DEEP ENCODER-DECODER NETWORKS FOR ARTEFACTS SEGMENTATION IN ENDOSCOPY IMAGES

Yun Bo Guo, Qingshuo Zheng, Bogdan J. Matuszewski

Computer Vision and Machine Learning (CVML) Group
School of Engineering
University of Central Lancashire
{YBGuo1,QZheng5,BMatuszewski1}@uclan.ac.uk

ABSTRACT

Automated analysis of endoscopic images is becoming increasingly significant for an early detection of numerous cancers and minimally invasive surgical procedures. The paper briefly describes the methodology adopted for the 2020 Endoscopy Artefact Detection and Segmentation (EAD2020) challenge[1]. A number of novel variants of the DeepLab V3+ encoder-decoder architecture have been investigated, implemented and tested for the segmentation sub-challenge. Modifications were introduced to improve: selection of image futures, segmentation of small objects, and use of the encoder output information. The proposed methods achieved competitive segmentation score results on both release-I and release-II test datasets. For the detection sub-challenge three off-the-shelf deep detection networks have been optimised and evaluated on the EAD data.

1. INTRODUCTION

Automated analysis of endoscopic images has obvious practical clinical importance. For example, colorectal cancer is one of the leading causes of death worldwide, e.g. in the United States, it is the third largest cause of cancer deaths; whereas in Europe, it is the second largest with 243,000 deaths in 2018 [1]. Colonoscopy is the gold standard for colon screening, with colon cancer survival rate strongly depending on the early detection, i.e. a colonoscopy procedure.

Automation of the analysis of endoscopic images poses significant technical difficulties. As evident from the EAD challenge, the segmentation task is a very demanding problem, with multiple difficult to define semantic categories, possibly represented within the same/similar image locations and structures of significantly different sizes. Additionally, some of these categories (e.g. "bubbles") are difficult to discriminate with respect to appearance and spatial distribution.

Segmentation is one of the key enabling technologies in medical image analysis with a great variety of methods proposed [2, 3, 4]. More recently, methods based on deep learning showed significant improvement in the quality of the segmentation also in analysis of colonoscopy images [5, 6].

The key architectures used as the baseline for the segmentation methods, developed for the EAD challenge, are Dilated ResFCN [5], previously proposed by the authors, and the well-known DeepLab V3+ [7]. The summary of the changes made to these baseline architectures is briefly explained in section 3. For completeness the detection sub-task has been also investigated with the YOLO V3 [8], Faster R-CNN [9], and Cascade R-CNN [10] methods used as the baseline, with their design parameters optimised.

2. DATASETS

Only the data, which have been made available as part of the EAD2020 challenge [11, 12] have been directly used for the reported methods' development. Some of the networks and/or sub-networks used in the designed architectures, have been acquired from the GitHub repository[2]. These are normally pre-trained on open generic image datasets, such as ImageNet or COCO. Apart from such cases, no data other than EAD2020, have been used for training, validation or testing of the developed architectures.

The original EAD2020 training images are augmented by rotation, colour jitter and elastic deformations. For the segmentation task, all the images have been scaled to 513×513 pixels in size, with two training data subsets created. The smaller training subset consists of 11,376 images, augmented from the phase-I training dataset. The networks trained on this smaller subset have been validated on the phase-II training dataset and online on the test datasets. This small training subset was predominantly used to quickly verify specific design choices made during the methods' development. The larger training subset consists of 38,195 images augmented

[1] It refers to the results submitted by the CVML team.

[2] //github.com{/ultrlytics/yolov3,/open-mmlab/mmdetection,/hujie-frank/SENet}.

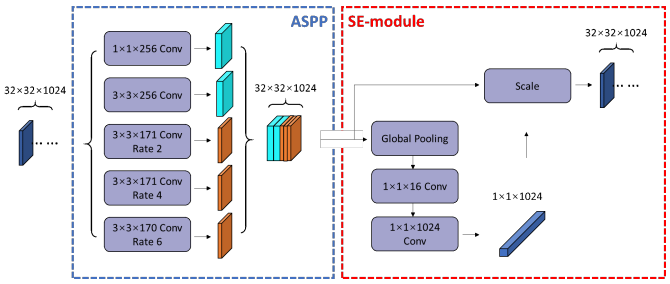

Fig. 1. Flowchart of the Network 3 encoder architecture.

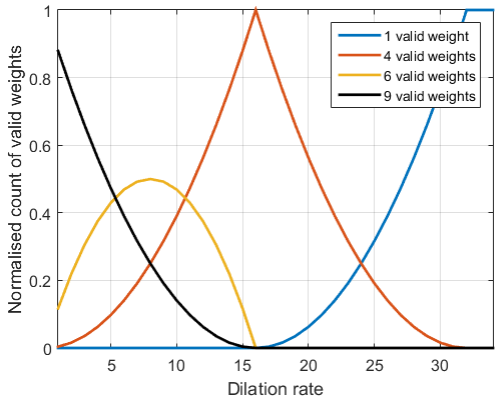

Fig. 2. The number of valid weights in the dilation kernels shown in Fig.1.

from the phase-I and phase-II training datasets. That bigger training set was used to train architectures which have been thought to provide competitive results when trained on the smaller dataset. The networks trained on the larger training subset were only evaluated online on the EAD2020 test datasets.

For the detection sub-problem, the images have been scaled to 667×400 pixels in size. As for the segmentation, two augmented training subsets were created. The smaller subset with images augmented from phase-I training dataset consists of 8800 images, whereas the large subset has 30,372 images augmented from the phase-I and phase-II training datasets.

3. METHODS

DeepLab V3+ [7] is an end-to-end trained semantic segmentation network, where lower down-sampling rate and dilated convolutions are used to maintain the size of feature maps, and an atrous spatial pyramid pooling (ASPP) module generates the final features based on multiple receptive fields. Finally, these features are up-sampled, and the classifier assigns the unique class label to each pixel.

A number of novel network architectures (here collectively named as DeepEAD), based on the DeepLab V3+, have been proposed and validated for the EAD2020 segmentation challenge. In order to segment the overlapping objects, the original multi-class classifier is replaced with 5 binary classifiers. Further changes lead to three network architectures:

- Network 1: The original DeepLab V3+ main subnetwork is replaced by the SE-ResNeXt-50 [13]. It is expected to provide better image features, as it outperforms both Xception and ResNet architectures (originally used by different implementations of the DeepLab V3+) on the image classification task.
- Network 2: Based on Network 1, with the global pooling removed from the ASPP and replaced with 3×3 convolutions. The corresponding receptive fields are expected to improve segmentation of the small objects. Furthermore, the number of the convolution kernels at each resolution is selected to emphasise small objects.
- Network 3: Shown in Fig.1, is based on Network 2, with the squeeze and excitation module added behind the ASPP module. This is to introduce attention gating at the output of the original encoder to better utilise information available in the computed feature maps.

Following on the methodology proposed in [14], Fig.2 shows the number of active kernel weights of the dilated subnetworks. It can be seen that with a too high dilation rate the 3×3 kernel is effectively reduced to a 1×1 kernel. However, a too small dilation rate results in a small receptive field, having a negative effect on the network performance. The selected dilation rates of 2, 4, and 6 provide an effective compromise with kernels having between 4 and 9 valid weights.

Since the proposed networks don't have built-in rotation invariance, to improve the segmentation accuracy the image rotation augmentation during test time has been investigated. For this purpose, rotated versions of the test image are presented to the network and the corresponding outputs are averaged to better utilise generalisation properties of the network. The adopted test time augmentation process is explained in Fig.3. The corresponding results, shown in section 4, demonstrate that the test time augmentation does indeed have a significant impact on the segmentation performance.

4. RESULTS

This section reports on a sample of results obtained for the segmentation and detection methods described above. Table 1 shows a representative sample of the results obtained for the segmentation task on both validation and release-I test datasets. The results obtained on the validation data (phase-II training data) are reported in the second column, with all the networks trained only on the augmented images from the phase-I training dataset. The results on the release-I test dataset are reported in the third column. The symbol "*"

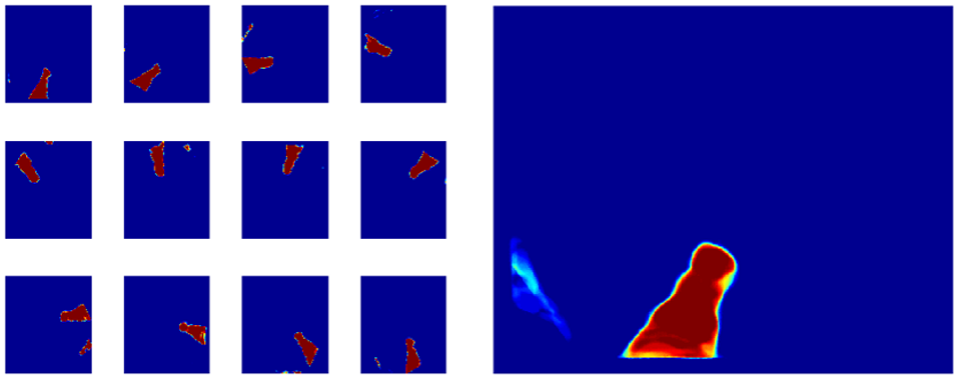

Fig. 3. Test time augmentation, with images on the left showing network outputs for the original image and its rotated, in 30 degree intervals, versions. Image on the right shows the result after augmentation with the individual results superimposed in the original image reference frame.

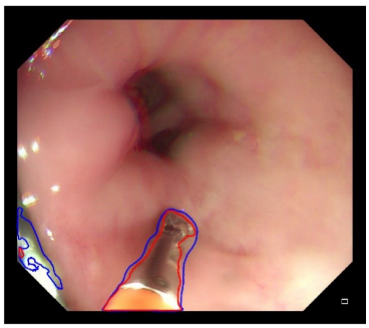

Fig. 4. The result from Network 3 with (in red) and without (in blue) test time augmentation.

indicates that the result has been obtained for the network trained on the large training dataset (i.e. images augmented from the phase-I and phase-II training sets), otherwise results have been obtained for the network trained on the small training dataset (i.e. images augmented from the phase-I training dataset only - see section 2 for more details). It could be concluded that the gradual improvement of the results on the validation data is replicated on the test data. As expected the use of the large training set also improves performance. This can be seen from the results reported for Network 2, with the segmentation score of 0.50 for the network trained on a smaller training set, and score of 0.59 for the exactly the same network but trained on the large dataset. It seems that the segmentation score of 0.5934 (for the Network 2 trained on the larger training set) was a competitive result on the release-I test dataset.

The best results obtained on the release-II test dataset, with all the networks trained on the larger training dataset, are reported in Table 2. As evident from the table, Network 3 provides the best segmentation results with the test time augmentation improving the segmentation score by 0.0434, i.e. about 8%. The effects of the test time augmentation are

Method	**sscore** (validation)	**sscore** (test data)
DeepLab v3+	0.45	0.40
Network 1	0.50	0.48
Network 2	0.52	0.50 / 0.59*
Network 3	0.54	0.52

Table 1. The segmentation score results for various segmentation networks, obtained on the validation (second column) and release-I test (third column) data.

Method	**sscore**(release-II test)
Network 2	0.5406
Network 3	0.5488
Network 3 (+ test time augmentation)	0.5922

Table 2. Segmentation scores on the release-II test data.

shown in Fig.4 demonstrating impact of the augmentation on segmentation of the ”instrument” class.

Various post-processing operations have been also tested, including hole filling and removal of objects from the image black boundary. These, though, had a relatively small, and difficult to predict, effect on the segmentation score. The segmentation score result for the final submission was reported as 0.5916, which was slightly lower than the best result of 0.5922 (see Table 2).

Table 3 shows results obtained for different detection networks tested on the release-II test data. It could be observed that R-CNN networks outperform the Yolo network, with the best detection score achieved by the Faster R-CNN. This is different from the results obtained on the release-I test set (not reported here) where the Yolo network achieved better result. This though could be possibly explained by optimisation of the networks design parameters during the second phase of testing.

Method	**Detection scores** (release-II test data)
Yolo V3	0.1992
Faster R-CNN	0.2335
Cascade R-CNN	0.2162

Table 3. Detection scores on the release-II test data.

5. DISCUSSION & CONCLUSION

The paper describes novel segmentation networks, highlighting the key characteristics of the proposed deep architectures. The proposed methods achieved segmentation scores of 0.5934 on the release-I test data and 0.5922 on the release-II test data, which seem to be competitive. The overall detection performance also seems comparatively reasonable with best detection score of 0.2335 on the release-II test data. However, the statistical significance of these results would need to be investigated. Further improvements could be possible, e.g. with the image aspect ratio augmentation to reflect the input format of the adopted networks, or use of the segmentation network as a pre-selection tool for detection of small objects (e.g. specularity artefacts).

6. REFERENCES

[1] J. Ferlay, M. Colombet, I. Soerjomataram, T. Dyba, G. Randi, M. Bettio, A. Gavin, O. Visser, and F. Bray. Cancer incidence and mortality patterns in europe: Estimates for 40 countries and 25 major cancers in 2018. *European Journal of Cancer*, 103:356 – 387, 2018.

[2] Aymeric Histace, Bogdan J. Matuszewski, and Yan Zhang. Segmentation of myocardial boundaries in tagged cardiac MRI using active contours: A gradient-based approach integrating texture analysis. *Int. J. Biomedical Imaging*, 2009:983794:1–983794:8, 2009.

[3] Yan Zhang, Bogdan J. Matuszewski, Aymeric Histace, Frédéric Precioso, Judith Kilgallon, and Christopher J. Moore. Boundary delineation in prostate imaging using active contour segmentation method with interactively defined object regions. volume 6367 of *Lecture Notes in Computer Science*, pages 131–142. Springer, 2010.

[4] Yan Zhang, Bogdan J. Matuszewski, Aymeric Histace, and Frédéric Precioso. Statistical model of shape moments with active contour evolution for shape detection and segmentation. *Journal of Mathematical Imaging and Vision*, 47(1-2):35–47, 2013.

[5] Yun Bo Guo and Bogdan J. Matuszewski. GIANA polyp segmentation with fully convolutional dilation neural networks. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2019, Volume 4: VISAPP, Prague, Czech Republic, February 25-27, 2019*, pages 632–641. SciTePress, 2019.

[6] Yun Bo Guo and Bogdan J. Matuszewski. Polyp segmentation with fully convolutional deep dilation neural network. In *Medical Image Understanding and Analysis - 23rd Conference, MIUA 2019, Liverpool, UK, July 24-26, 2019, Proceedings*, volume 1065 of *Communications in Computer and Information Science*, pages 377–388. Springer, 2019.

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[8] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018.

[9] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.

[10] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: high quality object detection and instance segmentation. *CoRR*, abs/1906.09756, 2019.

[11] Sharib Ali, Felix Zhou, Barbara Braden, Adam Bailey, Suhui Yang, Guanju Cheng, Pengyi Zhang, Xiaoqiong Li, Maxime Kayser, Roger D. Soberanis-Mukul, Shadi Albarqouni, Xiaokang Wang, Chunqing Wang, Seiryo Watanabe, Ilkay Oksuz, Qingtian Ning, Shufan Yang, Mohammad Azam Khan, Xiaohong W. Gao, Stefano Realdon, Maxim Loshchenov, Julia A. Schnabel, James E. East, Geroges Wagnieres, Victor B. Loschenov, Enrico Grisan, Christian Daul, Walter Blondel, and Jens Rittscher. An objective comparison of detection and segmentation algorithms for artefacts in clinical endoscopy. *Scientific Reports*, 10, 2020.

[12] Sharib Ali, Felix Zhou, Christian Daul, Barbara Braden, Adam Bailey, Stefano Realdon, James East, Georges Wagnieres, Victor Loschenov, Enrico Grisan, et al. Endoscopy artifact detection (ead 2019) challenge dataset. *arXiv preprint arXiv:1905.03209*, 2019.

[13] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *CoRR*, abs/1709.01507, 2017.

[14] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017.