Bigger Networks are not Always Better: Deep Convolutional Neural Networks for Automated Polyp Segmentation

Adrian Krenzer[1], Frank Puppe[1]
[1]Julius-Maximilian University of Würzburg, Germany
adrian.krenzer@uni-wuerzburg.de,frank.puppe@uni-wuerzburg.de

ABSTRACT

This paper presents our team's (AI-JMU) approach to the Medico automated polyp segmentation challenge. We consider deep convolutional neural networks to be well suited for this task. To determine the best architecture we test and compare state of the art backbones and two different heads. Finally, we achieve a Jaccard index of 73.74% on the challenge's test set. We further demonstrate that bigger networks do not always perform better. However, the growing network size always increases the computational complexity.

1 INTRODUCTION

Worldwide colorectal cancer (CRC) represents the third most commonly diagnosed cancer [6, 17]. According to Herszenyi and Tulassay [10] CRC attributes to 9% of all cancer incidence globally and is the fourth cause of cancer death worldwide [6, 17]. In order to detect potentially cancerous tissues early, physicians conduct a so-called colonoscopy. During this procedure, the physician searches for polyps inside the colon in order to remove them. Polyps are abnormally growing tissues that usually look like small, flat bumps or tiny fungal stems. Due to the aberrant cell growth, they can eventually become malignant or cancerous. Nevertheless, even the best physicians have a risk of overlooking a polyp. Missed polyps are not removed and can therefore have fatal consequences. Automated detecting and segmenting polyps is the task of the medico challenge [12]. This challenge is special because it is not allowed to include training data other than the 1000 provided polyp images of Jha et al. [13]. In this paper, we present our challenge results and explain how we select the networks for our final predictions.

2 RELATED WORK

In the domain of object segmentation with deep learning, there are two general state of the art approaches: Fully convolutional networks [7, 16, 21] and encoder-decoder architectures [1, 5, 24]. Some state of the art polyp segmentation methods include encoder-decoder architectures. However due to the high computational complexity of those models, polyp segmentation research focuses mostly on fully convolutional architectures to enable real-time segmentation systems [11, 28]. We consider our approaches to belong to the field of fully convolutional networks. The chosen models are based on our previous study [14], which we advance for this challenge by: Focusing exclusively on polyp segmentation, testing a new state of the art backbone in polyp segmentation [27] and comparing different architectures comprehensively.

MediaEval'20, December 14-15 2020, Online

3 APPROACH

This section focuses on our approaches for the Medico automated polyp segmentation tasks. We train all our models using a Tesla Turing RTX 8000 Nvidia GPU. For this challenge, Deep CNNs are best suited as they provide very stable outcomes in multi-class segmentation tasks like the COCO challenge [15]. Since both bounding boxes and segmentation masks are available in the dataset, we choose networks that can handle both inputs. Therefore we select the Mask R-CNN [8] and the Cascade Mask R-CNN [3]. We build both architectures based on two-stage object detection models using Faster R-CNN [19]. Therefore a region proposal network first suggests candidate bounding boxes (Regions of Interest, RoI) before making the final prediction. In this case, an additional branch is added designed to predict segmentation masks, where the suggested RoIs enhance the segmentation mask predictions. A Cascade Mask R-CNN uses an extended framework which is defined by a cascade-like composition utilizing several Mask R-CNNs with shared weight on the backbones. We train both the Cascade Mask R-CNN and Mask R-CNN with the open-source Detectron2 framework [23].

We select these types of models because of two rationales: First, the availability of both bounding boxes and segmentation masks for training purposes allows us to maximize the Mask R-CNN performance, because RoI and segmentation are closely related. Second, because the mask of polyps included in the KVASIR-SEG dataset [13] often vary significantly in size and shape we desire a network that is unaffected by those variations and determines a pixel-wise mask of the polyp. Because we use semantic segmentation, we deal with this as an instance segmentation defined by a single instance per incident per class. Therefore, we alter the ground truth bounding boxes in our data to include only one instance instead of multiple instances.

We test the Cascade Mask R-CNN and Mask R-CNN with ResNet [9] as well as the new ResNeSt [27] backbone. The latter adds a split attention block to the ResNet backbone and reconfigures the ResNet structure. This block and structure enable the network to share attention across feature-map groups. This might offer some benefits to the polyp segmentation task. Additionally, we vary the depth of both backbones, with depths of 50 and 101 for ResNet as well as 50, 101, and 200 for ResNeSt. The backbones we use consist of CNN classifiers pre-trained using the ImageNet-1k dataset [20]. The whole architecture is pre-trained on the COCO dataset [15]. Consequentially we use transfer learning to compensate for the small size of the training dataset. We train networks with the Detectron2 framework [23] and a fork of the Detectron2 framework published by Zhang et al. [27]. Both provide a wide range of pre-trained object detection and segmentation models. Prior to the actual processing, we convert our data to the COCO dataset format. Afterward, the required image preprocessing steps, i.e. padding,

Table 1: Segmentation results on the validation data. R50 and R101 denote ResNet50 and ResNet100. Rt50, Rt101 and Rt200 denote ResNeSt50, ResNeSt101 and ResNeSt200. Cascade R-CNN denotes Cascade Mask R-CNN. All values excluding FPS are in %.

	Mask R-CNN				Cascade R-CNN			
	IoU	Dice	Acc	FPS	IoU	Dice	Acc	FPS
R50	71.0	78.9	90.9	**13**	73.2	81.4	93.8	9.8
R101	72.3	80.0	91.8	11.8	74.1	82.1	94.3	8.7
Rt50	72.8	78.7	90.4	10.9	75.2	81.9	94.2	8.2
Rt101	73.9	80.8	93.2	9.0	**75.9**	**83.1**	**95.7**	7.1
Rt200	-	-	-	-	73.3	81.6	93.4	2.9

resizing, rescaling the pixel values, etc., are automatically performed within the framework.

We define the total loss as the sum of classification, box-regression and mask loss $L = L_{\text{cls}} + L_{\text{box}} + L_{\text{mask}}$ [8], where L_{mask} is the binary cross-entropy for autonomous segmentation of all masks. The training of all models includes a stochastic gradient descent using a learning rate of 0.00025 and a batch size of 16. Every model trains for up to 80000 iterations, maintaining checkpoints every 300 iterations. Afterward, we adopt the checkpoint with the lowest validation loss for the final outcome. Additionally, we utilize random horizontal flipping, vertical flipping, and random resizing as data augmentation while retaining aspect ratio to diminish the generalization error.

4 RESULTS AND ANALYSIS

We evaluate the models on our validation dataset which is a subset of the Kvasir-SEG data [13]. For the evaluation we consider quality and speed. For quality we compute the dice coefficient, intersection over union (IoU), and accuracy (Acc). For speed we specify frames per second (FPS). All our validations are carried out using an Nvidia V100 GPU within the cloud solution of Google Colab [2]. Table 1 depicts our results. While Cascade Mask R-CNN outperforms Mask R-CNN in every quality metric, Mask R-CNN is faster with computation. However, the architecture's speed shows a clear pattern: the Mask R-CNN using the smallest backbone (lowest computational complexity) is the fastest, and Cascade Mask R-CNN (highest computational complexity) with the largest backbone is the slowest. Comparing the ResNet and RestNeSt backbone: Using the ResNeSt backbone results in higher scores in all metrics. Nevertheless, the RestNeSt backbone increases the computational complexity and therefore decreases FPS. Concerning the depth of the network: Changing the depth from 50 to 101 increases the quality of the results. This implies that a deeper backbone may always result in better quality. However, our results show that a larger backbone not always causes better quality, but always diminishes the speed due to higher computational complexity, in our case dropping FPS down to 2.9 for ResNeSt200. We evaluate ResNeSt200 backbone only with the Cascade Mask R-CNN because there are no pre-trained weights available for the Mask R-CNN version.

Overall, Cascade Mask R-CNN with a ReStNest101 backbone provides the best quality results. Therefore, we consider this backbone

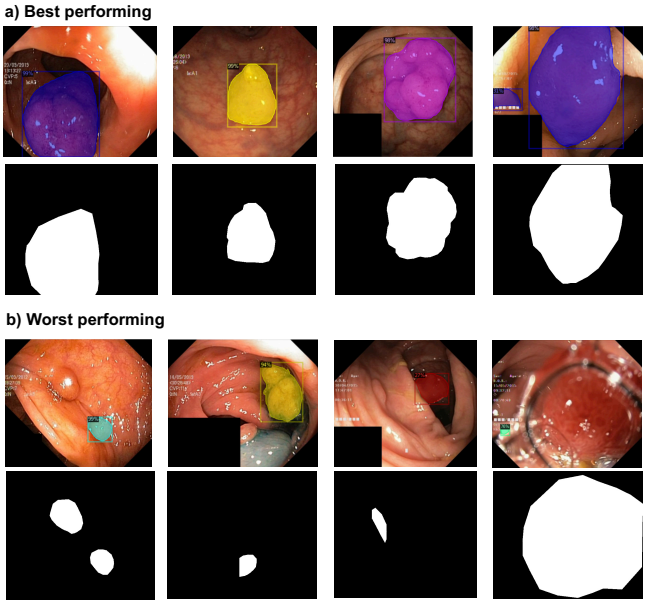

Figure 1: Qualitative results of the Cascade Mask R-CNN with ResNeSt101 backbone. Binary images are ground truth and rgb images with are predictions. Different colors are just highlighting the predictions.

for the quality task of the Medico challenge. For the efficacy task of the challenge we choose the Cascade Mask R-CNN with ReStNest50 backbone. It is faster and less taxing on memory than ReStNest101 while still maintaining high-quality results. Our challenge scores for the quality task are an IoU of 0.737. For the efficacy task our results are an IoU of 0.721 while computing with 3.36 FPS on an Nvidia GTX 1080. To qualitatively demonstrate a set of our results, we depict the four best and worst classified images of our validation set in figure 1. The algorithm performs best on big, unconcealed polyps. Nevertheless, small polyps like shown in the first three images of figure 1_b are harder to segment. In addition, concealing the polyp with a tool like in the last image of figure 1_b prevents the algorithm from detecting the polyp.

5 CONCLUSION AND OUTLOOK

In summary, our results suggest that using a deeper neural network, extending it with another backbone, or adding a computationally more expensive architecture like Cascade Mask R-CNN leads to higher quality segmentations. Nevertheless, the increasing network size is not always beneficial. Moreover, we demonstrate that the ReStNeSt101 backbone combined with the Cascade Mask R-CNN structure is the best segmentation algorithm among our examples.

Further research could extend our architectures and compare them with other state of the art segmentation models like the DeepLabv3+ [18], HRNet [22], MRFM [26]. Those three architectures and the proposed architecture are currently the best performing architectures on object segmentation benchmarks [18, 22, 26, 27]. Especially promising is the speed and quality trade off using HarDNet [4] and BiSeNet [25] for further evaluations.

REFERENCES

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence* 39, 12 (2017), 2481–2495.

[2] Ekaba Bisong. 2019. Google Colaboratory. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Springer, 59–64.

[3] Zhaowei Cai and Nuno Vasconcelos. 2019. Cascade R-CNN: High Quality Object Detection and Instance Segmentation. *CoRR* abs/1906.09756 (2019). arXiv:1906.09756 http://arxiv.org/abs/1906.09756

[4] Ping Chao, Chao-Yang Kao, Yu-Shan Ruan, Chien-Hsiang Huang, and Youn-Long Lin. 2019. HarDNet: A Low Memory Traffic Network. *CoRR* abs/1909.00948 (2019). arXiv:1909.00948 http://arxiv.org/abs/1909.00948

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

[6] Pasqualino Favoriti, Gabriele Carbone, Marco Greco, Felice Pirozzi, Raffaele Emmanuele Maria Pirozzi, and Francesco Corcione. 2016. Worldwide burden of colorectal cancer: a review. *Updates in surgery* 68, 1 (2016), 7–11.

[7] Chaoyi Han, Yiping Duan, Xiaoming Tao, and Jianhua Lu. 2019. Dense convolutional networks for semantic segmentation. *IEEE Access* 7 (2019), 43369–43382.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. 2017. Mask R-CNN. *CoRR* abs/1703.06870 (2017). arXiv:1703.06870 http://arxiv.org/abs/1703.06870

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *CoRR* abs/1512.03385 (2015). arXiv:1512.03385 http://arxiv.org/abs/1512.03385

[10] Laszlo Herszenyi and Zsolt Tulassay. 2010. Epidemiology of gastrointestinal and liver tumors. *Eur Rev Med Pharmacol Sci* 14, 4 (2010), 249–258.

[11] Debesh Jha, Sharib Ali, Håvard D Johansen, Dag D Johansen, Jens Rittscher, Michael A Riegler, and Pål Halvorsen. 2020. Real-Time Polyp Detection, Localisation and Segmentation in Colonoscopy Using Deep Learning. *arXiv preprint arXiv:2011.07631* (2020).

[12] Debesh Jha, Steven A. Hicks, Krister Emanuelsen, Håvard D. Johansen, Dag Johansen, Thomas de Lange, Michael A. Riegler, and Pål Halvorsen. Medico Multimedia Task at MediaEval 2020:Automatic Polyp Segmentation.

[13] Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D Johansen. 2020. Kvasir-SEG: A Segmented Polyp Dataset. In *Proc. of International Conference on Multimedia Modeling (MMM)*. 451–462.

[14] Adrian Krenzer, A. Hekalo, and F. Puppe. 2020. Endoscopic Detection And Segmentation Of Gastroenterological Diseases With Deep Convolutional Neural Networks. In *EndoCV@ISBI*.

[15] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[16] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3431–3440.

[17] Michael Marmot, T Atinmo, T Byers, J Chen, T Hirohata, A Jackson, W James, L Kolonel, S Kumanyika, C Leitzmann, and others. 2007. Food, nutrition, physical activity, and the prevention of cancer: a global perspective. (2007).

[18] Heungmin Oh, Minjung Lee, Hyungtae Kim, and Joonki Paik. 2020. Metadata Extraction Using DeepLab V3 and Probabilistic Latent Semantic Analysis for Intelligent Visual Surveillance Systems. In *2020 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 1–2.

[19] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *CoRR* abs/1506.01497 (2015). arXiv:1506.01497 http://arxiv.org/abs/1506.01497

[20] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. 2014. ImageNet Large Scale Visual Recognition Challenge. *CoRR* abs/1409.0575 (2014). arXiv:1409.0575 http://arxiv.org/abs/1409.0575

[21] Evan Shelhamer, Jonathan Long, and Trevor Darrell. 2017. Fully convolutional networks for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence* 39, 4 (2017), 640–651.

[22] Andrew Tao, Karan Sapra, and Bryan Catanzaro. 2020. Hierarchical Multi-Scale Attention for Semantic Segmentation. *arXiv preprint arXiv:2005.10821* (2020).

[23] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. 2019. Detectron2. https://github.com/facebookresearch/detectron2. (2019).

[24] Jimei Yang, Brian Price, Scott Cohen, Honglak Lee, and Ming-Hsuan Yang. 2016. Object contour detection with a fully convolutional encoder-decoder network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 193–202.

[25] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. 2020. BiSeNet V2: Bilateral Network with Guided Aggregation for Real-time Semantic Segmentation. *arXiv preprint arXiv:2004.02147* (2020).

[26] Jianlong Yuan, Zelu Deng, Shu Wang, and Zhenbo Luo. 2020. Multi Receptive Field Network for Semantic Segmentation. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1883–1892.

[27] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Mueller, R. Manmatha, Mu Li, and Alexander Smola. 2020. ResNeSt: Split-Attention Networks. (2020). arXiv:cs.CV/2004.08955

[28] Jiafu Zhong, Wei Wang, Huisi Wu, Zhenkun Wen, and Jing Qin. 2020. PolypSeg: An Efficient Context-Aware Network for Polyp Segmentation from Colonoscopy Videos. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 285–294.