Pyramid-Focus-Augmentation: Medical Image Segmentation with Step-Wise Focus

Vajira Thambawita[1,2], Steven Hicks[1,2], Pål Halvorsen[1,2], Michael A. Riegler[1]
[1]SimulaMet, Norway [2]Oslo Metropolitan University, Norway
Contact:vajira@simula.no

ABSTRACT

Segmentation of findings in the gastrointestinal tract is a challenging but also important task which is an important building stone for sufficient automatic decision support systems. In this work, we present our solution for the Medico 2020 task, which focused on the problem of colon polyp segmentation. We present our simple but efficient idea of using an augmentation method that uses grids in a pyramid-like manner (large to small) for segmentation. Our results show that the proposed methods work as indented and can also lead to comparable results when competing with other methods.

1 INTRODUCTION

Segmented polyp regions in Gastrointestinal Tract (GI) images [1] can provide detailed analysis to doctors to identify correct areas to proceed with treatments compared to other computer-aided analysis such as classification [2, 9, 10] and detection [7] which provide less detailed information about the exact region and size of the affected area. However, training Deep Learning (DL) models to perform segmentation for medical data is challenging because of the lack of medical domain images as a result of tight privacy restrictions, the high cost for annotating medical data using experts, and a lower number of true positive findings compared to true negatives. In this paper, we present our approach for the participation in the 2020 Medico Segmentation Challenge [4], for which we introduce a novel augmentation technique called **pyramid-focus-augmentation (PYRA)**. PYRA can be used to improve the performance of segmentation tasks when we have a small dataset to train our DL models or if the number of positive findings is small. Further, our method can focus doctors' attention to regions of polyps gradually. In addition to that the output of the method is also adjustable meaning, we could present a lower resolution of the grid if this is sufficient for the task at hand which can help to save processing time. Finally, our technique can also be applied to any segmentation task using any deep learning segmentation model.

2 METHOD

Our method has two main steps: data augmentation with PYRA using pre-defined grid sizes followed by training of a DL model with the resulting augmented data. The source code for our method can be found in our GitHub[1] repository. The development dataset [5] provided by the organizers has 1000 polyp images with corresponding ground truth masks. We divided it into two parts such that 800 images are used for model training and 200 for testing.

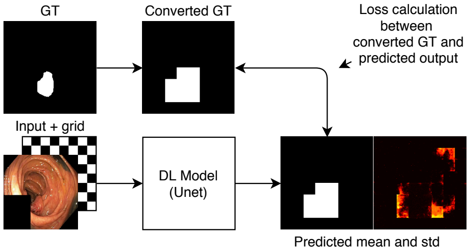

Figure 1: Training steps for a segmentation model with the new augmentation technique.

2.1 PYRA Data Augmentation

As the first step in PYRA, we generate checker board grids as illustrated in the first row of Figure 2 with sizes of $N \times N$ with N values of 4, 8, 16, 32, 64, 128 and 256. N should be selected such that $image\_size \ \% \ N = 0$. Applying these eight grid augmentations to the training dataset with 800 images increases the training data to $800 \times 8 = 6400$ images.

For the second step, we convert the Ground Truth (GT) segmentation masks into a grid-based representation of the GT corresponding to the grid sizes. For example, if the grid size is 8×8, then the corresponding GT is a 8×8 converted GT.

The transformation of the ground truth masks to gridded masks is performed as following: (i) we divide the gt into the input grid size, (ii) we counted true pixels of each grid cell, (iii) if the number of true pixels is larger than 0, we converted the whole cell into a true cell. An example of a converted GT is depicted on the top of Figure 1.

2.2 Experimental Setup and Model training

We have set up four experiments: Exp-1, Exp-2, Exp-3, and Exp-4 to show the performance of PYRA. Exp-1 and Exp-2 represent two baseline experiments. Exp-1 uses only the 800 training images without any augmentations. In Exp-2, we used general augmentations such as Affine, Coarse Dropout, and Additive Gaussian Noise from the library called *imgaug* [6]. Exp-3 and Exp-4 are using our PYRA with the data from Exp-1 and Exp-2, respectively. The training dataset size was changed from 800 to 6400 after applying PYRA. However, we validated our experiments only using 200 images reserved for testing. We have used one data loader for all experiments to maintain a fair evaluation. The baseline experiments Exp-1 and Exp-2 used the data loader with a grid size of 256×256 which represents the original GT masks without any conversion.

[1] GitHub: https://vlbthambawita.github.io/PYRA/

MediaEval'20, December 14-15 2020, Online

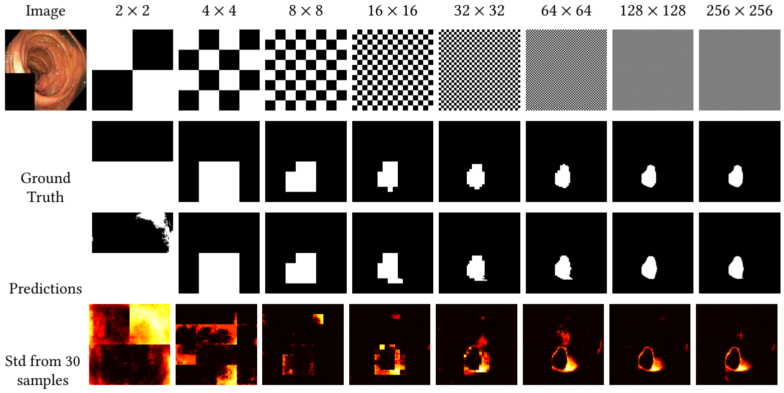

Figure 2: A representation of input and corresponding outputs of grid-augmentation-based segmentation. The first row shows an input image and all grid sizes used as stacked grid image with the input image. The second row represent ground truth. The third and fourth rows show predicted mean and std output images calculated from 30 samples.

Table 1: Result collected from validation data and test data. All test data results were provided by organizers of Medico task in MediaEval 2020.

	Validation results		Official test results	
Method	mIOU	Dice	mIOU	Dice
Exp-1	0.7640	0.8422	0.6934	0.7817
Exp-2	0.7077	0.7957	0.6759	0.7700
Exp-3	**0.7693**	**0.8447**	**0.6981**	**0.7887**
Exp-4	0.6898	0.7822	0.6696	0.7665

We have used the Unet architecture [8] as our DL model to perform the polyp segmentation task. We trained the Unet model with a stacked input using a polyp image and a random grid image selected from the eight sizes. Then, the model was trained to predict converted GT which were formed by converting the real GT into a grid-based GT as in the previous section.

The Unet model used dropout layers with the probability of 0.5. Then, we used our Unet model as a stochastic model to perform Monte Carlo sampling for the validation data. We kept our Unet model in the training state to perform this sampling while predicting the output for the validation data. In the Pytorch library, which is used for all our implementations, we can do this simply by keeping the model state in the `model.train()` state. We iterated 50 times for a single input to predict the output. We calculated the mean from these 50 predictions, which is used as the final prediction for the competition and Standard Deviation (std) images to know the model's confidence for the predictions. The whole training process is illustrated in Figure 1 with an example image and a grid size of 8 × 8 as an input. However, we submitted the predicted mean images for the gird size of 256 × 256 which generate predictions with the size of true GT (without any transformations). All the experiments used a fixed learning rate of 0.001 with the RMSprop optimizer [3], which were selected from preliminary experiments.

3 RESULT AND DISCUSSION

Table 1 summarizes the Mean Intersection over Union (mIoU) and the Dice Coefficient (DC) for the validation dataset and the test dataset. The final results to the competition were collected from mean images calculated by sampling 50 outputs for the same input with the grid size of 256. Additionally, we have calculated std images for the validation dataset to show the benefits of using PYRA. Example outputs for a given input image are illustrated in Figure 2.

According to the results in Table 1, Exp-3 which use only Pyramid-focus-augmentation shows the best validation results with mIoU of 0.7693 and DC of 0.8447, and the best test results with mIoU of 0.6981 and DC of 0.7887. The advantage of our Pyramid-focus-augmentation can be identified using the third row of Figure 2 along the fourth row of the same figure. We can see that our model can focus on polyp regions step by step. The third row of Figure 2 shows how our model predicts correct polyp cells in 2×2, 4×4, 8×8, 16×16, 32×32, 64×64, 128×128 and 256×256 grid sizes, respectively. When we compare this row with the last row of the images of std, we can see that the model has high confidence for the identified polyp regions. For example, it shows high confidence (black color region) for the middle part of the polyps. In contrast, our model shows less confidence (yellow color region) for a polyps' outer borders.

4 CONCLUSION AND FUTURE WORK

In this paper, we presented a novel augmentation method called Pyramid-focus-augmentation (PYRA), which can be used to train segmentation DL methods. Our method shows a large benefit in the medical diagnosis use-case, by focusing a doctors' attention on regions with findings step by step.

Our experiments did not use post-processing to clean up output corresponding to the input grid. In future work, we will evaluate our approach with additional post-processing steps for smaller grid sizes. For example, we can do convolution operations to the output using a convolutional window equal to the input grid size to clean the results. However, post-processing techniques will not improve the final results when the grid size equals the input images' resolution.

5 ACKNOWLEDGMENT

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] M. Akbari, M. Mohrekesh, E. Nasr-Esfahani, S. M. R. Soroushmehr, N. Karimi, S. Samavi, and K. Najarian. 2018. Polyp Segmentation in Colonoscopy Images Using Fully Convolutional Network. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 69–72. https://doi.org/10.1109/EMBC.2018.8512197

[2] Steven Alexander Hicks, Pia H Smedsrud, Pål Halvorsen, and Michael Riegler. 2018. Deep Learning Based Disease Detection Using Domain Specific Transfer Learning. *Proc. of MediaEval*.

[3] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. 2012. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. (2012).

[4] Debesh Jha, Steven A. Hicks, Krister Emanuelsen, Håvard Johansen, Dag Johansen, Thomas de Lange, Michael A. Riegler, and Pål Halvorsen. 2020. Medico Multimedia Task at MediaEval 2020: Automatic Polyp Segmentation. In *Proc. of the MediaEval 2020 Workshop*.

[5] 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 *International Conference on Multimedia Modeling*. Springer, 451–462.

[6] Alexander B. Jung, Kentaro Wada, Jon Crall, Satoshi Tanaka, Jake Graving, Christoph Reinders, Sarthak Yadav, Joy Banerjee, Gábor Vecsei, Adam Kraft, Zheng Rui, Jirka Borovec, Christian Vallentin, Semen Zhydenko, Kilian Pfeiffer, Ben Cook, Ismael Fernández, François-Michel De Rainville, Chi-Hung Weng, Abner Ayala-Acevedo, Raphael Meudec, Matias Laporte, and others. 2020. imgaug. https://github.com/aleju/imgaug. (2020). Online; accessed 01-Nov-2020.

[7] Ji Young Lee, Jinhoon Jeong, Eun Mi Song, Chunae Ha, Hyo Jeong Lee, Ja Eun Koo, Dong-Hoon Yang, Namkug Kim, and Jeong-Sik Byeon. 2020. Real-time detection of colon polyps during colonoscopy using deep learning: systematic validation with four independent datasets. *Scientific Reports* 10, 1 (2020), 8379.

[8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

[9] Vajira Thambawita, Debesh Jha, Hugo Lewi Hammer, Håvard D. Johansen, Dag Johansen, Pål Halvorsen, and Michael A. Riegler. 2020. 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, 3, Article 17 (June 2020), 29 pages. https://doi.org/10.1145/3386295

[10] Vajira Thambawita, Debesh Jha, Michael Riegler, Pål Halvorsen, Hugo Lewi Hammer, Håvard D Johansen, and Dag Johansen. 2018. The medico-task 2018: Disease detection in the gastrointestinal tract using global features and deep learning. *Proc. of MediaEval* (2018).