A U-NET++ WITH PRE-TRAINED EFFICIENTNET BACKBONE FOR SEGMENTATION OF DISEASES AND ARTIFACTS IN ENDOSCOPY IMAGES AND VIDEOS

Le Duy Huỳnh, Nicolas Boutry

EPITA Research and Development Laboratory (LRDE), 14-16 rue Voltaire, F-94270 Le Kremlin-Bicêtre, France

ABSTRACT

Endoscopy is a widely used clinical procedure for the early detection of numerous diseases. However, the images produced are usually heavily corrupted with multiple artifacts that reduce the visualization of the underlying tissue. Moreover, the localization of actual diseased regions is also a complex problem. For that reason, EndoCV2020 challenges aim to make progress in the state-of-the-art in the detection and segmentation of artifacts and diseases in endoscopy images. In this work, we propose approaches based on U-Net and U-Net++ architecture to automate the segmentation task of EndoCV2020. We use the EfficientNet as our encoder to extract powerful features for our decoders. Data augmentation and pre-trained weights are employed to prevent overfilling and improve generalization. Test-time augmentation also helps in improving the results of our models. Our methods performs well in this challenge and achieves a score of 60.20% for the EAD2020 semantic segmentation task and 59.81% for the EDD2020's.

***Index Terms*—** Endoscopy · U-Net++ · EfficientNet · Test-time augmentation · Segmentation · Detection

1. INTRODUCTION

Endoscopy is a widely used clinical procedure for the early detection of cancers, therapeutic procedures, and minimally invasive surgery in hollow-organs. Computer-assisted methods would improve diagnosis and assist in surgical planning. These endoscopic applications face two main challenges: 1) Endoscopy video frames are usually heavily corrupted with multiple artifacts that reduce the visualization of the underlying tissue and affect post-analysis. Accurate detection of these artifacts allows corrections of frames and is, therefore, a core challenge in a wide range of endoscopic applications [1]. 2) Even with uncorrupted video frames, temporally consistently localizing and segmenting of disease ROIs is a challenge due to non-planar geometries, variation in imaging modalities, and deformations of organs. For these reasons, after the success of EAD2019 [2], the endoscopy computer vision challenges on segmentation and detection 2020 (EndoCV 2020) are held to make progress in the state-of-the-art further. It consists of two sub-challenges: Endoscopy Artefact Detection and Segmentation (EAD2020), and Endoscopy Disease Detection and Segmentation (EDD2020) [3]. Each challenge is further divided into the detection and the semantic segmentation tasks.

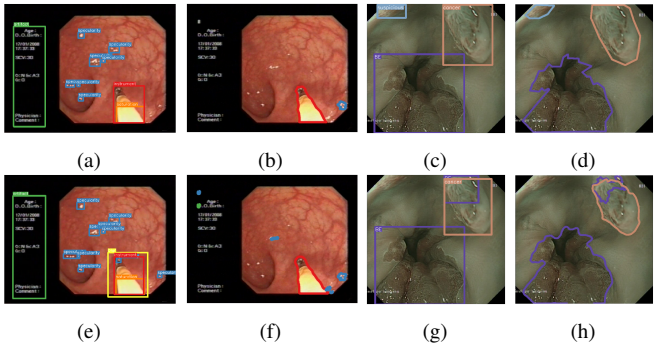


Fig. 1. Some output examples of our U-Net++ with EfficientNet B1 and YOLOv3. The top row is the ground truths (GT). a),b): EAD2020 detection and segmentation GT. c),d): EDD2020 detection and segmentation GT. The bottom row is the results for corresponding tasks on the top row.

In this paper, we introduce our works on these challenges. First, Sec. 2 presents our observations of the datasets. Then, we describe our methods in Sec. 3. We participate in all tasks of both challenges but focus mainly on the two semantic segmentation ones using fully convolutional networks such as U-Nets [4] and U-Net++ [5] with EfficientNet [6] backbones (Sec. 3.1). We also did some experiments with the detection tasks using our segmentation results or the YOLOv3 model [7] (Sec. 3.2). Finally, we will present our results on the testing data in Sec. 4 and conclude in Sec. 5.

2. DATASETS

2.1. About the dataset

The EndoCV2020 challenge consists of two datasets. The EAD2020 dataset is an extended version of last year's EAD2019 challenge [8] with annotations corrected, and a

new class added. The EAD2020 train data were released in four subsets. They consist of a total of 3005 images, among which 2531 are annotated with bounding boxes for eight classes (specularity, saturation, artifact, blur, contrast, bubbles, instrument, and blood), and 643 come with segmentation ground truth for five classes (instrument, specularity, artifact, bubbles, saturation, and blood). The EDD2020 train data contains 386 images with boxes and segmentation ground-truth for five disease classes (normal dysplastic Barrett's oesophagus (BE), suspicious area, high-grade dysplasia (HGD), adenocarcinoma (cancer), and polyp).

2.2. Data correction

The ground-truth for these challenges contain some issues. They can be classified into annotator disagreement and systematic error.

Some annotator disagreement was spotted in the EAD2020 detection dataset. For example, an artifact region is identified in one frame but is not in the next frame, or one connected region is marked with two bounding boxes in one frame but only one in the other. There is also misclassified segmentation mask. We consider this type of error as noises in the dataset since we do not have the resource to make adjustments.

We corrected all the systematic errors that we noticed. There are some small anomalies in bounding box ground truths, likely due to rounding when converting from Pascal-VOC to Yolo format. There is also a line at the bottom of some EAD2020 segmentation ground-truths that do not correspond to any artifact.

3. OUR APPROACHES

We focus mainly on the two segmentation tasks with models in the U-Net family. In the case of disease detection, we will take advantage of our segmentation results. For the artifacts detection tasks, we train a separate YOLOv3 [7] network due to the difference in the number of classes (eight classes for detect and five classes for segmentation), and due to disagreement between EAD2020 segmentation and detection ground-truths (e.g, Fig 1(a) and Fig 1(b)).

3.1. Segmentation of Diseases and Artifacts

3.1.1. Models

The state-of-the-art of semantic segmentation are methods based on encoder-decoder architecture such as U-Net, U-Net++. These encoder-decoder architectures use skip-connections to combine low-resolution, semantically-rich at deeper feature maps with shallower, fine-grained ones to recover fine detail of region of interest.

Instead of using the original U-Net encoder, we use EfficientNet [6], which claims to be balanced between network depth, width, and resolution. This architecture achieved better accuracy on ImageNet [9] with fewer parameters and requires fewer FLOPS than other networks such as ResNet [10] or DenseNet [11]. EfficientNet is available with different versions, starts from B0 at 5.3 million parameters to B7 at 66 million. We extract five feature maps at different scales from EfficientNet as the input of our decoders. An illustration of the EfficientNet B1 encoder and where intermediate feature maps are extracted is in Fig. 2(a).

We start our experimentations with the U-Net architecture and the EfficientNet B5. Subsequent tests show that a deeper encoder is not needed, so we transit to a larger decoder (U-Net++) with smaller encoder (EfficientNet B2 and latter B1). An illustration of our networks could be found in Fig. 2

We speed up the training process with pre-trained weights. Although it was trained on ImageNet, which is a database of natural images, the pre-trained weights do improve the training and local validation scores.

3.1.2. Data augmentation

Input images are resized (while keeping their aspect ratio) and zeros-padded to fit into 512x512 pixels. We randomly apply these augmentation techniques with 50% probability: Rotation with random angle, RGB value shift, horizontal or vertical flipping, random scaling, elastic deformation [12], cropping.

3.1.3. The loss function

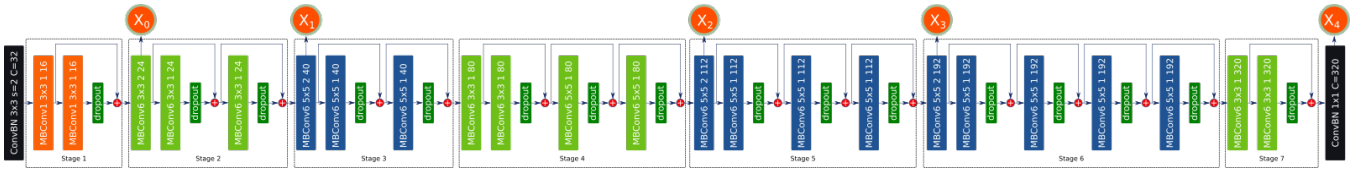
The semantic segmentation tasks are evaluated with four metrics: F_1 (i.e., Dice score), F_2, precision, and recall. The semantic segmentation score is the average of these four metrics [3]. Let us remind that $F_\beta = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$ weights recall β times as important as precision. Therefore the final score places more emphasis on recall. As a result, we will train our network using F_2-loss, which is defined similarly to Dice-loss: F_2-loss = 1-F_2. We also apply L2 regularization with a factor of 0.0001.

3.1.4. Training

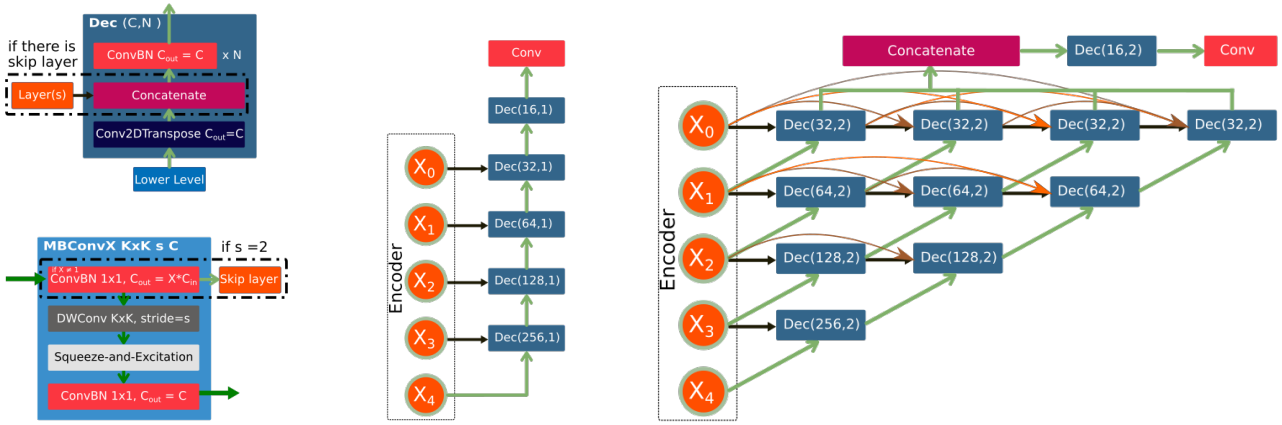
We train on 80% of the train set and use the other 20% for validation. We start by fixing the pre-trained encoder and train the decoder part for 80 epochs using Adam optimizer with a learning rate (LR) of 10^{-3}. From the 41st epochs, we train the whole network, starting at LR=10^{-3} and decrease with a factor of 0.5 if the validation score does not decrease after 40 epochs. We trained for a total of 1000 epochs. The training is early stopped if the training score could not be increased after 88 epochs. We select the weights that maximized the score on the validation set for evaluation of the test set.

3.1.5. Prediction and Post-processing

Test images are resized and zero-padded to 512x512. We keep the aspect ratio and do not enlarge small images. Since our prediction sometimes contains small holes which rarely appear in the EAD2020 train set, a small hole-filling operation is applied at the end of the pipeline. This hole-filling process did not make a significant improvement in the final score.



(a) EfficientNet B1 encoder and extracted intermediate feature maps (X_i)



(b) Building blocks (c) U-Net (d) U-Net++

Fig. 2. Illustration of our version of U-Net and U-Net++. 2(a): EfficientNet B1 encoder. 2(b) EfficientNet building block $MBConv$ and the building block ouf our decoder Dec. 2(c), 2(d): our version of U-Net and U-Net++

3.1.6. Test-time Augmentation

The segmentation results could be further improved with Test-time augmentation (TTA). This approach has been demonstrated in the literature (e.g., in semantic segmentation of brain tumor [13]). In short, we will make predictions on the test image and several of its transformed versions and then combine these results. We use five transformations: horizontal, vertical flipping, rotations of 90°, 180°, and 270°.

3.2. Detection of Diseases and Artifacts

3.2.1. Detection of Diseases (EDD2020)

For this task, we will take advantage of our segmentation results. The bounding boxes of connected components (CCs) larger than 0.5% of the image area are presented as our detection results. This approach is not optimal because it cannot handle slit-and-merge of detection bounding boxes, that is the cases where a single CC is marked with more than one box, or a single box marks several CCs.

3.2.2. Detection of Artifacts (EAD2020)

For reasons stated earlier, we have to train a separate detector for this task. We choose YOLOv3 because this model's effectiveness has been shown in this type of data [1]. It is also relatively faster than two-states detectors such as RetinaNet [14]. We have tried to address the class imbalance issue with focal loss [14]. However, it did not improve our local validation mAP, similar to the remark in [7].

Table 1. Number of images in test data for each task

	Detection	Segmentation	Out-of-sample detection
EAD2020	317	162	99
EDD2020	43	43	0

We train our standard YOLOv3 on 416x416 inputs. We start by training the last three detection layers for 20 epochs at LR=10^{-2}, then the upscaling part of the network for 40 epochs at LR=10^{-3}. Finally, we train the whole network at LR =10^{-5} and reduce the LR by a factor of 0.5 if validation loss does not decrease after five epochs.

4. EVALUATION

As presented in Sec. 3.1.3, the semantic segmentation are evaluated by the mean of F_1, F_2, precision, and recall. The detection tasks are evaluated by a combination of mean average precision (mAP), and intersection over union (IoU). However, how these metrics are weighted was not disclosed until after the challenge ends.

The evaluation was done online at https://endocv.grand-challenge.org/. It is divide into two phases. The first test-phase only evaluate 50% of the test data. The size of EAD2020 and EDD2020 test data is summarized in Tab. 1. The EAD2020 detection task includes an out-of-sample set, which contains images provided exclusively from the training or other test datasets.

The quantitative results of our models are presented in Tab. 2 and Tab. 3. Our segmentation models performed well

Table 2. Quantitative results of our segmentation models on EAD2020 and EDD2020 test data. (4) TTA using the first four transformations mentioned in Sec. 3.1.6, (5) TTA using all transformations in Sec. 3.1.6, (F) apply holes filling, (U-Net++^{512}): model with same architechture as in Fig. 2(d) but with double the number of decoder filters, i.e., $Dec(x, 2)$ becomes $Dec(2x, 2)$, (-): information is not available because that method was not submitted to that test phase.

Endoscopy Diseases and Artifacts Segmentation						
Model	**50% of Test Set**					**Full Test Set**
	F1	**F2**	**Precision**	**Recall**	**Score**	**Final Score**
EAD 2020						
Unet++B1^{5F}	-	-	-	-	-	**0.6020**
Unet++B1^{4F}	**0.5777**	**0.5629**	0.6584	**0.5661**	**0.5913**	0.5989
Unet++B1^{4}	0.5766	0.5624	0.6556	0.5659	0.5901	**-**
Unet++B1	0.5305	0.5246	0.5963	0.5391	0.5476	-
Unet++B2	0.5210	0.4921	**0.6586**	0.4872	0.5397	-
Unet B5	0.5126	0.5093	0.5767	0.5436	0.5355	-
EDD2020						
U-Net++^{512}B1^{5F}	-	-	-	-	-	**0.5981**
Unet++B1^{4}	0.4956	0.5676	0.4280	0.6562	0.5369	0.5436

Table 3. Quantitative results of our detection models on EAD2020 and EDD2020 test data.

Endoscopy Diseases and Artifacts Detection		
Method	**Score**	**Out-of-sample score**
EAD2020		
YOLOv3	0.1702 ± 0.0567	0.1130 ± 0.0752
EDD2020		
U-Net++^{512}B1^{5F}	0.1565 ± 0.0547	-
Unet++B1^{4}	0.1487 ± 0.0578	-

and consistently on both test subset. In Tab. 2, we can observe that the hole-filling operator adds a small boost while TTA improves the final score significantly. Our best approach is U-Net++ with B1 encoder, ran with TTA and post-processed with holes-filling. We could also see that our detection model needs improvements.

5. CONCLUSION AND PERSPECTIVE

In this work, we demonstrate the effectiveness of U-Net and U-Net++ with pre-trained EfficientNet backbone for the segmentation of disease and artifact in endoscopy images. Our experiments also show that TTA can provide better segmentation results compared to only predicting on original images.

For further works, we intend to improve the generalization of our segmentation and detection models by applying more data augmentation techniques and using synthetic data. We are also considering to improve further the decoder of our models with an attention mechanism.

6. REFERENCES

[1] Sharib Ali, Felix Zhou, Adam Bailey, Barbara Braden, James East, Xin Lu, and Jens Rittscher. A deep learning framework for quality assessment and restoration in video endoscopy. *arXiv preprint arXiv:1904.07073*, 2019.

[2] 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.

[3] Sharib Ali, Noha Ghatwary, Barbara Braden, Lamarque Dominique, Adam Bailey, Stefano Realdon, Cannizzaro Renato, Jens Rittscher, Christian Daul, and James East. Endoscopy disease detection challenge 2020. *CoRR*, abs/2003.03376, February 2020.

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241. Springer International Publishing, 2015.

[5] 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, 2018.

[6] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019.

[7] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement.

[8] 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.

[9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016.

[11] 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)*, pages 2261–2269, 2017.

[12] P.Y. Simard, D. Steinkraus, and J.C. Platt. Best practices for convolutional neural networks applied to visual document analysis. In *ICDAR, 2003.*, volume 1, pages 958–963. IEEE Comput. Soc.

[13] Guotai Wang, Wenqi Li, Sébastien Ourselin, and Tom Vercauteren. Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, pages 61–72, Cham, 2019. Springer International Publishing.

[14] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollr. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Oct 2017.