ENSEMBLE METHOD FOR PANOPTIC SEGMENTATION

Darwin Bautista, Jose Marie Mendoza, Rowel Atienza

Electrical and Electronics Engineering Institute
University of the Philippines
{darwin.bautista,jose.marie.mendoza,rowel}@eee.upd.edu.ph

ABSTRACT

In this work, we used an ensemble model for the COCO 2018 Panoptic Segmentation Challenge. For the instance segmentation task, we fine tuned a pretrained Mask-RCNN [1] network with the COCO panoptic annotations while we trained from scratch a DeepLabv3+ [2] model for the semantic segmentation task. The two-channel output is created by combining the output of the said models, prioritizing the instance masks over the semantic masks. The panoptic quality of the final output reached 36.0 on the COCO test-dev2017 subset.

***Index Terms*—** Panoptic Segmentation, Mask-RCNN, DeepLabv3+, two-channel output

I. PANOPTIC SEGMENTATION

The task of panoptic segmentation [3] can be roughly divided into two subtasks: semantic segmentation of *stuff*, and instance segmentation of *things*. However, due to the task definition of panoptic segmentation (e.g. no overlaps), we can not use as-is the outputs of individual state-of-the-art segmentation models.

1. Instance Segmentation of *things*

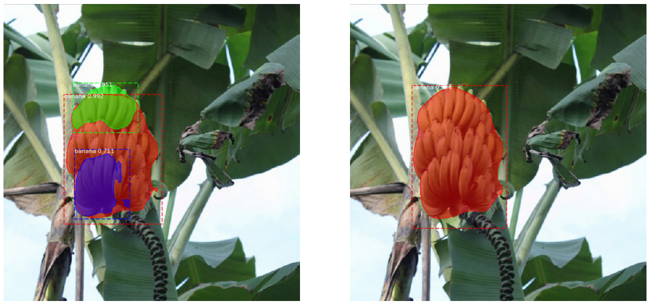
For the instance segmentation task, we used Mask R-CNN [1], particularly the implementation and COCO-trained weights from Matterport [4]. This model can be readily used in producing the instance segmentations for the panoptic task save for one problem: overlapping instance masks.

Mask R-CNN predicts instance masks per region-of-interest (ROI). ROIs could overlap with each other which means that the predicted masks could also overlap. One way to resolve the overlaps is to reason about the depth ordering of the scene. That is, order the masks by depth (background first), then *paint* each one according to that order, overwriting pixels of previous masks in case of an overlap. However, Mask R-CNN doesn't predict depth ordering. One technique we found effective is sorting the masks by size (largest first). This is based on two intuitions: that larger masks correspond to thing classes which are normally in the *background*; and that smaller masks contain more information than larger masks (inspired by information theory). Empirically, we found that this method increases the *Recognition Quality* (**RQ**) by about 1.3 points on the COCO *val2017* subset (Table 1).

Matterport's original Mask R-CNN implementation performs poorly on the panoptic task, as seen on Table 1. To increase the model's performance, we experimented by tuning its hyperparameters. We found that the default Non-Max Suppression (NMS) threshold on both the initial RPN proposals (0.7) and the final detection ROIs (0.3) were greatly affecting the results. In Figure 1, the original model predicts three instances of *banana* (because of the high NMS thresholds) instead of just one, as in the output of our tuned model. These extra predictions penalizes the **RQ** due to more segments being classified as False Positives (FP).

Table 1: Mask R-CNN evaluation results on the *val2017* subset for *thing* classes. The *baseline* corresponds to the baseline results shown in the challenge leaderboard, *ours* corresponds to our final instance segmentation model, *ours (unsorted)* is our model without mask sorting, and *Matterport* corresponds to the base model with default hyperparameters

	PQ	SQ	RQ
baseline	46.2	80.2	56.2
ours	44.8	78.8	55.4
ours (unsorted)	43.6	78.5	54.1
Matterport	34.3	77.5	43.0



(a) Matterport

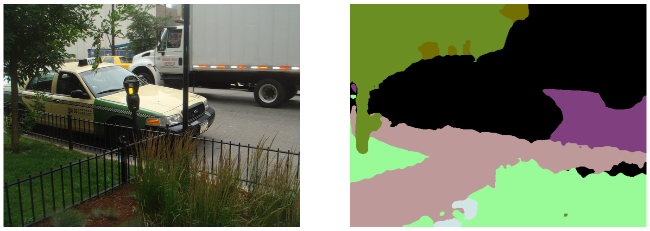
(b) Ours

Figure 1: Effect of the NMS threshold on the Mask R-CNN output. The high NMS threshold of the base model produces three predictions (left) for a single object instead of just one (right).

In our final model, we set both the RPN and final detection NMS thresholds to 0.2. This immediately increased the **RQ** of the base model by 9.1. Then we fine-tuned the model for just 1 epoch (due to time and hardware constraints) with the new NMS thresholds in place, which further increased **RQ** by 2.0.

2. Semantic Segmentation of *stuff*

An encoder-decoder structure of the DeepLabv3 is used, which is DeepLabv3+ [2], to generate the semantic masks. Given that *stuff*



(a) Sample Image **(b)** Predicted Semantic Mask

Figure 2: Sample prediction of the DeepLabv3+ model on relabeled semantic masks

classes are amorphous, ground truth semantic masks for the training data is re-labeled such that things class are placed in the void category so that our trained model will focus only on getting the shape of the *stuff* classes. A prediction example is illustrated in Figure 2b with the input shown in Figure 2a. It can be seen that both vehicles, truck and car, are identified as void class. Since the combination of masks prioritizes instance segments, as described on the next section, things could easily be superimposed on semantic masks while retaining definite edges.

3. Combining Masks

To get the final panoptic segmentation, we combined the instance segmentation and the semantic segmentation outputs described in the previous subsections. We started with the semantic segmentation output and superimposed the instance segmentations on it. We gave a higher priority to the instance segmentations, similar to the heuristic approach on [3], because these are predicted over smaller ROIs (which translates to higher mask accuracy) vs the whole image semantic predictions. Superimposing the instance segmentations on top of the semantic segmentation also refines the predictions for the *stuff* classes, increasing overall accuracy as seen in Table 3, versus the individual model results shown in Table 2.

Table 2: Panoptic Quality (PQ), Segmentation Quality (SQ) and Recognition Quality (RQ) on Things and Stuff separately on MS COCO validation dataset.

	Things			Stuff		
	PQ	SQ	RQ	PQ	SQ	RQ
Mask-RCNN	44.8	78.8	55.4	0	0	0
DeepLabv3+	0	0	0	20.9	70.8	28.1

Table 3: Panoptic Quality (PQ) combined Mask RCNN and DeepLabv3+ on MS COCO dataset including separate PQ on things and stuff.

	COCO val			COCO test-dev		
	PQ	SQ	RQ	PQ	SQ	RQ
All	35.6	75.8	44.6	36.0	76.7	44.9
Things	43.7	78.5	54.1	44.1	80.0	54.5
Stuff	23.6	71.6	30.2	23.6	71.6	30.3

II. ACKNOWLEDGMENT

This work was funded by CHED-PCARI Project IIID-2016-005.

III. REFERENCES

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2980–2988.

[2] L.-c. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," 2018. [Online]. Available: https://arxiv.org/abs/1802.02611

[3] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic Segmentation," 2018. [Online]. Available: http://arxiv.org/abs/1801.00868

[4] W. Abdulla, "Mask r-cnn for object detection and instance segmentation on keras and tensorflow," https://github.com/matterport/Mask_RCNN, 2017.