Heterogeneous model ensemble for polyp detection and tracking in colonoscopy

Amine Yamlahi[1], Patrick Godau[1], Thuy Nuong Tran[1], Lucas-Raphael Müller[1], Tim Adler[1], Minu Dietlinde Tizabi[1], Michael Baumgartner[2], Paul Jäger[3] and Lena Maier-Hein[1]

[1]*Div. Intelligent Medical Systems, German Cancer Research Center (DKFZ), Heidelberg, Germany*
[2]*Div. Medical Image Computing, DKFZ, Heidelberg, Germany*
[3]*Interactive Machine Learning Group, DKFZ, Heidelberg, Germany*

Abstract

Regular colonoscopy screening substantially contributes to the prevention of colon cancer, as a polyp found in early stages can safely be removed. Assisting physicians during screening with automated detection systems can potentially increase the sensitivity of polyp detection. In this work, we present our polyp detection and tracking approach, submitted to the EndoCV2022 challenge. The core of our method is a heterogeneous ensemble of YOLOv5 models, each trained with a different strategy based on external data and varying data augmentation concepts. The output of the ensemble members is merged with the weighted boxes fusion algorithm, and the final output bounding boxes are reduced in size. Our method yields a mean Average Precision (mAP) of 0.44 on our validation test set.

Keywords

Polyp detection, model ensembling, image augmentation

1. Introduction

Colorectal cancer is one of the most commonly found cancer types, ranking second in females and third in males [1]. By detecting and subsequently resecting polyps during colonoscopy screenings, the risk of developing the disease can be reduced significantly. With the advance of machine learning in the medical domain, deep learning-based methods have the potential to assist in detecting these polyps with high accuracy. Generalizability across diverse and heterogeneous populations, devices and hospitals is a major issue regarding these methods that needs to be addressed to allow for realistic clinical translation. The method presented in this paper tackles this issue by ensembling heterogeneous, complementary training strategies (see Figure 1). The remaining part of this paper is structured as follows: Sec. 2 first introduces the data we use and goes on to describe all steps of training and post-processing the outputs of the models in the ensemble. Cross-validation results, including ablations, are reported in sec. 3, which is followed by a brief discussion in sec. 4.

2. Methods

Our strategy for algorithm design comprised the following steps:

1. Data preparation: Identification and curation (sec. 2.1) as well as splitting (sec. 2.2) of relevant datasets.
2. Ensemble training: Development of a heterogeneous model ensemble for per-frame polyp detection (sec. 2.3).
3. Tracking: Development of a strategy for leveraging the temporal information in endoscopic video sequences (sec. 2.4).
4. Post-processing: Development of a post-processing step to avoid systematic over-segmentation (sec. 2.5).

2.1. Datasets

The dataset provided by the EndoCV2022 polyp segmentation sub-challenge [2, 3, 4] consists of 46 sequences of varied length, totalling 3290 image frames and their corresponding polyp segmentation masks. Furthermore, we identified four public polyp datasets, namely CVC-ColonDB [5] (segmentation), CVC-ClinicDB [6] (segmentation), ETIS-Larib [7] (segmentation) and CVC-ClinicVideoDB [8, 9] (detection). We converted segmentation challenge datasets to detection datasets by computing the tightest possible bounding box for the provided segmentation masks.

4th International Workshop and Challenge on Computer Vision in Endoscopy (EndoCV2022) in conjunction with the 19th IEEE International Symposium on Biomedical Imaging ISBI2022, March 28th, 2022, IC Royal Bengal, Kolkata, India
m.elyamlahi@dkfz-heidelberg.de (A. Yamlahi)

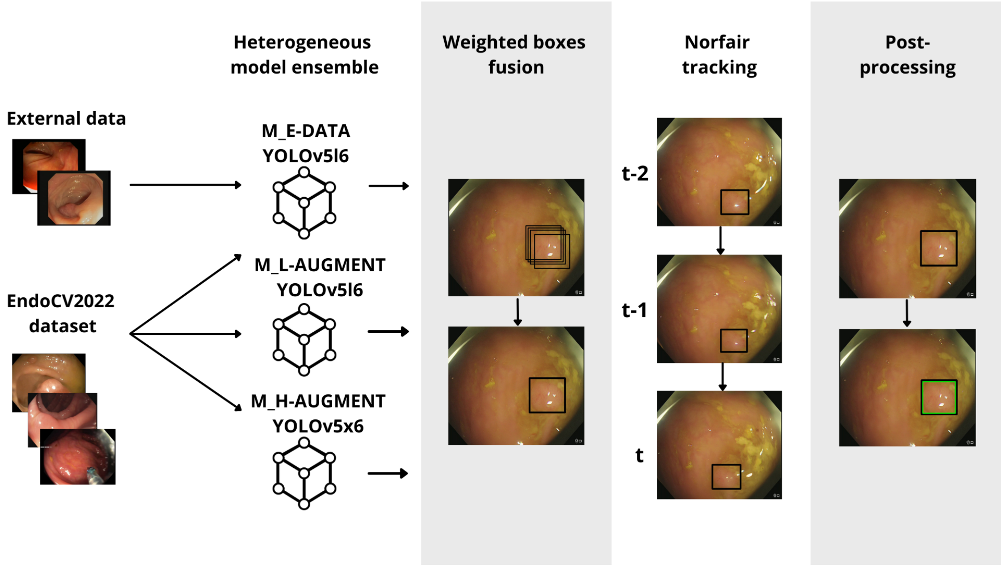

Figure 1: Method overview: A heterogeneous model ensemble comprises three YOLOv5 models, each trained with a different strategy based on external data and data augmentation. The output of the ensemble members is merged with the weighted boxes fusion algorithm and passed on to a Norfair tracking-based algorithm. The final output bounding boxes are reduced in size.

2.2. Validation strategy

We split the 46 EndoCV2022 sequences into four folds using the GroupK-Fold algorithm from the sklearn library [10]. The split was based on the sequence ID in order to prevent leakage, and we stratified based on the sequence length to have a balanced number of frames per fold. We used the validation performance on the left out fold for selecting our model checkpoints in the ensemble. For a faster training and inference time, we used two out of the four folds for both training and validation. As a validation metric we used the mean average Precision (mAP) over the Intersection Over Union (IoU) threshold range between 0.5 and 0.95 (mAP@$[.5 : .95]$) as proposed by the organizers of the EndoCV2022 challenge.

2.3. Heterogenous model ensemble

We based our method upon YOLOv5x6 and YOLOv5l6 [11] as our detection models as we identified them as being a good compromise between accuracy and speed. To build our heterogeneous model ensemble, we tested different augmentation strategies aimed to improve the model generalization. We group our trained models in three categories, based upon model architecture and the training data. Each category comprises models trained upon two of the folds.

1. Model M_H-AUGMENT: YOLOv5x6 trained with images of size 768x768 with heavy image augmentations. The augmentations applied on the first fold comprise mosaic and mixup augmentations with a probability of 1.0 and 0.5, respectively, Hue-Saturation-Value (HSV) channel enhancements with a maximal magnitude of 0.2 each, horizontal flip, vertical flip and Copy-Paste augmentation with a probability of 0.5 each as well as a final rotation of up to 25 degrees. We will refer to this combination of augmentations as the "default augmentation pipeline". The augmentations on the second fold are almost identical, setting the HSV enhancement to more deliberate magnitudes 0.015, 0.7 and 0.4. In addition, the Copy-Paste augmentation was omitted.
2. Model M_L-AUGMENT: YOLOv5l6 trained with images of size 768x768 with light image augmentations. On the first fold, we drastically reduced the default augmentation pipeline: Omitting mixup, vertical flipping, rotation as well as Copy-Paste transform. Furthermore, we used the deliberate HSV magnitudes again. The augmentations on the second fold are closer to the default augmentation pipeline in terms of augmentations used. The single difference is to drastically reduce the magnitude of mosaic from 1.0 to 0.2. We aimed to bring diversity to the ensemble by including both models trained with light and heavy augmentations.
3. Model M_E-DATA: YOLOv5l6 trained with the

resized external data described in sec. 2.1. The first fold was trained with images of size 768x768 while the second fold with images of size 512x512. With the enriched training data, comprising additional 13,251 frames from additional data sources, this model specifically targeted generalizability to new settings.

All models were initiated with the standard-pretrained weights on the COCO dataset [12] and trained for 20 epochs. In cases of slow convergence, the training period was extended up to 40 epochs using a Stochastic Gradient Descent optimizer with momentum set to 0.937, a learning rate of 0.01 and complete intersection over union (CIoU) loss [13] as the loss function. We saved the weights on the epoch with the best mAP score based on the validation data for the current fold. The predicted bounding boxes of each model were post-processed using the Non-Maximum-Suppression (NMS) algorithm with an IoU threshold of 0.5, to pick one bounding box out of many overlapping entities. To ensemble the bounding box predictions of multiple models, we used the weighted boxes fusion (WBF) algorithm [14] with an IoU threshold of 0.5 and the skip box threshold of 0.02. All models were weighted equally.

2.4. Tracking

In order to leverage the temporal information in the video sequences, we added a second stage tracker on top of the detection model to track the bounding boxes. We used Norfair [15], a multiple-object tracker, to track the polyps by calculating the Euclidean distance between the already tracked polyp and the prediction provided by the detection model. The tracker only considers bounding boxes within a distance of a set threshold to each other. On a 1080x1920 image, we experimented with several distance thresholds in the range 50px-250px, minimum hit inertia values in the range of 3-30, maximum hit inertia values in the range 6-50, and initialization delay values in the range of 1-20. The best results were obtained with a distance threshold of 50px, minimum hit inertia value of 10, maximum inertia value of 25 and an initialization delay of 10.

2.5. Post-processing

While bounding boxes are generated from the segmentation masks and are calculated to fit tightly around the polyp, the predictions by object detection models tend to cover more surface than the reference labels, which results in the inclusion of false-positive pixels inside the bounding box. To avoid this over-segmentation, we shrink the bounding boxes with a confidence score higher than 0.4 by 2% of their size.

3. Results

In the interest of a shorter inference time, we only considered the models M_L-AUGMENT, M_H-AUGMENT and M_E-DATA trained over two folds out of the original four folds for evaluation and inference. Table 1 compares the results of the three models averaged over two folds and validated on their respective validation fold. We inferred the models with the following hyperparameters configuration: a confidence threshold of 0.01 and image size of 768x768 for the models without external data and image size of 512x512 for the models with external data. Our best single model M_L-AUGMENT obtained an mAP@[$.5 : .95$] score of 0.42 on the validation set. With the ensemble of three different models trained with post-processing, we obtained the best performance of 0.44 mAP@[$.5 : .95$] on the validation split thanks to the variation in model architectures and augmentations. Adding the bounding box tracking to the pipeline did not improve performance with respect to the entire area under the precision-recall curve, as measured by mAP. However, we observed improved F2 scores at relevant working points of the curve and leave an in-depth analysis of potential benefits to future research.

Table 1
Mean Average Precision (mAP) scores of the selected models and the ensemble with tracking and post-processing.

Model	AP	AP50	AP75
M-L_AUGMENT	0.42	0.55	0.46
M-H_AUGMENT	0.37	0.56	0.45
M-E_DATA	0.33	0.49	0.37
Ensemble	0.43	0.59	0.49
Ensemble + tracking	0.42	0.59	0.49
Ensemble+ post-processing	**0.44**	**0.60**	**0.50**

4. Conclusion

We presented a new approach to polyp detection in endoscopic video sequences that leverages a heterogeneous ensemble of YOLOv5 models to achieve generalization. According to our analyses, the biggest performance gains were obtained from application-specific augmentation strategies and the ensemble of different architectures. Future work should aim for generating substantial performance gains by incorporating temporal information.

5. Compliance with ethical standards

This work was conducted using public datasets of human subject data made available by [2, 3, 4, 5, 6, 7, 8, 9].

6. Acknowledgments

This project was supported by a Twinning Grant of the German Cancer Research Center (DKFZ) and the Robert Bosch Center for Tumor Diseases (RBCT). Part of this work was funded by Helmholtz Imaging (HI), a platform of the Helmholtz Incubator on Information and Data Science.

References

[1] F. A. Haggar, R. P. Boushey, Colorectal cancer epidemiology: incidence, mortality, survival, and risk factors, Clinics in colon and rectal surgery 22 (2009) 191–197.

[2] S. Ali, M. Dmitrieva, N. Ghatwary, S. Bano, G. Polat, A. Temizel, A. Krenzer, A. Hekalo, Y. B. Guo, B. Matuszewski, et al., 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.

[3] S. Ali, D. Jha, N. Ghatwary, S. Realdon, R. Cannizzaro, O. E. Salem, D. Lamarque, C. Daul, K. V. Anonsen, M. A. Riegler, et al., Polypgen: A multi-center polyp detection and segmentation dataset for generalisability assessment, arXiv preprint arXiv:2106.04463 (2021). doi:10.48550/arXiv.2106.04463.

[4] S. Ali, N. Ghatwary, D. Jha, E. Isik-Polat, G. Polat, C. Yang, W. Li, A. Galdran, M.-Á. G. Ballester, V. Thambawita, et al., Assessing generalisability of deep learning-based polyp detection and segmentation methods through a computer vision challenge, arXiv preprint arXiv:2202.12031 (2022). doi:10.48550/arXiv.2202.12031.

[5] J. Bernal, J. Sánchez, F. Vilarino, Towards automatic polyp detection with a polyp appearance model, Pattern Recognition 45 (2012) 3166–3182.

[6] J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, D. Gil, C. Rodríguez, F. Vilariño, Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians, Computerized Medical Imaging and Graphics 43 (2015) 99–111.

[7] J. Silva, A. Histace, O. Romain, X. Dray, B. Granado, Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer, International journal of computer assisted radiology and surgery 9 (2014) 283–293.

[8] Q. Angermann, J. Bernal, C. Sánchez-Montes, M. Hammami, G. Fernández-Esparrach, X. Dray, O. Romain, F. J. Sánchez, A. Histace, Towards real-time polyp detection in colonoscopy videos: Adapting still frame-based methodologies for video sequences analysis, in: Computer assisted and robotic endoscopy and clinical image-based procedures, Springer, 2017, pp. 29–41.

[9] J. Bernal, A. Histace, M. Masana, Q. Angermann, G., dray, x., and sanchez, j. polyp detection benchmark in colonoscopy videos using gtcreator: A novel fully configurable tool for easy and fast annotation of image databases, in: Proceedings of 32nd CARS Conference (Berlin, Germany, 2018.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[11] G. R. Jocher, ultralytics/yolov5, 2022. URL: https://github.com/ultralytics/yolov5.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: European conference on computer vision, Springer, 2014, pp. 740–755.

[13] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, D. Ren, Distance-iou loss: Faster and better learning for bounding box regression, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 12993–13000.

[14] R. Solovyev, W. Wang, T. Gabruseva, Weighted boxes fusion: Ensembling boxes from different object detection models, Image and Vision Computing 107 (2021) 104117.

[15] J. Alori, A. Descoins, KotaYuhara, David, B. Ríos, fatih, shafu, A. Castro, D. Huh, tryolabs/norfair: v0.4.0, 2022. URL: https://doi.org/10.5281/zenodo.6095785. doi:10.5281/zenodo.6095785.