.08	92.63
OP4	-	98.18	94.41	90.01	98.31	95.2	91.09
OP5-OP6	-	98.0	94.66	90.17	98.3	94.73	90.32
Average	89.00	98.27	94.87	90.59	98.44	95.38	91.41

Table 1. EndoVis'15 results. The evaluation results are presented as per the splits provided with the dataset.

a model trained on PASCAL VOC 2012 segmentation benchmark. Our resize module always scales the images to be 1/16 of the original image size.

For evaluation of our approach, we rely on two datasets: EndoVis'15 segmentation challenge and a laparoscopic sleeve gastrectomy dataset (LapSleeve). We use the EndoVis'15 rigid instrument dataset [2]. This dataset is generated from six laparoscopic colorectal surgeries. From each surgery, 50 frames are annotated. The train set includes the first 40 frames from OP1 to OP4 and the rest of the frames constructs the test set. A sample frame is shown in Fig. 1.

The LapSleeve dataset is generated from recordings of five laparoscopic sleeve gastrectomy procedures. We have randomly selected 600 to 900 frames from each video during the stomach dissection phase. In total, we have chosen 3600 frames. All these frames are annotated to provide full pixel-level segmentation masks. The dataset contains 14 class labels, namely stapler tip, stapler handle, stapler trigger, atraumatic grasper handle, atraumatic grasper tip, liver retractor, ligasure tip, ligasure handle, marylands tip, marylands handle, bandage, liver, stomach and background. We used all 750 frames from one of the videos as the test set and the rest as the training set.

We assessed the performance of our model by computing pixel intersection over union averaged across all classes (mean IOU). In case of EndoVis'15, we also compute the Dice Similarity Coefficient (DSC) as in [2], which is computed among prediction and ground-truth. As this is biased towards classes with high number of instances, we report average DSC across all classes (mean DSC).

EndoVis'15. We evaluate our model on the EndoVis'15 dataset following the experimental setup suggested in [2]. In other words, we follow a leave-one-surgery-out fashion, where frames from the test surgery are not used during training. We thus train five different models to evaluate on the different subsets provided in the test set. Table 1 presents the evaluation results in comparison to results of two other methods. Bodenstedt et al. [2] summarised the performance of the approaches participated in the EndoVis'15 challenge on instrument segmentation and tracking challenge. They obtained the best results by merging prediction results from several approaches using the STAPLE algorithm. In [2], the DSC metric is used to evaluate the performance of models in discriminating

	DeepLabv3+	FAD	FAD[-1CNN]	FAD[+1CNN]	FAD[-Add]	FAD[1/8]
mean IOU	42.76	47.81	45.74	46.88	46.16	41.54

Table 2. LapSleeve results. The mean IOU metric is used to compare the performance of our Feature Aggregation Decoder (FAD) with DeepLabv3+ on the same experimental. Different variants of our FAD are also evaluated. See the text for explanation.

.

tools vs background[3]. As the DSC is however sensitive to the number of instances per class and the dataset is extremely unbalanced, where $\sim 70\% - 90\%$ is the background class, we report mean DSC and also mean IOU that tends to penalise more wrong detections. In addition, we have reported the results of finetuned DeepLabv3+ initialised from a model trained on PASCAL VOC 2012. Our feature aggregation decoder preforms similarly to the DeepLabv3+, but always better, on the same experimental setup. This indicates that our decoder is capable of effectively aggregating information across different scales. Furthermore, our model achieves the DSC of 98.44%, which significantly outperforms the best model in [2].

Laparoscopic Sleeve. We use LapSleeve to train and evaluate our feature aggregation decoder and DeepLabv3+. All weights are initialised from models trained on PASCAL VOC 2012. The evaluation results on the LapSleeve dataset are presented in Table 2. Because of the higher complexity of LapSleeve that includes more classes and body organ segmentation classes, the performance of both models has dropped on this dataset compared to EndoVis'15 results. However, our model improves the performance by 5% over DeepLabv3+ on the same experimental setup. While DeepLabv3+ performs slightly better in segmenting body organs (80.01 vs 79.83), we found that our model is better in discriminating tool tips and tool handles. We should note that a handle and a tip of tool are in the same semantic group and only low-level edge information can help to distinguish these classes. Even though, given *enough* training data one can expect to retrieve this information from the presentation built at the end of encoder, this information is often better captured at shallower layers of the encoder. Our higher precision in discriminating tool tips from handles underlines the benefits of our aggregation decoder in reusing multi-scale features across the encoder as opposed to DeepLabv3+, which tries to obtain all this information at the end of the encoder.

Fig. 3 shows two sample frames along with corresponding labels and predictions. Our model is better in distinguishing grasper shaft from tip. The sample frame in the first row shows an example, where our model has successfully used low-level information to detect the stapler trigger. We have also used this dataset to evaluate different parameters of our model presented in Table 2. The perfor-

[3] Even though this dataset has been annotated for shaft, manipulator and background classes, the author of [2] confirms that shaft and manipulator are merged. We also merge these classes during our experiments

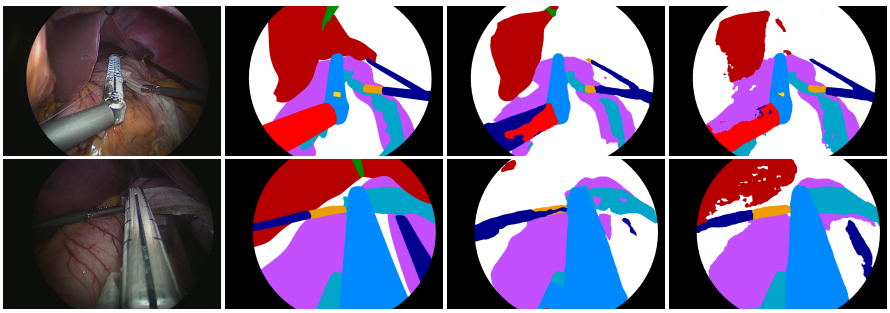

Fig. 3. Qualitative results: input image, label, DeepLabv3+ and our model. The predicted pixel-wise semantic labels are color coded.

mance of our model degrades dramatically when the resize module scales feature channels to 1/8 of the original image (FAD[1/8]). We believe that it is due to noise introduced by upsampling deep feature representations at the middle and the exit flows of Xception. Excluding the residual connection (FAD[-ADD]) also decreases the performance, which agrees with the findings in [9]. We remove (FAD[-1CNN]) and add (FAD[+1CNN]) a convolution layer after the feature aggregation module. The performance drops in both cases. Removing the convolution layer degrades the performance more, indicating that this layer is needed for reducing the number of channels in the representation built by the aggregation module and for converging to a better model.

4 Conclusions

In this paper, we proposed a simple yet effective decoder to perform laparoscopic scene segmentation. We use the modified aligned Xception model as our encoder. Our decoder relies on an aggregation module to reuse and calibrate representations extracted by the encoder at different scales. This aggregation module allows us to select the most informative feature channels and reuse them effectively for predicting pixel-level semantic labels. Our experiments on two different datasets highlights the effectiveness of our decoder. Our model significantly advances the state-of-the-art results on EndoVis'15 and achieves 98.44% DSC. We believe that the forward nature of our decoder enables systematic study of features at different modules that would boost the explainablility of our model and it would be interesting to look at this aspect in future work.

References

1. Pascal VOC 2012: segmentation leaderboard. `http://host.robots.ox.ac.uk/leaderboard/displaylb.php?challengeid=11&compid=6`, last access: March 2019

2. Bodenstedt, S., Allan, M., Agustinos, A., Du, X., Garcia-Peraza-Herrera, L., Kenngott, H., Kurmann, T., Müller-Stich, B., Ourselin, S., Pakhomov, D., et al.: Comparative evaluation of instrument segmentation and tracking methods in minimally invasive surgery. arXiv preprint arXiv:1805.02475 (2018)
3. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Computer Vision – ECCV 2018. pp. 833–851. Springer (2018)
4. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1800–1807. IEEE (July 2017)
5. da Costa Rocha, C., Padoy, N., Rosa, B.: Self-supervised surgical tool segmentation using kinematic information. In: International Conference on Robotics and Automation (ICRA). IEEE (2019)
6. D'Ettorre, C., Dwyer, G., Du, X., Chadebecq, F., Vasconcelos, F., De Momi, E., Stoyanov, D.: Automated pick-up of suturing needles for robotic surgical assistance. In: International Conference on Robotics and Automation (ICRA). pp. 1370–1377. IEEE (2018)
7. Garca-Peraza-Herrera, L.C., Li, W., Fidon, L., Gruijthuijsen, C., Devreker, A., Attilakos, G., Deprest, J., Poorten, E.V., Stoyanov, D., Vercauteren, T., Ourselin, S.: ToolNet: Holistically-nested real-time segmentation of robotic surgical tools. In: International Conference on Intelligent Robots and Systems (IROS). pp. 5717–5722. IEEE (Sep 2017)
8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Conference on computer vision and pattern recognition (CVPR). pp. 580–587. IEEE (2014)
9. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (June 2018)
10. Jin, A., Yeung, S., Jopling, J., Krause, J., Azagury, D., Milstein, A., Fei-Fei, L.: Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks. In: Winter Conference on Applications of Computer Vision (WACV). pp. 691–699 (March 2018)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
12. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Conference on computer vision and pattern recognition (CVPR). pp. 3431–3440. IEEE (2015)
13. Maier-Hein, L., Eisenmann, M., Reinke, A., Onogur, S., Stankovic, M., Scholz, P., Arbel, T., Bogunovic, H., Bradley, A.P., Carass, A., et al.: Why rankings of biomedical image analysis competitions should be interpreted with care. Nature communications 9(1), 5217 (2018)
14. Münzer, B., Schoeffmann, K., Böszörmenyi, L.: Content-based processing and analysis of endoscopic images and videos: A survey. Multimedia Tools and Applications 77(1), 1323–1362 (Jan 2018)
15. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI). pp. 234–241. Springer (2015)
16. Wilson, M., Coleman, M., McGrath, J.: Developing basic hand-eye coordination skills for laparoscopic surgery using gaze training. BJU Int 105(10), 1356–1358 (2010)