ENSEMBLE MASK-AIDED R-CNN

Pengyi Zhang, Xiaoqiong Li, YunXin Zhong

Beijing Institute of Technology, Beijing, China

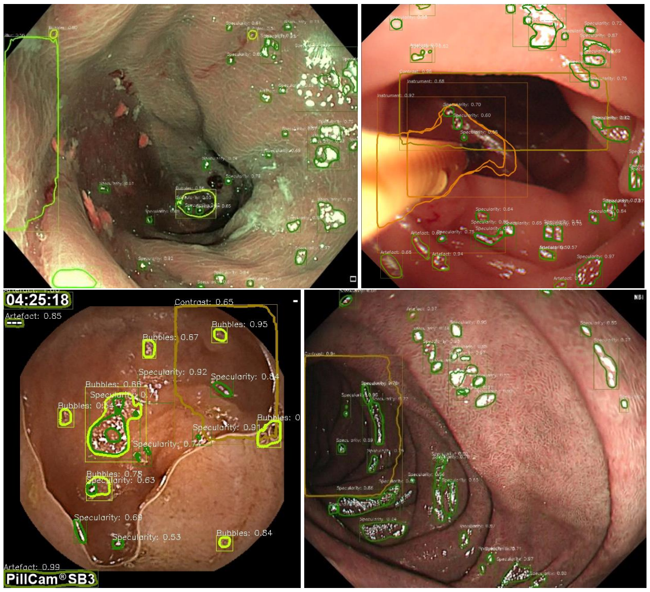
ABSTRACT

Recently the strategy of integrating instance mask prediction header into one-stage or two-stage object detector has been immensely popular for instance segmentation (e.g., RetinaMask or Mask R-CNN). This strategy notably improve the object detector at the meantime of learning to predict instance mask. In this paper, we introduce a Mask-aided R-CNN model with a flexible and multi-stage training protocol to address the problems of EAD2019 Challenge (a multi-class artefact detection in video endoscopy). The proposed training protocol aims to facilitate the implementation of this strategy for the detection task and segmentation task and to improve the detection and segmentation performance using pixel-level labeled samples with incomplete categories. This training protocol consists of three principal steps, of which the core part is augmenting the training set with soft pixel-level labels. The Mask-aided R-CNN is modified from Mask R-CNN by pruning its mask header to support training on pixel-level labeled samples with incomplete categories. We propose a simple yet effective ensemble method based on graph clique for object detectors to furtherly improve the detection performance. The ensemble method votes on graph cliques to fuse the detection results from different detectors. It produces robust detection results from different detectors. It produces robust detection results, which is quite important for clinical application. Extensive experiments on EAD2019 challenging dataset have demonstrated the effectiveness of our proposed ensemble Mask-aided R-CNN.As a result, we won the 1^{ST} place in detection task of EAD2019 Challenge.

Index Terms— Soft label, Ensemble, Graph clique, Mask-aided R-CNN

1. INTRODUCTION

Recently with the rapid development of medical imaging technology, the medical imaging diagnosis and treatment equipment and digital health records have been widely used in clinic. Among those medical imaging technologies, endoscopy is an important clinical procedure for early detection of cancers in hollow organs. However, the endoscopy video frames are easily corrupted with multiple artefacts (e.g., motion blur, specular reflections, bubbles etc.), thus increasing the difficulty of visual diagnosis. In order to retrieve high-quality endoscopic frame and facilitate the visual diagnosis, the algorithms of endoscopic frame restoration based on the priori knowledge of artefacts are generally used in existing endoscopy workflows. Therefore, identifying the types and the locations of those artefacts accurately is essential for high-quality endoscopic frame restoration and is crucial for realizing reliable computer assisted endoscopy tools for improved patient care. However, the methods for identifying artefacts in existing endoscopy workflows support only one single artefact type in an endoscopic frame, which generally contains multiple artefacts as shown in **Figure 1**. Moreover, different types of artefacts unequally contaminate the frame, thus requiring specific restoration algorithms for specific types of artefacts. Therefore, it is an urgent problem to develop accurate detection algorithms for multi-class artefact detection task.



]

Fig. 1. Illustration of detection and segmentation results of proposed ensemble Mask-aided R-CNN.

Driven by the growth of computing power (e.g., Graphical Processing Units and dedicated deep learning chips) and the availability of large labelled data sets (e.g., ImageNet [1] and COCO [2]), deep neural networks have been extensively studied due to their fast, scalable and end-to-end learning framework. In recent years, Convolution Neural Network (CNN) [3] models have achieved significant improvements compared with conventional shallow methods in image classification (e.g., ResNet [4] and DenseNet [5]), object detection (e.g., Faster R-CNN [6] and SSD [7]) and semantic segmentation (e.g., UNet [8] and Mask R-CNN [9]) etc. The advantages of CNN models, i.e. modular design and end-to-end learning architecture, enable existing CNN models to be easily used in complex problems by adding task-specific network branch. Recently the strategy of integrating instance mask prediction header into one-stage or two-stage object detector has been immensely popular for instance segmentation (e.g., RetinaMask [10] or Mask R-CNN [9]). This strategy notably improve the object detector at the meantime of learning to predict instance mask. In this paper, we aim at addressing the problems of multi-class endoscopic artefact detection by developing instance segmentation algorithm using this strategy in EAD2019 Challenge [11][12]. The EAD2019 Challenge provides two kinds of labelled samples, i.e. endoscopic frames with bounding box annotation for detection task and endoscopic frames with pixel-level annotations for segmentation task. The frames for segmentation task are contained in the frames for detection task, which means only part of endoscopic frames in detection task have pixel-level annotations.

We present ensemble Mask-aided R-CNN for multi-class endoscopic artefact detection with three highlights. First, we propose to integrate the detection task and segmentation task into an end-to-end framework of instance segmentation, i.e. Mask-aided R-CNN, which are able to take full advantage of all labelled samples to improve the performance of multi-class endoscopic artefact detection. Second, we design a flexible and multi-stage training protocol based on soft pixel-level annotations to train proposed Mask-aided R-CNN. The soft pixel-level annotations are firstly generated by initially trained Mask R-CNN models and furtherly refined by subsequently retrained models. The effectiveness of designed training protocol has been verified in training and improving Mask-aided R-CNN. Third, we propose a simple yet effective ensemble method based on graph clique for object detectors to furtherly improve the detection performance. Extensive experiments on EAD2019 challenging dataset have demonstrated the effectiveness of our proposed ensemble Mask-aided R-CNN. As a result, we won the 1^{ST} place in detection task of EAD2019 Challenge.

2. METHOD

2.1. Training Protocol of Mask-aided R-CNN

Adding a branch of mask header in one-stage or two-stage object detector is a common strategy to enable instance segmentation. The effectiveness of this strategy in improving both detection and segmentation performance has been witnessed in recent years (e.g., RetinaMask [10] and Mask R-CNN [9] illustrated in **Figure 2**). To take full advantage of this strategy, we introduce a Mask-aided R-CNN model with a flexible and multi-stage training protocol.

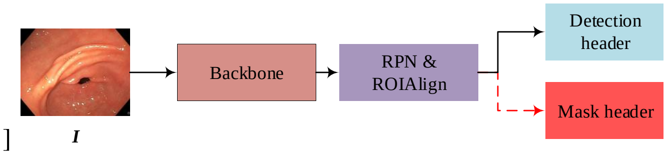


Fig. 2. Illustration of adding mask header to enable instance segmentation (Based on Faster R-CNN [6] and Mask R-CNN [9]).

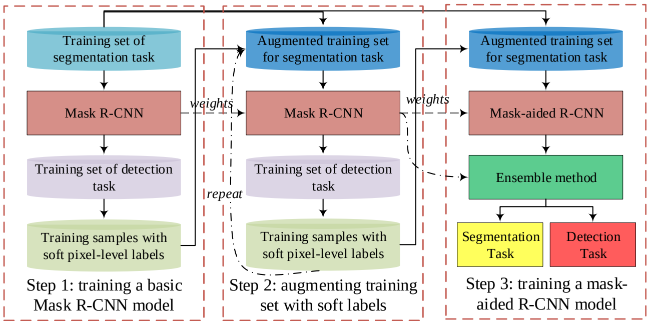

Training set of segmentation task
Mask R-CNN
Training set of detection task
Training samples with soft pixel-level labels
Step 1: training a basic Mask R-CNN model
weights
repeat
Augmented training set for segmentation task
Mask R-CNN
Training set of detection task
Training samples with soft pixel-level labels
Step 2: augmenting training set with soft labels
weights
Augmented training set for segmentation task
Mask-aided R-CNN
Ensemble method
Segmentation Task
Detection Task
Step 3: training a mask-aided R-CNN model

Fig. 3. Outline of proposed multi-stage training protocol.

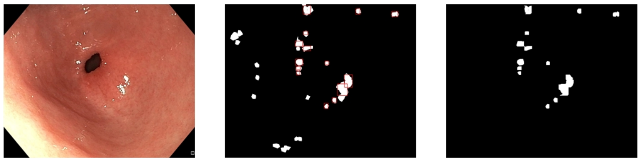
The outline of proposed multi-stage training protocol shown in **Figure 3** consists of three principal steps, of which the core part is augmenting the training set with soft pixel-level labels.

2.1.1. *Step 1: training a basic Mask R-CNN model for the segmentation task*

We first train a basic Mask R-CNN model on the training set of segmentation task to implement instance segmentation. In order to maintain consistency of semantic segmentation and object detection, the instance masks are bounded by bounding box annotations acquired from the training set of detection task. The process is illustrated in **Figure 4**.

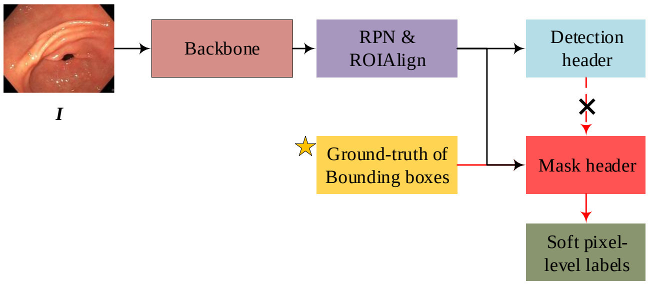
2.1.2. *Step 2: augmenting the training set of segmentation task with soft pixel-level labels*

The trained Mask R-CNN model is subsequently used to predict instance masks for the training samples of detection task that have no pixel-level labels. One thing that needs to be noted is that during the inference process the results of object detection are toughly replaced with the ground truth bounding boxes. It means that we enforce the mask prediction only for the ground truth instances. This trick shown in **Figure 5** aims to improve segmentation accuracy and to maintain consistency of semantic segmentation and object detection.



(a) Original image (b) Ground truth (c) Bounded mask

Fig. 4. Illustration of maintaining consistency of semantic segmentation and object detection. (a) is the original image ("00024.jpg"); (b) shows the ground truth mask of segmentation task ("Specularity"), where the bounding boxes marked in red are ground truth of detection task. (c) is the bounded mask used to train a Mask R-CNN model.



]

Fig. 5. Illustration of retrieving soft pixel-level labels. We perform mask prediction only on the ground-truth bounding boxes to maintain the consistency of semantic segmentation and object detection.

These predicted instance masks, called soft pixel-level labels, are assigned to the corresponding training samples. These training samples with soft pixel-level labels are furtherly added to the training set of segmentation task. We retrain the Mask R-CNN model on the augmented training set of segmentation task. Subsequently, the soft pixel-level labels are refined with the new instance masks predicted by the retrained Mask R-CNN model. This step might be performed multiple times for higher segmentation accuracy. The final augmented training set will be used in next step, while the final retrained Mask R-CNN model can be used by the ensemble module.

2.1.3. Step 3: training a Mask-aided R-CNN model for detection and segmentation task

To take full advantage of all available training samples and the strategy of boosting object detection by adding mask prediction branch, we generate soft pixel-level labels for training samples with no pixel-level annotations through the first two steps. In this step, we train multiple Mask-aided R-CNN models with different backbone networks on the final augmented training set. The Mask-aided R-CNN model supporting to be trained on pixel-level labeled samples with incomplete categories is detailed later in next section. These trained Mask-aided R-CNN models will be used by the ensemble module to furtherly improve the detection performance.

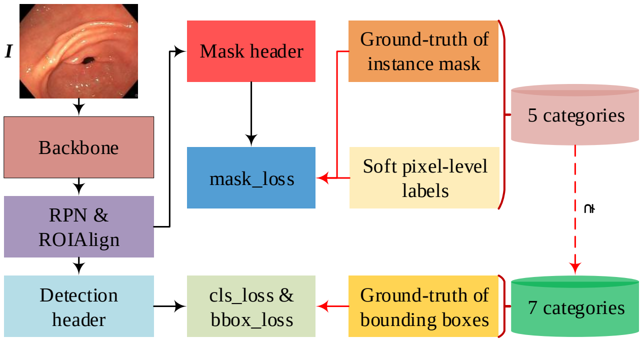


Fig. 6. Proposed Mask-aided R-CNN. We compute the mask loss only for the five segmentation categories.

2.2. Mask-aided R-CNN

The Mask-aided R-CNN shown in **Figure 6** is modified from Mask R-CNN by pruning its mask header to support training on pixel-level labeled samples with incomplete categories. In EAD2019 Challenge[11], the detection task has seven categories while the segmentation task has five categories, where the five segmentation categories are a subset of the seven detection categories. Therefore, the Mask-aided R-CNN model for EAD2019 Challenge is designed by following the two steps below: (1) Design a Mask R-CNN model with seven semantic categories; (2) Prune the neural units and connections related with the two extra categories in the mask header of this Mask R-CNN to get a mask header with five semantic categories. When training such a Mask-aided R-CNN model, we compute the mask loss only for the five segmentation categories. The remaining defaults in training process are kept unchanged.

2.3. Ensemble method

Ensemble strategy is commonly used to improve the performance in image classification tasks. In the detection task of EAD2019 challenge, we propose a simple yet effective ensemble method based on graph model for object detection tasks to furtherly improve the detection performance. Our proposed ensemble method is able to fuse the detection results from multiple object detectors by voting on one graph clique for the same object and mutually reinforcing each other among graph cliques.

2.3.1. Construction of Graph model

Given a single image I, C semantic categories and N object detectors, the detection result set can be formalized as $\{Det_n^c | n = 1, 2, \dots, N,\ c = 1, 2, \dots, C\}$. For convenience, we simply extract the detection results of a single category

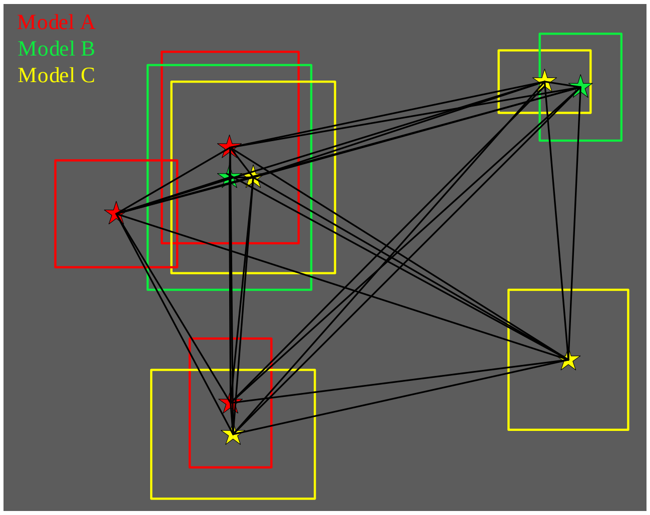


Fig. 7. The construction process of graph model for proposed ensemble method. Each rectangle denotes one detection and each color in the figure denotes one detector model.

$\{Det_n^c|n = 1, 2, \ldots, N\}$ to introduce and formalize our ensemble method (illustrated in **Figure 7**). Each detection Det_n^c consist of $\{uuid,\ score,\ bbox\}$, where $uuid$ denotes the universally unique identifier of the detector, $score$ denotes the confidence score of this detection and $bbox$ denotes the bounding box of this detection.

A weighted undirected graph $G^c(V, E)$ with dense connections can be established from the detections $\{Det_n^c|n = 1, 2, \ldots, N\}$, where V denotes the set of vertexes and E denotes the set of edges. Each vertex represents a single detection. The vertexes are densely connected with each other by edges. We assign a weight, i.e. the intersection over union (IOU) score of two detections, to the corresponding edge.

2.3.2. *Inference of Graph model*

We formulize the inference of established graph model as the maximum clique problem, which aims to maximize the sum of edge weights in each clique here. Several reasonable constraints are introduced to simplify the partition process. Post processing, e.g. non-maximum suppression (NMS), to remove redundant detections is commonly used in object detector. Therefore, the vertexes in one clique are required to be different in the $uuid$ attribute, which means removing the edges constructed by the same detector. Moreover, we introduce a threshold of IOU score to remove the edges with low weights, which means that the two detections with higher IOU score are more likely to be the same object.

After the simplification step, we design a greedy approach to solve this maximum clique problem iteratively. Initially, each vertex is adopted as a clique. We iteratively merge the two cliques, which has largest edge weight and different $uuid$ attributes of all the vertexes.

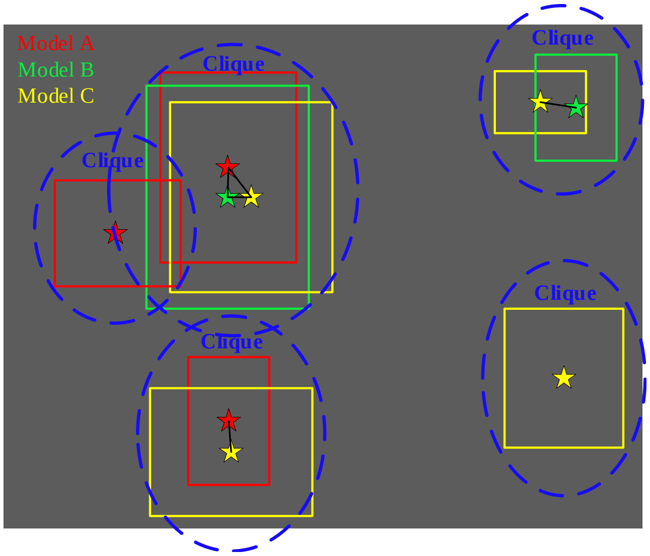


Fig. 8. The inference process of graph model for proposed ensemble method. Each clique denotes one detection result, which can be calculated by **Formula (1) and (2)**.

The last step is voting on partitioned cliques and therefore, each clique output a single detection by calculating its confidence score and bounding box. Given a clique $\{Det_k^c = \{uuid_k,\ score_k,\ bbox_k\}|k = 1, 2, \ldots, K\}$, the voting result $Det^c = \{0,\ score,\ bbox\}$ is formalized as follows:

$$score = 1 - \prod_k^K (1 - score_k) \tag{1}$$

$$bbox = \frac{\sum_k^K score_k \times bbox_k}{\sum_k^K score_k} \tag{2}$$

where K denotes the number of vertexes in this clique

3. EXPERIMENTS

Experiments on EAD2019 Challenge [1] are performed by following the proposed training protocol of Mask-aided R-CNN. We train our models on servers with two 1080Ti GPUs.

3.1. Experiments on training a basic Mask R-CNN model for the segmentation task

First, we generate the bounded mask for the released samples of segmentation task to enable instance segmentation. The released data (498 images with pixel-level labels in total) is split into training set (90%, 448 images) and validation set (10%, 50 images). Second, we train a Mask R-CNN model

[1] https://ead2019.grand-challenge.org

Table 1. The evaluation results of basic Mask R-CNN model tested on validation set.

Task	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Detection	25.8	50.4	22.4	12.0	34.6	23.8
Segmentation	23.3	44.6	20.3	7.7	28.4	26.5

with the backbone network of RseNet101 and feature pyramid network (FPN) [13] on this training samples. We perform two augmentation operations, i.e., random scaling and random horizontal flipping. The network is trained end-to-end using SGD with the momentum of 0.9 and weight decay of 0.0001. We train the model using mini-batches of size 2. We use an initial learning rate of 0.005 that is decayed by a factor of 10 at the iteration step of 24000 and 48000. The maximum training iteration is set as 72000.

The trained model is tested on the validation set and the evaluation results are shown in **Table 1** .

3.2. Experiments on augmenting the training set of segmentation task with soft pixel-level labeled samples

The trained Mask R-CNN is then used to generate soft pixel-level labels for training samples of detection task. We follow the trick detailed in **Chapter 2** to enforce the mask prediction only for the ground truth instances to maintain consistency of semantic segmentation and object detection and to improve segmentation accuracy. We evaluate the generated soft mask on validation set of segmentation task and the results are shown in **Table 2**. Compared with **Table 1**, the quality of predicted masks has been improved significantly, which verifies the effectiveness of proposed trick.

The second step of proposed training protocol is performed only once in this experiment. We generate soft mask for each released sample in detection task. These soft mask annotations, together with the released samples of detection task and the corresponding bounding box annotations, constitute the whole dataset for instance segmentation task.

3.3. Experiments on training the Mask-aided R-CNN models for detection task

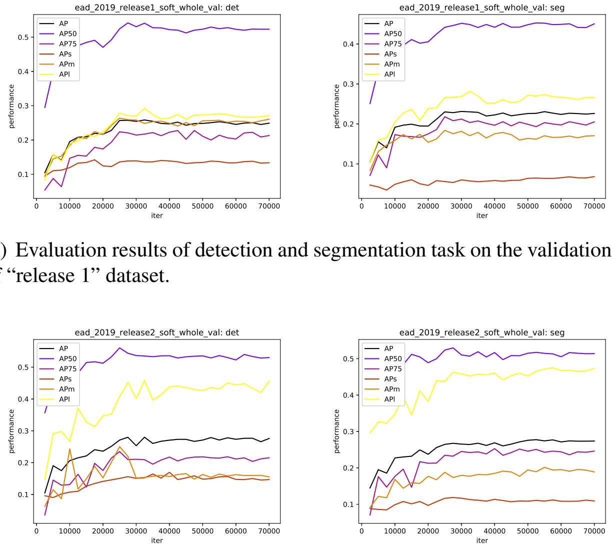
The whole dataset consists of two released datasets, of which the first released dataset contains 889 images and the second released dataset contains 1306 images. We split the whole dataset into one training set (90%, 800 images in the first released dataset and 90%, 1175 images in the second released dataset), a validation set of “release 1” (10%, 89 image in the first released dataset) and a validation set of “release 2” (10%, 131 images in the second released dataset). We successively train three Faster R-CNN models and three Mask-aided R-CNN models on the training set. The corresponding backbone networks of these models are RseNet50, RseNet50+FPN and RseNet101+FPN, respectively. The Faster R-CNN models are trained only with bounding box annotations, while the Mask-aided R-CNN are trained with both soft mask annotations and bounding box annotations. Here, we perform the data augmentation on the training set with random scaling, random horizontal flipping, random vertical flipping and random cropping on-the-fly. Each model is trained end-to-end using SGD with the momentum of 0.9. A weight decay factor of 0.0002 is adopted when training the models with a ResNet101+FPN backbone, while a weight decay factor of 0.0001 is adopted when training other models. We train each model using mini-batches of size 2. We use an initial learning rate of 0.005 that is decayed by a factor of 10 at the iteration step of 24000 and 48000. The maximum training iteration is set as 72000.

Table 2. The evaluation results of proposed trick to enforce the mask prediction only for the ground truth instances.

Task	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Detection	-	-	-	-	-	-
Segmentation	31.5	61.4	26.3	19.0	32.1	33.6

We evaluate the iteration snapshots of each model on the validation set of “release 1” and “release 2”. The average precision curves of each model are shown in **Figure 9, 10, 11, 12, 13, and 14**.

For quantitatively evaluation, we uniformly select two iteration snapshots (iteration of 40000 and iteration of 72000)



(a) Evaluation results of detection and segmentation task on the validation set of “release 1” dataset.

(b) Evaluation results of detection and segmentation task on the validation set of “release 2” dataset.

Fig. 9. Evaluation results of Mask-aided R-CNN with the backbone of ResNet50 on the detection and segmentation task.

Table 3. The evaluation results of three Faster R-CNN models, three Mask-aided R-CNN models and three ensemble models on validation set of "release 1" dataset.

Model	iter	AP	AP$_{50}$	AP$_{75}$	AP$_{S}$	AP$_{M}$	AP$_{L}$	AP$_{1}$	AP$_{10}$	AP$_{100}$	AR$_{S}$	AR$_{M}$	AR$_{L}$	EC
faster+ResNet50	40000	24.9	52.9	20.6	13.6	24.9	26.1	18.5	35.2	38.9	19.8	36.1	45.2	
	72000	25.4	52.9	21.9	14	24.9	26.4	19.1	35.3	38.7	19.9	36	44.8	29.8
mask-aided+ ResNet50	40000	24.7	52.2	22.3	13.9	24.9	26.4	19	34.1	38.3	19.9	35.9	48	
	72000	24.9	51.9	21.5	13.2	25.3	27.4	19.7	35.3	38.8	18.7	35.8	48.8	30
faster+ ResNet50+FPN	40000	25.4	52.8	21.6	14.4	23.3	27	19.2	34.7	39	21.1	34.1	45.9	
	72000	25.2	52	20.9	14.8	23.9	26	19.8	34.7	38.8	20.8	35.2	42.9	29.7
mask-aided +ResNet50+FPN	40000	25.4	52.3	21.9	13.9	21.5	27.2	18.6	35.5	40.3	21	36	45.9	
	72000	25.9	53.1	21.7	13.9	23.4	28.7	20.1	36.6	41	20.3	36.6	51.7	30.5
faster+ ResNet101+FPN	40000	26	52.1	22.9	13.6	23.4	27.2	20.2	35.8	39.9	20.6	36.1	48	
	72000	26.2	51.8	24.1	13.8	24.3	27	20.3	35.5	39.7	20.5	36.3	43.7	30.4
mask-aided+ ResNet101+FPN	40000	26.7	54.3	23.9	17.2	23.6	28.9	20.7	37	41.5	26.2	36.2	49.2	
	72000	26.5	52.7	24.9	14.1	24	28.4	21	36.9	40.6	20	36.2	46	31.5
faster ensemble		28.5	53.4	26.6	15.5	27.5	30	20.8	38.4	43.3	21.3	39.2	47.5	32.7
mask-aided ensemble		28.4	54.6	26.8	15	27.1	31	21.7	38	42.6	21.3	37.9	52.6	33.1
all ensemble		29.6	55.5	28	16.2	28.5	31.7	21.9	39.9	45.8	23.3	41	54.3	34.6

Table 4. The evaluation results of three Faster R-CNN models, three Mask-aided R-CNN models and three ensemble models on validation set of "release 2" dataset.

Model	iter	AP	AP$_{50}$	AP$_{75}$	AP$_{S}$	AP$_{M}$	AP$_{L}$	AP$_{1}$	AP$_{10}$	AP$_{100}$	AR$_{S}$	AR$_{M}$	AR$_{L}$	EC
faster+ResNet50	40000	27.4	56.7	25.3	16.9	24.7	37.6	28.3	40	42.8	25.1	33.3	50.1	
	72000	27.6	55.1	24.5	16.7	24.4	38.6	28.5	40	42.6	24.3	32.7	51.1	33.9
mask-aided+ ResNet50	40000	27.1	53.6	21.8	17	15.6	43.8	25.5	37.1	39.8	26.2	25.4	57.7	
	72000	27.3	52.4	21.2	14.9	16	44.2	25.7	37.9	40.7	22.3	25.7	58.7	32.4
faster+ ResNet50+FPN	40000	26.2	55.4	20.8	15	18	37	26.9	38.5	41.2	24.4	33.4	52.3	
	72000	24.2	52.7	19.9	14.9	14.7	36.8	23.1	34.6	37.1	22.8	24.8	49.2	31.0
mask-aided+ ResNet50+FPN	40000	24.7	54.7	21.2	15.8	17.8	35.2	23.4	36	39.1	24.8	28.3	52.3	
	72000	25	52.8	20.5	14.3	16.8	35.2	24.3	36.6	39.4	22.6	27.1	55.6	31.0
faster+ ResNet101+FPN	40000	27.8	57.6	22	13.6	21.6	40.9	27.2	39.7	42.3	21.9	32.9	55.7	
	72000	27	55.7	22.1	14	19.9	41.3	26.8	38.2	40.6	21.4	31.6	57.3	33.3
mask-aided+ ResNet101+FPN	40000	30.7	60.4	22.8	14.4	26.1	42.5	30	42.2	45.6	24.1	35.3	60.9	
	72000	28.4	60.5	20.9	13.3	21.8	41.6	27	39.7	42.5	22.4	31.7	57.6	35.1
faster ensemble		27.7	56.5	23.9	17.4	21.6	40.6	27.1	39.8	42.5	27	32.6	56.6	34.4
mask-aided ensemble		29.5	57.6	23.4	15.2	25.8	43.4	26.2	41.9	44.7	23	34.9	58.3	35.3
all ensemble		30.1	58	24.1	17.5	26.2	44.2	26	43.6	46.8	28.2	36.5	59.5	36.7

from each trained model and evaluate the models on the validation sets of "release 1" and "release 2". The evaluation results of average precision (AP) and average recall (AR) are shown in **Table 3** and **Table 4**. The average of AP and AR is adopted as the Evaluation Criterion (EC) score in the experiments. In **Table 3**, the EC scores of Mask-aided R-CNN models are consistently higher than the EC scores of Faster R-CNN. Specifically, the significance of the EC score difference between Mask-aided R-CNN and Faster R-CNN increases as the complexity of backbone network increases. It implicates that the generated soft pixel-level labels facilitate to train a deeper convolutional network, thus improving detection performance. The EC scores in **Table 4** also reveals a consistent implication.

3.4. Experiments on ensemble method

The two selected iteration snapshots of each model in **Section 3.3** are enrolled in the proposed ensemble method. In this experiment, we implement three ensemble models, involving ensemble of Faster R-CNN models, ensemble of Mask-aided R-CNN models and ensemble of all the Faster R-CNN models and Mask-aided R-CNN models. The threshold of IOU score

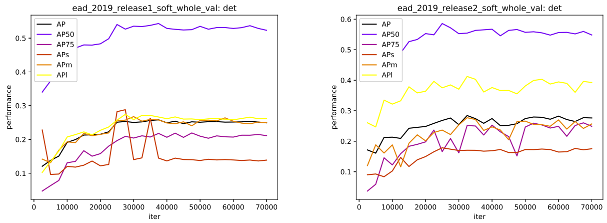
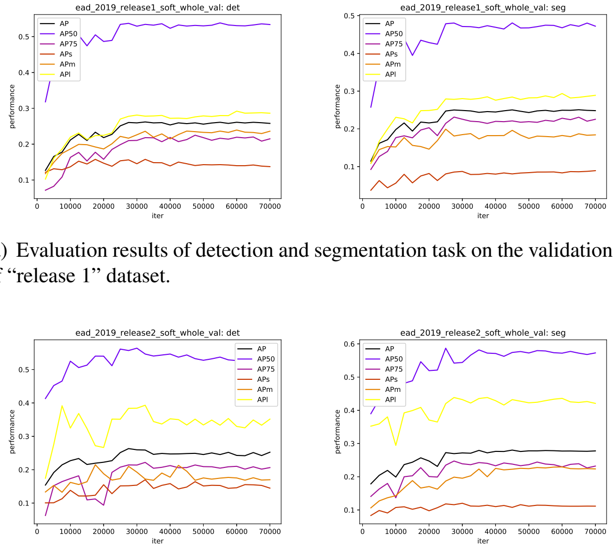


Fig. 10. Evaluation results of faster R-CNN with the backbone of ResNet50 on the detection task.



(a) Evaluation results of detection and segmentation task on the validation set of "release 1" dataset.

(b) Evaluation results of detection and segmentation task on the validation set of "release 2" dataset.

Fig. 11. Evaluation results of Mask-aided R-CNN with the backbone of ResNet50 and FPN on the detection and segmentation task.

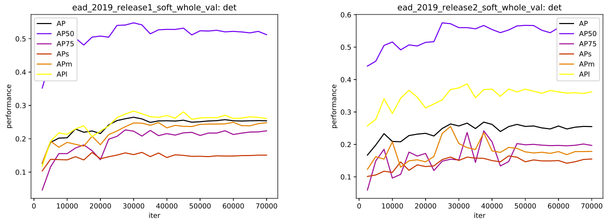
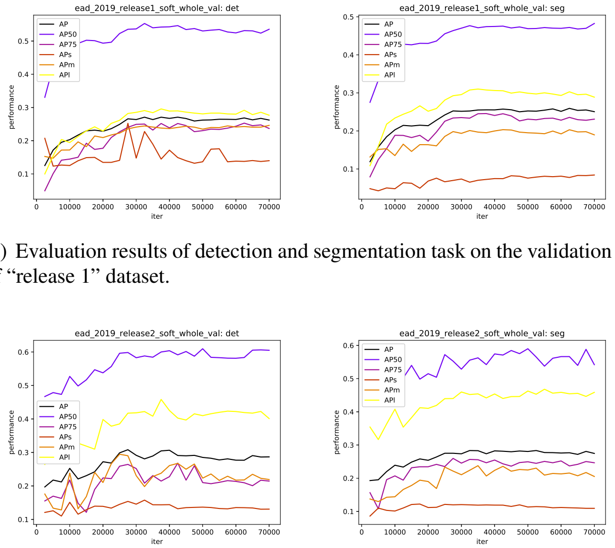


Fig. 12. Evaluation results of faster R-CNN with the backbone of ResNet50 and FPN on the detection task.

in ensemble method is consistently set as 0.4. We evaluate the three ensemble models on the validation sets. The evaluation results are shown in **Table 3** and **Table 4**.

The EC scores of ensemble Faster R-CNN models and Mask-aided R-CNN in **Table 3** and **Table 4** are significantly higher than the EC scores of corresponding single models.



(a) Evaluation results of detection and segmentation task on the validation set of "release 1" dataset.

(b) Evaluation results of detection and segmentation task on the validation set of "release 2" dataset.

Fig. 13. Evaluation results of Mask-aided R-CNN with the backbone of ResNet101 and FPN on the detection and segmentation task.

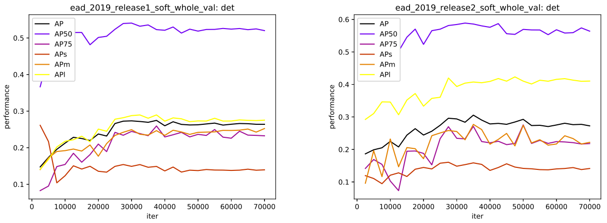


Fig. 14. Evaluation results of faster R-CNN with the backbone of ResNet101 and FPN on the detection task.

Furtherly, the ensemble of all the Faster R-CNN models and Mask-aided R-CNN models significantly improves the EC scores, which is adopted as the final model for the EAD2019 challenge. Such robust and significant improvements verify the effectiveness of proposed ensemble method.

4. CONCLUSION

In this paper, we introduce ensemble Mask-aided R-CNN with a flexible and multi-stage training protocol for the detection task and segmentation task of EAD2019 Challenge. Numerous experiments have demonstrated the effectiveness of our work. More specifically, Mask-aided strategy using soft pixel-level labels of incomplete categories facilitates to train a deeper convolutional network and to improve detection performance. The proposed ensemble method is able to fuse detection results from different detectors and furtherly improve detection performance with no training cost. Certain parts of proposed method remain to be furtherly explored, such as how to furtherly improve the segmentation performance with soft pixel-level labels.

5. REFERENCES

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[2] Tsung Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollr, and C. Lawrence Zitnick, "Microsoft coco: Common objects in context," 2014.

[3] Y Lecun, Y Bengio, and G Hinton, "Deep learning.," *Nature*, vol. 521, no. 7553, pp. 436, 2015.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[5] Li Da, Li Lin, and Li Xiang, "Classification of remote sensing images based on densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[6] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[7] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg, "Ssd: Single shot multibox detector," in *European Conference on Computer Vision*, 2016.

[8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*, 2015.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[10] Cheng-Yang Fu, Mykhailo Shvets, and Alexander C Berg, "Retinamask: Learning to predict masks improves state-of-the-art single-shot detection for free," *arXiv preprint arXiv:1901.03353*, 2019.

[11] Sharib Ali, Felix Zhou, Christian Daul, Barbara Braden, Adam Bailey, Stefano Realdon, James East, Georges Wagnires, Victor Loschenov, Enrico Grisan, Walter Blondel, and Jens Rittscher, "Endoscopy artifact detection (EAD 2019) challenge dataset," *CoRR*, vol. abs/1905.03209, 2019.

[12] 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.

[13] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125.