DivergentNets: Medical Image Segmentation by Network Ensemble

Vajira Thambawita[a,b], Steven A. Hicks[a,b], Pål Halvorsen[a,b] and Michael A. Riegler[a]

[a] *SimulaMet, Pilestredet 52, 0167 Oslo, Norway*
[b] *OsloMet, Pilestredet 46, 0167 Oslo, Norway*

Abstract

Detection of colon polyps has become a trending topic in the intersecting fields of machine learning and gastrointestinal endoscopy. The focus has mainly been on per-frame classification. More recently, polyp segmentation has gained attention in the medical community. Segmentation has the advantage of being more accurate than per-frame classification or object detection as it can show the affected area in greater detail. For our contribution to the EndoCV 2021 segmentation challenge, we propose two separate approaches. First, a segmentation model named *TriUNet* composed of three separate UNet models. Second, we combine TriUNet with an ensemble of well-known segmentation models, namely UNet++, FPN, DeepLabv3, and DeepLabv3+, into a model called *DivergentNets* to produce more generalizable medical image segmentation masks. In addition, we propose a modified Dice loss that calculates loss only for a single class when performing multi-class segmentation, forcing the model to focus on what is most important. Overall, the proposed methods achieved the best average scores for each respective round in the challenge, with TriUNet being the winning model in Round I and DivergentNets being the winning model in Round II of the segmentation generalization challenge at EndoCV 2021. The implementation of our approach is made publicly available on GitHub.

Keywords

Deep learning, medical image segmentation, colonoscopy, generalisation, computer-assisted diagnosis

1. Introduction

Automatic segmentation of medical images is a common use case in machine learning that has gained a lot of attention over the last few years. Popular applications include segmenting tumors in computed tomography (CT) scans [1, 2], finding abnormalities in magnetic resonance images (MRIs) [3, 4], or segmenting organs and tissue in medical applications [5, 6]. Segmentation goes a step beyond standard classification and object detection as it extracts the area in an image that corresponds to the target class or classes at pixel-level precision. This comes with two advantages that are important in the medical field. The first one is that the algorithm learns pixel-wise and has more examples to learn from compared to if it would learn image-wise [7, 8]. This can help for use cases where one does not have many images from a disease. Secondly, the segmented area makes it easier for the physician to determine what the algorithm

3rd International Workshop and Challenge on Computer Vision in Endoscopy (EndoCV2021) in conjunction with the 18th IEEE International Symposium on Biomedical Imaging ISBI2021, April 13th, 2021, Nice, France

Corresponding author: vajira@simula.no (V. Thambawita)
0000-0001-6026-0929 (V. Thambawita); 0000-0002-3332-1201 (S. A. Hicks); 0000-0003-2073-7029 (P. Halvorsen); 0000-0002-3153-2064 (M. A. Riegler)

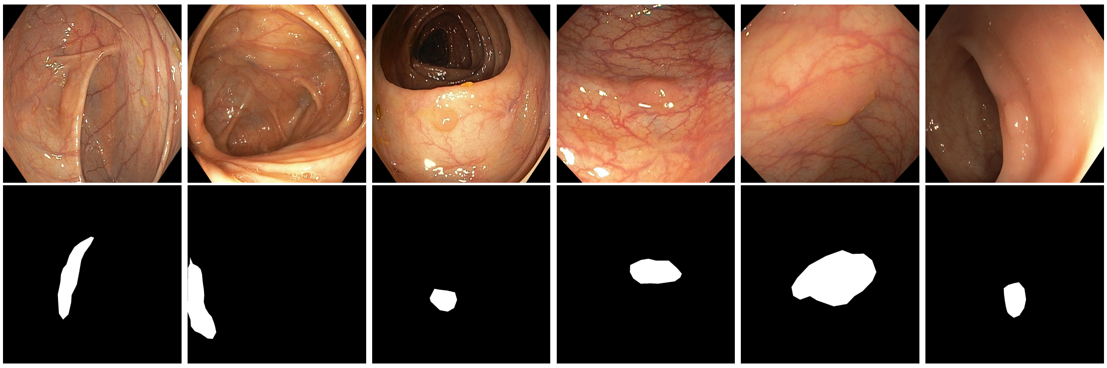

Figure 1: Some example images and their corresponding ground truth masks taken from the development dataset of EndCV2021 [10]

detected and classified as a disease which serves in a broader sense as an explanation. Thus, detailed image segmentation can also be seen as a type of explanation method. This makes it highly desired by medical professionals as explainable machine learning is seen as one of the requirements for the successful implementation of automatic decision support systems in hospitals [9]. As part of the EndoCV2021 challenge (https://endocv2021.grand-challenge.org/), we were tasked with creating machine learning models that automatically segment polyps [11, 12, 13] in video frames collected from real-world endoscopies. This is a complex task as polyps come in various shapes and sizes, where some (e.g., flat lesions) are barely detectable by even the most experienced endoscopists. Figure 1 shows some of the more difficult examples taken from EndoCV's development dataset [10] provided by the challenge organizers. The challenge presented two separate tasks, the *detection generalization challenge* and the *segmentation generalization challenge*. We participated in the segmentation generalization challenge, where we achieved the best results among 13 other competitors in both rounds. The code for the experiments presented in this paper is available on GitHub[1].

This paper summarizes our approaches to the EndoCV2021 challenge. In particular, we developed the *TriUNet* segmentation model combining three separate UNet models, and the *DivergentNets* that combines TriUNet with an ensemble of the well-known segmentation models, namely UNet++, FPN, DeepLabv3, and DeepLabv3+. The rest of the paper is structured as follows. Section 2 present our approach to this year's challenge, where we use two unique models that achieve state-of-the-art performance on the EndoCV dataset. Section 3 gives a description of the implementation details on the models and training procedure and how the data was split and prepared. Section 4 presents the preliminary and official results for our tested models and performs a qualitative analysis on some of the predicted masks. Lastly, Section 5 concludes this paper with a summary and plans for future work.

[1] https://github.com/vlbthambawita/divergent-nets

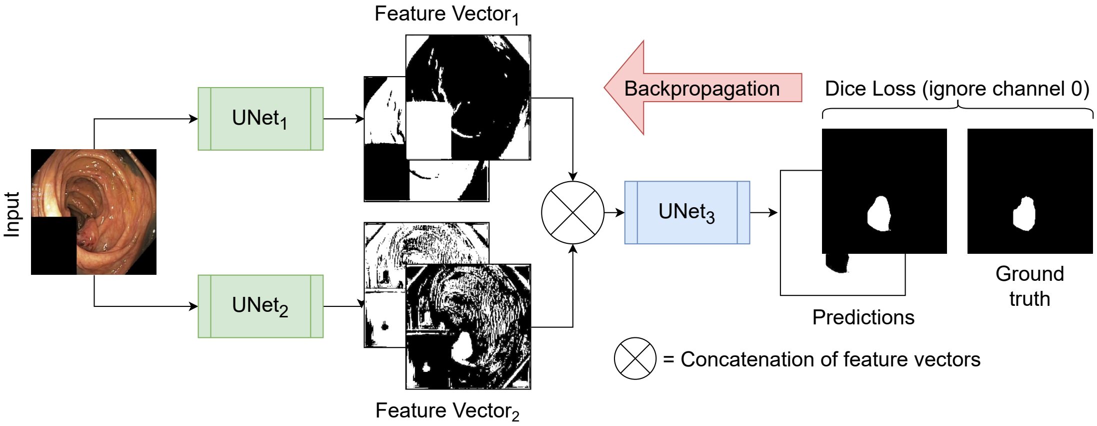

Figure 2: An illustration of the TriUNet architecture. First, the image is passed through two separate UNets in parallel, which produce the feature vectors V_1 and V_2, respectively. These two vectors are then concatenated before being passed through a third UNet that predicts the final segmentation mask. The loss is calculated by taking the Dice coefficient of the mask corresponding to the main class and the ground truth, which is then back-propagated through the entire model.

2. Approach

In this section, we introduce three approaches that we developed for the segmentation generalization challenge at EndoCV 2021, which are two new architectures, TriUNet and DivergentNet, and a modified loss function.

2.1. TriUNet

TriUNet is a convolutional neural network (CNN) architecture that utilizes multiple UNet [14] architectures arranged in a triangular structure as depicted in Figure 2. The model takes a single image as input, which is passed through two separate UNet models with different randomized weights. The output of both models is then concatenated before being passed through a third UNet model to predict the final segmentation mask. Figure 2 also shows an example of the intermediate representations provided by the two initial UNet models. The loss is calculated and back-propagated through the whole model, meaning the entire network is trained in one go. From the intermediate representations, we clearly see that the different UNets learn different interpretations of the data, which then are combined in one final output.

2.2. DivergentNets

The DivergentNets network is inspired by the idea of ensembles made with multiple high-performing image segmentation architectures and the TriUNet architecture presented in the previous section. We constructed this DivergentNets assuming that cumulative decisions taken from multiple intermediate models should give a more precise decision than the predictions from a single network. The included models were selected based on what has previously been shown to produce good results on different segmentation tasks and some preliminary experiments

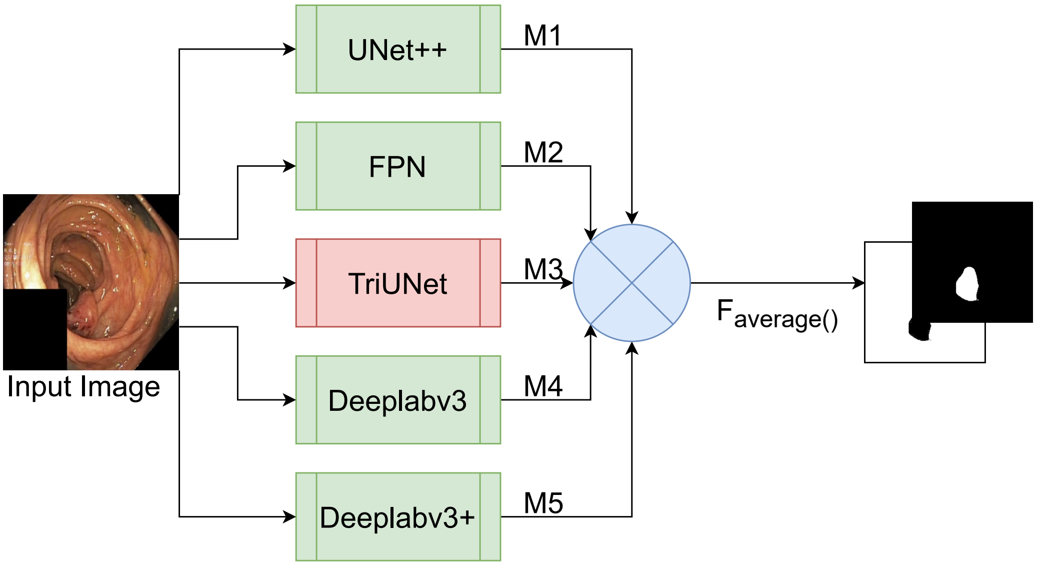

Figure 3: An illustration of the DivergentNets architecture. First, five different models are trained using the U-Net++, FPN, TriUnet, DeepLabv3, and DeepLabv3 architectures. Then, an image is passed through each model separately, which produces masks M_1 to M_5. Last, the masks are averaged to make the final segmentation mask.

using each model independently. Furthermore, the selection was limited by the hardware we had available.

As shown in Figure 3, our configuration comprises five intermediate models, namely UNet++, FPN, DeepLabv3, DeepLabv3+, and TriUNet. The five intermediate models are first trained for N number of epochs separately, where the best checkpoint of each model is selected to be combined in DivergentNet. This N should be selected using a preliminary experiment. In our case, we identified that $N = 200$ is enough to produce high-quality masks. However, training for more epochs may result in better checkpoints to use in DivergentNet. To produce the intermediate masks, the output of each model is passed through a *softmax2d* activation function. However, this should be changed based on the application. In our case, we predict masks for two classes, background and polyp, where no two categories may overlap. The masks produced by each intermediate model represent the divergent views on the data. The final output of DivergentNets is made by averaging the pixels between each intermediate mask and rounding to the nearest integer (either 0 or 1).

2.3. Single-channel Dice

All models were trained to predict masks for both polyps and background (mostly containing the mucosal wall lining the inside of the colon). As the primary focus of EndoCV is to segment colon polyps, we use a modified Dice loss to calculate the prediction error. We call this loss function *single-channel Dice loss* as it only considers one channel when calculating error. This

Table 1
An overview of how the data was split between training, validation, and testing.

Dataset	Partition	# Samples	# Polyp	# Non-Polyp
EndoCV	Training	1,754	1,329	435
EndoCV	Validation	2,756	1,400	1,356
HyperKvasir	Testing	1,000	1,000	0

is shown in Equation 1:

$$\text{Single-Channel Dice Loss} = \frac{2 \cdot |A_n \cap B_n|}{2 \cdot |A_n \cap B_n| + |B_n \backslash A_n| + |A_n \backslash B_n|} \tag{1}$$

where n represents the class for which loss should be calculated for. In this case, we only calculate loss for the polyp class and ignore the background.

3. Experiments

The experimenters can be categorized into two sub-groups, namely baseline experiments and experiments used for the challenge. The baseline experiments were used to benchmark common segmentation models. The baseline models tested were UNet [14], UNet++ [15], FPN [16], DeepLabv3 [17], and DeepLabv3+ [18]. In turn, we used these networks to design the TriUNet and DivergentNets architectures. This section describes the experimental setup, including how the data was prepared, training procedures, architecture implementations, and specifics on what hyperparameters were used.

3.1. Data details and preparation

The development dataset provided by the organizers was split between several directories, primarily one part consisting of a five-way center-wise split (directories *C1* through *C5*) containing a diverse set of data [10], and one part consisting of pure sequence data (directories *seq1* through *seq15*). For this challenge, we decided to use a standard three-way split of the data into training, validation, and testing datasets. The training data was made up of all the data contained within the center-wise split for training data, in addition to a few sequences only containing negative samples. For validation, we used the remaining sequence data. Table 1 gives an overview of how each directory was split between training, validation, and test datasets. All samples contained an image, a segmentation mask, bounding box coordinates, and the image with the bounding-box superimposed over it. As we were only participating in the segmentation generalization challenge, we only used the images and segmentation masks.

3.2. Implementation details

All models were implemented in PyTorch and trained on an Nvidia DGX-2. The Nvidia DGX-2 consists of 16 Tesla V100 GPUs, dual Intel Xeon Platinum 816 processors, and 1.5 terabytes of system memory. Despite that the system contains 16 GPUs, we only use one for training

Table 2
The results collected from the preliminary experiments on the internal validation dataset.

Model	All Classes				Polyp Class				Background Class			
	IoU	F1	REC	PREC	IoU	F1	REC	PREC	IoU	F1	REC	PREC
U-Net	0.973	0.985	0.985	0.985	0.774	0.802	0.831	0.926	0.984	0.991	0.996	0.988
U-Net++	0.972	0.984	0.984	0.984	0.787	0.815	0.847	0.918	0.983	0.991	0.995	0.989
FPN	0.973	0.985	0.985	0.985	0.778	0.810	**0.853**	0.904	0.984	0.991	0.995	0.989
DeepLabv3	0.971	0.984	0.984	0.984	0.764	0.798	0.842	0.902	0.983	0.991	0.994	0.989
DeepLabv3+	0.973	0.985	0.985	0.985	0.777	0.807	0.840	0.919	0.984	0.991	0.994	0.989
TriUNet	0.970	0.983	0.983	0.983	0.775	0.802	0.846	0.903	0.982	0.990	0.992	**0.989**
DivergentNets	**0.976**	**0.986**	**0.986**	**0.986**	**0.795**	**0.823**	0.844	**0.937**	**0.986**	**0.992**	0.997	0.989

so that we can train multiple models in parallel. For the baseline experiments, we used the implementations and pre-trained weights available in the *Segmentation Models* [19] library. These networks were also used as the basis for our proposed TriUNet and DivergentNets. Each model was implemented SE-ResNeXt-50-32x4D [20] as the encoder, which was initialized with ImageNet [21] weights. Images and masks were resized to 256×256 and resized back to the original resolution using bilinear interpolation. The final prediction was produced by passing the output through a two-dimensional softmax function. For training, all models started with a learning rate of 0.0001 and reduced to 0.00001 after 50 epochs. The model error was calculated using the proposed single-channel Dice for the polyp class (as explained in Section 2.3), and the weights were optimized using Adam [22].

As the size of the development dataset is relatively small, we use a series of different image augmentations to make the model more generalizable. These augmentations include horizontal flip, shift scale rotation, resizing, additive Gaussian noise, perspective shift, contrast limited adaptive histogram equalization (CLAHE), random brightness, random gamma, random sharpen, random blur, random motion blur, random contrast, and hue saturation. The augmentations were implemented using the Python library *Albumentations* [23]. No augmentations were applied to the validation and testing data.

4. Results and Discussion

In this section, we discuss the preliminary and official results of our approach to the EndoCV 2021 challenge. We also perform a qualitative analysis of the models, showing how the different modes diverge to a final prediction.

4.1. Preliminary results

Table 2 and Table 3 show the initial results on the provided development validation and testing datasets. Overall, we see that all models perform well on segmenting the polyp class, with the DivergentNets architecture achieving the best performance and UNet++ at a close second place on both the validation and test datasets. Comparing UNet and TriUNet, we see that TriUNet performs slightly better on the polyp class, however, UNet++ outperforms both. With these results, it would be natural to assume that a TriUNet++ architecture would perform even better

Table 3
The results collected from the preliminary experiments on the internal testing dataset.

Model	All Classes				Polyp Class				Background Class			
	IoU	F1	REC	PREC	IoU	F1	REC	PREC	IoU	F1	REC	PREC
U-Net	0.941	0.967	0.967	0.967	0.823	0.883	0.876	0.938	0.959	0.977	0.988	0.970
U-Net++	0.945	0.969	0.969	0.969	0.834	0.894	0.882	0.942	0.961	0.979	0.988	0.972
FPN	0.944	0.968	0.968	0.968	0.824	0.887	0.870	0.943	0.961	0.978	**0.990**	0.970
DeepLabv3	0.942	0.968	0.968	0.968	0.821	0.885	0.874	0.935	0.959	0.977	0.988	0.970
DeepLabv3+	0.942	0.968	0.968	0.968	0.823	0.886	0.883	0.931	0.823	0.886	0.883	0.931
TriUNet	0.941	0.967	0.967	0.967	0.829	0.890	**0.891**	0.928	0.959	0.977	0.983	**0.975**
DivergentNets	**0.949**	**0.972**	**0.972**	**0.972**	**0.840**	**0.899**	0.886	**0.946**	**0.964**	**0.980**	0.990	0.973

Table 4
The official results provided by the EndoCV organizers. *Score* is an average score of F1-score, F2-score, PPV, and Recall provided by the organizers, and *SD* is the standard deviation of the metrics.

Round	Model	Score	SD
I	UNet++	0.917	0.168
	TriUNet	*0.925*	*0.152*
II	TriUNet	0.796	0.047
	DivergentNets	*0.823*	*0.043*

than TriUNet. However, due to hardware limitations (specifically GPU memory), we were unable to test this configuration and move this to future work.

4.2. Official results

The official evaluation was split into two rounds, where *Round I* used a subset of the testing data that was fully used for *Round II*. For both rounds, we were limited by the number of submissions that could be delivered per day. This limit started at five-per-day for *Round I* and was reduced to two-per-day for *Round II*. Due to this limitation, only a subset of the aforementioned models was submitted as official runs. Models were selected based on their performance on a different test dataset that we chose, namely the well-known and established HyperKvasir [24] dataset, in ascending fashion. From the HyperKvasir dataset, we only used the images with segmentation masks as an independent test set to determine the best generalizable model. It was not used in any way as training or validation data. Table 4 shows the official results for *Round I* and *Round II*. Note that the DivergentNets model was not part of *Round I* as it was developed during *Round I* and used in *Round II* once it was finished. From the results, we see that TriUNet achieved the best score for *Round I*, and DivergentNets achieved the best score for *Round II*, *i.e.*,both winning their respective rounds of the competition.

4.3. Qualitative analysis

Figure 4 shows some example masks predicted by our best performing model (DivergentNets) together with masks produced by the intermediate models. We see that each intermediate model learns slightly different features, making an overall more precise segmentation mask when

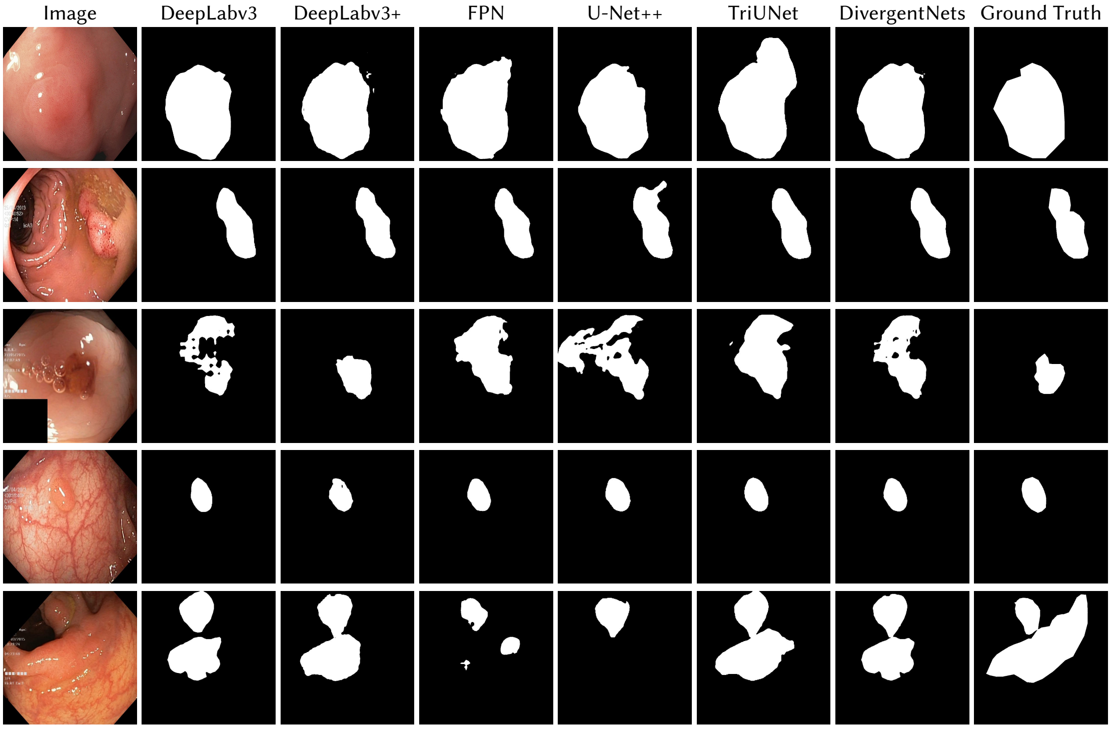

Figure 4: Some predicted mask examples taken from the divergent network and its five intermediate models. The images are taken from HyperKvasir.

combined. For example, the first row of Figure 4 shows the predicted masks and ground truth of a large polyp. We see that each model predicts slightly different masks for the same input and that TriUNet over-estimates the size of the polyp. After averaging the predicted masks for DivergentNets' final output, this area is smoothed out by the predictions from the other intermediate models.

Even though DivergentNets primarily produces more accurate masks than any single model, there are cases where masks from the intermediate model better match the ground truth. We see this in row three, where DeepLabv3+ produces a more precise mask than all other intermediate models, making the averaged output less accurate.

5. Conclusion and future work

In this paper, we presented our approaches to the EndoCV 2021 challenge. We trained a series of baseline models and two models based on novel architectures using a slightly modified Dice loss, which achieved the overall best score in both rounds of the generalization segmentation challenge. For the first round, we developed TriUNet, which reached an average score of 0.925 on the official testing dataset. For the second round, we developed the DivergentNets architecture, which combines the baseline models with the TriUNet to gain an average score of 0.823 on the official training dataset. Due to a limitation on time and computational resources, we could not experiment with another improved version where the UNet architectures are replaced with UNet++ architectures.

For future work, we plan to explore different configurations of TriUNet, such as implementing TriUNet++ and testing different architectures for three architectures that make up the TriUNet architecture, for example, combining the UNet, FPN, and DeepLabv3 as TriUNet nodes. We would also like to explore different configurations for the DivergentNets architecture with different networks for each node. Another idea could be to use a neural network to produce the final prediction instead of the current averaging technique, similar to the approach discussed in [25]. Further testing the approaches with datasets from other medical fields can help to identify the generalizability of our approach.

Acknowledgments

The research has benefited from the Experimental Infrastructure for Exploration of Exascale Computing (eX3), which is financially supported by the Research Council of Norway under contract 270053.

References

[1] O. I. Alirr, Deep learning and level set approach for liver and tumor segmentation from ct scans, Journal of Applied Clinical Medical Physics 21 (2020) 200–209.

[2] S. Pang, A. Du, M. A. Orgun, Z. Yu, Y. Wang, Y. Wang, G. Liu, Ctumorgan: a unified framework for automatic computed tomography tumor segmentation, European Journal of Nuclear Medicine and Molecular Imaging 47 (2020) 2248–2268. doi:10.1007/s00259-020-04781-3.

[3] N. Yamanakkanavar, J. Y. Choi, B. Lee, Mri segmentation and classification of human brain using deep learning for diagnosis of alzheimer's disease: a survey, Sensors 20 (2020). doi:10.3390/s20113243.

[4] L. Clarke, R. Velthuizen, M. Camacho, J. Heine, M. Vaidyanathan, L. Hall, R. Thatcher, M. Silbiger, Mri segmentation: Methods and applications, Magnetic Resonance Imaging 13 (1995) 343–368.

[5] P. M. Scheikl, S. Laschewski, A. Kisilenko, T. Davitashvili, B. Müller, M. Capek, B. P. Müller-Stich, M. Wagner, F. Mathis-Ullrich, Deep learning for semantic segmentation of organs and tissues in laparoscopic surgery, Current Directions in Biomedical Engineering 6 (2020). doi:10.1515/cdbme-2020-0016.

[6] O. Schoppe, C. Pan, J. Coronel, H. Mai, Z. Rong, M. I. Todorov, A. Müskes, F. Navarro, H. Li, A. Ertürk, B. H. Menze, Deep learning-enabled multi-organ segmentation in whole-body mouse scans, Nature Communications 11 (2020) 5626. doi:10.1038/s41467-020-19449-7.

[7] T. Sun, W. Zhang, Z. Wang, L. Ma, Z. Jie, Image-level to pixel-wise labeling: From theory to practice, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2018, pp. 928–934. doi:10.24963/ijcai.2018/129.

[8] A. Bansal, X. Chen, B. Russell, A. Gupta, D. Ramanan, Pixelnet: Representation of the pixels, by the pixels, and for the pixels, arXiv preprint arXiv:1702.06506 (2017).

[9] J. Amann, A. Blasimme, E. Vayena, D. Frey, V. I. Madai, Explainability for artificial intelligence in healthcare: a multidisciplinary perspective, BMC Medical Informatics and Decision Making 20 (2020) 1–9.

[10] S. Ali, D. Jha, N. Ghatwary, S. Realdon, R. Cannizzaro, M. A. Riegler, P. Halvorsen, C. Daul, J. Rittscher, O. E. Salem, D. Lamarque, T. de Lange, J. E. East, Polypgen: A multi-center polyp detection and segmentation dataset for generalisability assessment, arXiv (2021).

[11] S. Ali, M. Dmitrieva, N. Ghatwary, S. Bano, G. Polat, A. Temizel, A. Krenzer, A. Hekalo, Y. B. Guo, B. Matuszewski, M. Gridach, I. Voiculescu, V. Yoganand, A. Chavan, A. Raj, N. T. Nguyen, D. Q. Tran, L. D. Huynh, N. Boutry, S. Rezvy, H. Chen, Y. H. Choi, A. Subramanian, V. Balasubramanian, X. W. Gao, H. Hu, Y. Liao, D. Stoyanov, C. Daul, S. Realdon, R. Cannizzaro, D. Lamarque, T. Tran-Nguyen, A. Bailey, B. Braden, J. E. East, J. Rittscher, Deep learning for detection and segmentation of artefact and disease instances in gastrointestinal endoscopy, Medical Image Analysis 70 (2021) 102002. doi:10.1016/j.media.2021.102002.

[12] S. Ali, F. Zhou, B. Braden, A. Bailey, S. Yang, G. Cheng, P. Zhang, X. Li, M. Kayser, R. D. Soberanis-Mukul, S. Albarqouni, X. Wang, C. Wang, S. Watanabe, I. Oksuz, Q. Ning, S. Yang, M. A. Khan, X. W. Gao, S. Realdon, M. Loshchenov, J. A. Schnabel, J. E. East, G. Wagnieres, V. B. Loschenov, E. Grisan, C. Daul, W. Blondel, J. Rittscher, An objective comparison of detection and segmentation algorithms for artefacts in clinical endoscopy, Scientific Reports 10 (2020) 2748. doi:10.1038/s41598-020-59413-5.

[13] D. Jha, S. Ali, N. K. Tomar, H. D. Johansen, D. Johansen, J. Rittscher, M. A. Riegler, P. Halvorsen, Real-time polyp detection, localization and segmentation in colonoscopy using deep learning, IEEE Access 9 (2021) 40496–40510.

[14] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi (Eds.), Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer International Publishing, Cham, 2015, pp. 234–241.

[15] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, J. 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, Springer, 2018, pp. 3–11. doi:10.1007/978-3-030-00889-5_1.

[16] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2017, pp. 2117–2125.

[17] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, arXiv preprint arXiv:1706.05587 (2017).

[18] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 801–818.

[19] P. Yakubovskiy, Segmentation models pytorch, 2020.

[20] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2018, pp. 7132–7141.

[21] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision (IJCV) 115 (2015) 211–252. doi:10.1007/s11263-015-0816-y.

[22] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).

[23] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, A. A. Kalinin, Albumentations: fast and flexible image augmentations, Information 11 (2020) 125.

[24] H. Borgli, V. Thambawita, P. H. Smedsrud, S. Hicks, D. Jha, S. L. Eskeland, K. R. Randel, K. Pogorelov, M. Lux, D. T. D. Nguyen, et al., Hyperkvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy, Scientific Data 7 (2020) 1–14. doi:10.1038/s41597-020-00622-y.

[25] V. Thambawita, D. Jha, H. L. Hammer, H. D. Johansen, D. Johansen, P. Halvorsen, M. A. Riegler, An extensive study on cross-dataset bias and evaluation metrics interpretation for machine learning applied to gastrointestinal tract abnormality classification, ACM Trans. Comput. Healthcare 1 (2020). doi:10.1145/3386295.