Amodal Instance Segmentation

Xiangyi Zhang
Supervisior: Xuming He
Shanghaitech University

Abstract

Common visual recognition task such as object detection, semantic segmentation are reaching high manurity. We now consider a different task of amodal segmentation which predicts the complete region of each object instance. Amodal segmentation plays an indispensable role in scene understanding, depth estimation, etc. We propose a novel part-based model to infer amodal masks from a single RGB image, which then allows for an interative refinement of mask propopsals generated by a state-of-the-art Mask R-CNN variant finetuned for amodal instance segmentation. We evalute our results on the publicly available COCO amodal dataset.

1. Introduction

Recently, visual recognition tasks such as image classfication[29, 9], object detection[6, 24], edge detection[1], and semantic segmentation[25, 19, 18], have witnessed dramatic progress. While all of these progresses are in visible field. We consider one challenge of predicting the invisible part. We take our inspiration from the study of human visiual system. A remarkable advantage of human vision is the ease of interpolating information of the invisible part, which in the end tranforms correct information to our brain and makes resonable decision. A particularly prominent example of this, and one on which we focus, is amodal instance segmentation.

Amodal instance segmentation, the ability of perceiving the whole of an object when it is partially occluded, just like the basic form of mental vision system human has, which readily perceive partially occluded objects and guess at their true shape.

There are three critical challenges to amodal instance segmentation. First, for occlusion, it is hard to predict scope of an incomplete object due to missing information and lack of prior knowledge. Secondly, there exists large-scale variance on occlusion patterns. Thirdly, labeling such variant dataset is challenging and expensive.

Based on these challenges, for occlusion, we adopt part-based representation instead of enumerating all occlusion patterns. We construct a refenrence set as a external memory bank which has complete maskes, providing the prior knowledge. In the meantime, the reference set without any amodal annotation also relieves the labeling burden.

2. Related Work

Instance segmentation. Recent years witnessed a rapid progress of Convolutional Neural Network (CNN) based instance segmentation algorithms over conventional methods such as [10]. Some of the most prominent examples include SDS [7, 8, 12], CFM [3], MNC [4] and FCIS [15]. These methods typically start with proposing a large pool of image regions in the form of either boxes or superpixels, and then make refinement to produce final segmentations based on image features from deep neural networks. In contrast, there has also been work that does not rely on object proposals. For example, [2, 16, 26] attempt to recover instance labels from semantic segmentation results. Other work has attempted to segment instances sequentially with a recurrent neural network [23]. Most relevant to ours are some of the recent work proposed for amodal perception of objects. Kar et al. [11] propose to learn categoryspecific object size distributions for amodal bounding box prediction. Li et al.[13] proposes to predict amodal instance segmentations using the method from [12] with randomly overlaid occluders. In additioin, Ehasni et al. [5] propose a model for generating the appearance of the invisible parts of an object with a generative adversarial network. Of particular interest is the work from Zhu et al. [22] that creates the largest publicly available amodal instance segmentation dataset based on a subset of the COCO dataset [17]. They established a solid baseline for class-agnostic amodal instance mask prediction. Their methods were built based on successful prior work including

DeepMask [20] and SharpMask [21] for class-agonstic amodal segmentation. Although our method makes use of their dataset, we address the problem of class-specific amodal instance segmentation. We believe that it is essential to use strong top-down shape priors to guide the refinement process of bottom-up segmentation proposals. To our knowledge, our work is the first to consider the problem of amodal instance segmentation with both top-down and bottom-up visual cues.

Amodal Instance Segemntation. Recent works of instance segmentation are mainly on the visible field. Amodal segmentation focus on the invisible part. Research on amodal instance segmentation or semantic amodal segmentation has just started to emerge. Li and Malik [5] were the first to provide a method for amodal instance segmentation. In [14], Zhu et al. provide a new and pioneering dataset COCO amodal for amodal instance segmentation based on images from the original COCO [17] dataset.

Deep Voting. DeepVoting detects semantic parts of an object proposed by Zhang et al. They propose that all models should be trained without seeing occlusion while being able to transfer the learned knowledge to deal with occlusion.[27] proposed a voting mechansim that combiens multiple laocal visual cues to detect semantic parts. The semantic meaning part s are still be detected even though some visual cues are missing due to occlusion. DeepVoting incroporates the robustness shown by [27] into a deep network, and it adds two layers after the intermediate features of deep network. The first layer extracts the evidence of local visual cues, and the second layer perdorms a voting mechanism by utilizing the saptial relationship between visual cues and semantic parts.

3. Model Setting

The model setting preparing for the next two key branches contains pairset and rescaling.

3.1. Pairset Setting

We decide to take proposal data from Mask R-CNN as query set method, which is image patch I_q, size $W \times H$. And We adopt reference set as external memory bank I_r, size $W \times H$. It plays the role of amodal annotation and amodal variation. Then, we take query set and refenrence set as pairset data of input image patches(I_q, I_r).

3.1.1 Reference Set

Reference set is composed of a set of paired instance image patches I_r and corresponding mask patches M_r cropped by ground truth bounding box. Reference dataset $D_r = \{I_r^{gt}, M_r^{gt}\}_{i=1}^{N_r}$, where N_r is the size of refenrence set.

3.1.2 Query Set

Query set is composed of a set of paired instance image patches I_q and corresponding mask patches M_q cropped by proposal bounding box, and mask patches M_q^{gt} cropped by instance's ground truth bounding box. Query dataset $D_q = \{I_q^{prop}, M_q^{prop}, M_q^{gt}\}_{i=1}^{N_q}$, where N_q is the size of query set.

3.2. ScaleNet

ScaleNet takes image patches I_q or I_r as input and predicts object scale $\hat{s}$, which is the ratio of the object bounding box short edge to the fixed long edge L. Its ground truth scale is denoted by $s = \frac{min(H^{gt}, W^{gt})}{L}$, where we set $L = 224$ pixels.

$$\hat{s} = ScaleNet(I^{prop}) \tag{1}$$

For query set data, after training of scalenet, the predicted ratio $\hat{s}$ is used to normalize the object to the desired size, which is, its short edge contains L pixels.

$$\begin{aligned} I = Rescale(I^{prop}; \hat{s}) \in \mathbb{R}^{H \times W}, \\ where\ H = \frac{H_p}{\hat{s}}, W = \frac{W_p}{\hat{s}} \end{aligned} \tag{2}$$

For reference set, directly use its ground truth scale s for rescaling and denoted as $\tilde{D}_{ref} = \{I_r^i, M_r^i\}_{i=1}^{N_r}$. For query , we use predicted scale $\hat{s}$ for rescaling and denoted as $\tilde{D}_q = \{I_q^i, M_q^i\}_{i=1}^{N_q}$.

After rescaling, we use threshold τ_{pos} and τ_{neg} to form positive and negtive pairs. For each query-ref pair, if their amodal ground truthh mask's overlap $\text{IoU}(M_r, M_q) \in (\tau_{pos}, \tau_{neg})$, we will discard it. For reamined qualified pairs, we assign classfication label by

$$y = \begin{cases} 1 & if \quad IoU(M_r, M_q) \geq \tau_{pos} \\ 0 & if \quad IoU(M_r, M_q) \leq \tau_{neg} \end{cases}$$

The goal of ScaleNet is to put query and reference data into the same scale size.

4. Model

We design a mask transfer pipeline shown in Figure 1. The key idea is to find the most similar complete mask. Then go through the part-based voting branch to find the correct position for mask transfer. We propose a system to vote a heatmap which is the probable center of query and reference data, then find the most similar reference mask center-aligned to query and get the prediction of query's complete mask.

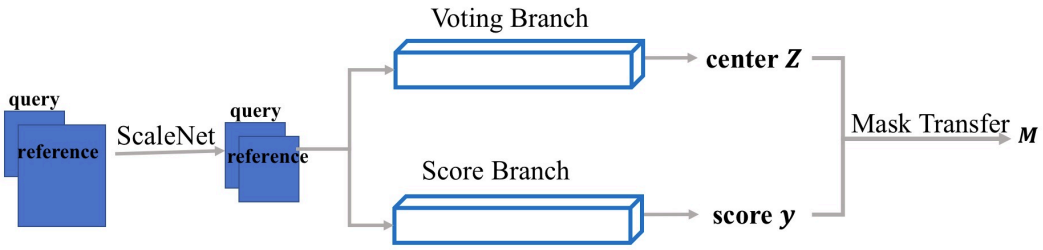


Figure 1. The mask transfer pipeline is to find the most similar complete mask for the correct position for mask transfer. We utilize voting and score these two branches.

4.1. Voting Branch

In voting branch, first we take query and reference image patches as input pairset(I_q, I_r), and go through a voting layer to vote the center of an object instance in feature map space[28, 30], shown in Figure 2

4.1.1 Visual Concept layer

This layer outputs response maps, which can be described as follows:

$$Y = f_1(X; w1, \overline{w}) \tag{3}$$

where X is feature map of image patch, parameter w_1 is the convolutional weight parameter for visual concept, size $1 \times 1 \times F$, and $\overline{w}$ is the convolutional parameter for visual concept layer, which measures how important of each visual concept.

4.1.2 Voting layer

Take response maps Y as input, this layer outputs object's voting heatmaps Z, whcih is the predicted center while ground truth is Z^*.

$$Z = f_2(Y; w_2) \tag{4}$$

Here Y consists of a pairset (Y_q, Y_r), since query and reference set they all get through the VC layer and generate the response maps. We utilize Y_q as filter to convolute with Y_r, which pads with query size. So the $H'_r = H_r + 2 \times H_q$. We use MSE loss between Z and Z^*. w_2 is the weight parameter for voting layer, size $k_2 \times k_2 \times F$.

$$Loss = MSE(Z, Z^*) \tag{5}$$

4.2. Score Branch

Based on voting branch, our goal in score branch is to find the most similar reference to query. So we use similarity score to measure how similar between current query image and reference image. If we choose the most similar reference mask, and center aligned to query mask's center, we can get amodal mask.

Score branch is similar to vote branch but add one classfication which represented in Figure 3. Here we use *Binary cross entropy* loss for mask similarity score, y^* is ground truth of score. *Mean Square Error* loss for regularization and coefficient is ε.

$$Loss = BCE(y, y^*) + \varepsilon * MSE(Z', Z^*) \tag{6}$$

4.3. Mask Transfer

We use a non-parametric way to transfer reference masks to query mask. We choose score top1 mask from score branch and center-aligned to the query incomplete mask from voting branch, which finally get complete mask.

5. Experiment

We perform a comparision of Mask R-CNN to the state of the art with our method on COCO amodal dataset. We report the standard COCO metrics including AP(averaged over IoU thresholds) evaluting mask IoU.

5.1. Dataset

Our dataset contains query dataset and reference set: COCO amodal dataset and Cityscape dataset in Table 1. And our method now only applies on car category.

Annotation		
Dataset	coco amodal	cityscape
Annotation	only car label	only car label
image	train:228 val: 144	320
categories	1: car	1: car

Table 1. We train our baseline of 228 train image in COCO amodal dataset, and 320 train image chose from Cityscape dataset as our refenrence set. And 144 val image from COCO amodal dataset.

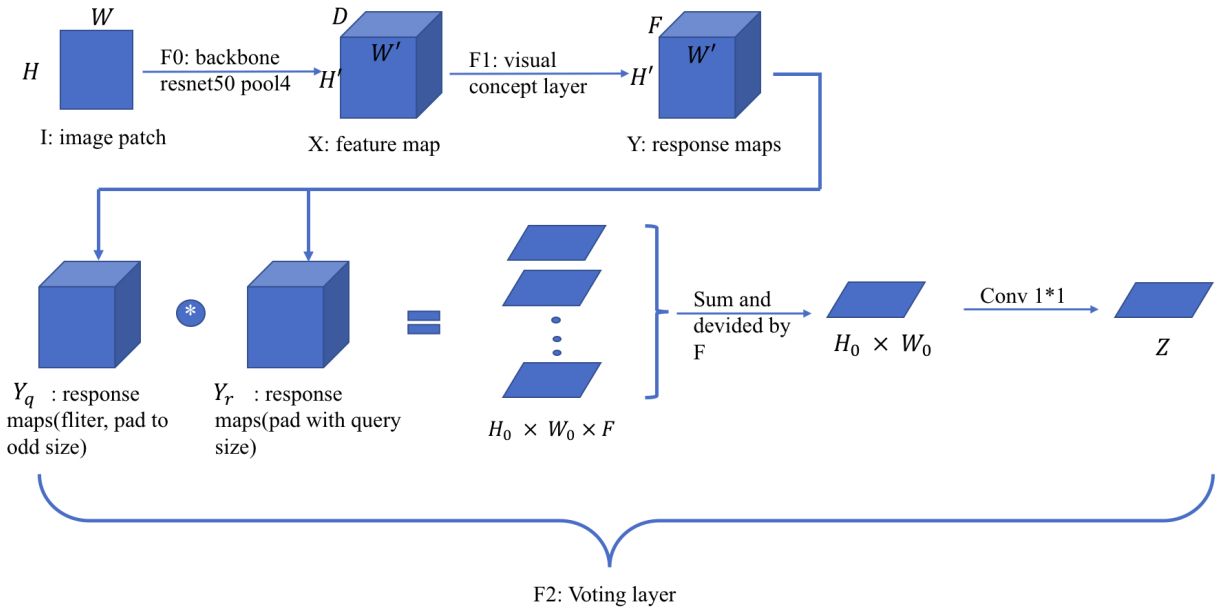


Figure 2. Voting Branch. We first go through F_0 :Backbone of resent50 and utilize a visual concept(VC) layerF_1 to generate response maps which represented in lower left quarter as Y_q: response map of query set and Y_r : response map of refenrence set. These two images both highlight in left and upper corner which means VC layer learns consistent semantic meaning there. Then go through a ConvNet to get the predicted center z: voting heatmap, size $W' \times H'$.

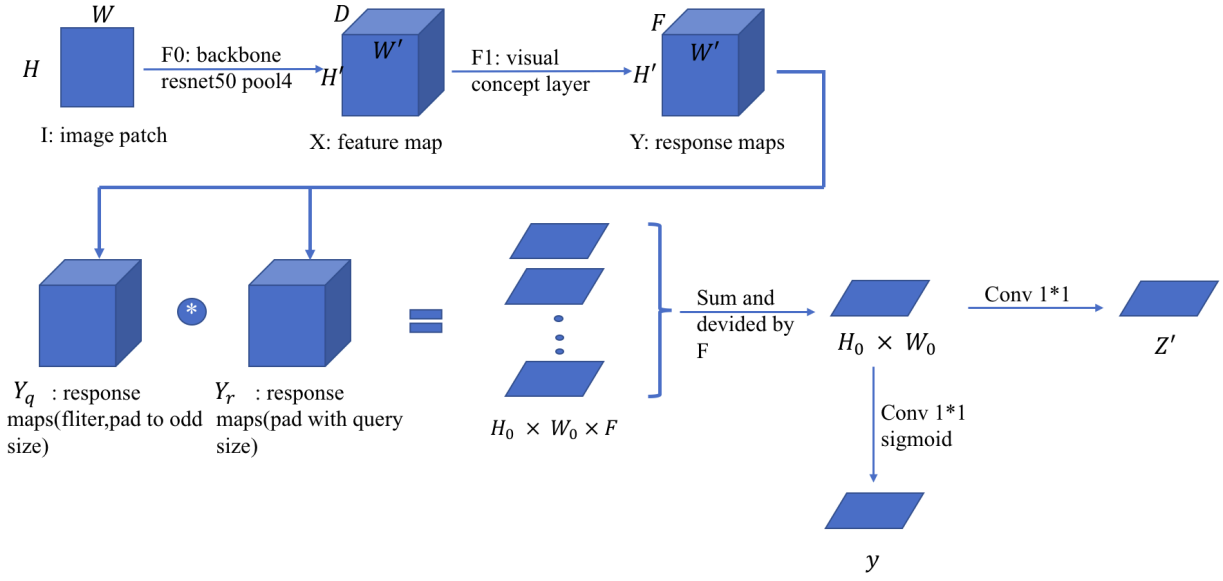


Figure 3. Score Branch. y is the similarity score between query and reference, z' is the voting center for votng branch but as regularization.

while training, we consider query and reference set data as pairset. Here is our training pairset-data. We compare Mask R-CNN to the state-of-the-art methods in amodal segmentation in Table 2.

Data Number			
train_pos	train_neg	val_pos	val_neg
30614	28532	18638	15529

Table 2. Negetive maskIoU threshold is 0.55, positive maskIoU threshold is 0.75

5.2. Voting and score Result

Voting branch give us center-aligned result, which predicts object center. And here we use two metrics: *mindelta* and *rgood*(0.1) to evalute our result of voting in Table 3. Score branch predicts similarity between query and reference set while evaluating in three accuracy metrics: *pos_acc*, *neg_acc* and *acc_all* in Table 4. Finally, the visualization results in Figure 5 shows response maps through VC layer. Figure 4 presents our voting branch result. And score branch's result is in Figure 6.

Voting Branch	
mindelta	0.067
rgood(0.1)	0.76

Table 3. we compute each detla between voting result and ground truth. *mindetla* means we get minimum delta from the best epoch. *rgood*(0.1) means relative error of both x and y axis is within 10%.

Score Branch	
pos_acc	0.742
neg_acc	0.847
acc_all	0.793

Table 4. *pos_acc* and *neg_acc* show positive and negative pairs' accuracy. The accuracy of positive and negative are balanced.

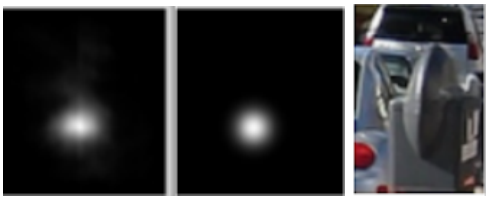


Figure 4. From left to right: Voting result, voting ground truth, query image patch. Voting result is close to voting target which proves voting accuracy. The voting result heatmap shows the direction and distance relative to voting target. When voting result is below the central point, it means our real center in image patch is above the central point.

Figure 5. First row shows query image patch on the left and ref set on the right. Response map of query and ref are on the second and third row. Response maps generated from VC layer indicates some semantic parts in different maps. Second row and third row show that VC layer learns the consistent semantic meaning of query and ref, so they highlight on the same parts.

Figure 6. First column shows score branch regularization result, and image patches are on the right. One car generates one highlight heatmap while two ouputs two highlight points.

Mask Transfer		
	meanIoU	AP
Mask R-CNN	0.5216	46.5
Our Method	0.6970	40.2

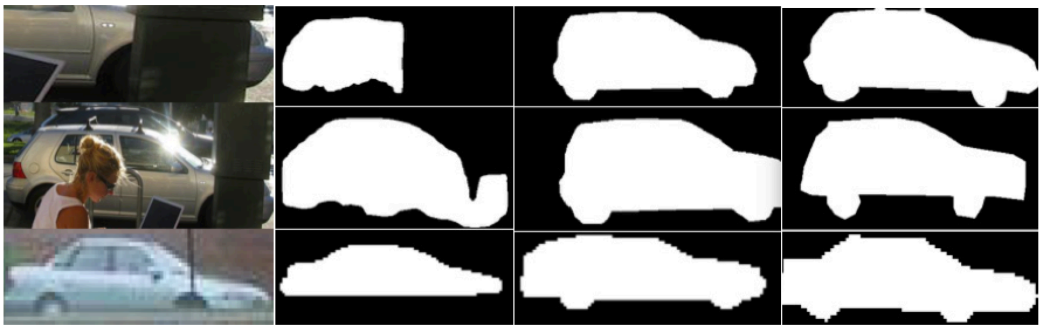
Table 5. *meanIoU* means IoU between predicted mask and ground truth mask. Our method improves meanIoU but get AP result not so good.

5.3. Mask Transfer

The key idea of mask transfer is to find the most similar complete mask of reference and put into the query's correct position. The visualization is shown in Figure 7 . In our result, we use two metrics: meanIoU and mAP, result shown in Figure 5. The analysis of lower AP is shown in Figure 6The key componet matching need further improved, we could use feature transfer in our feature work to get better results.

6. Conclusion

We proposed a Mask transfer system to solve the amodal instance segmentation problem. We design a



(a) Origin image patch (b) Mask R-CNN proposal (c) Our Result (d) Ground Truth

Figure 7. First colume(a) shows the origin image patches. Second column(b) shows Mask R-CNN proposals which is incomplete while third column(c) is the predicted complete mask for our method. And the last(d) is the ground truth mask.

Analysis for lower AP

IoU_thres	0.5	0.65	0.75	0.9	meanAP
Mask R-CNN	75.3	66.2	49.8	10.7	46.5
Our Method	84.4	68.9	25.6	0.0	40.2

Table 6. We get better AP in lower IoU thresholds but worse in higher IoU thresholds. Maybe there is some weakness in VC layer-based matching, it could be our future work to consider an alternative matching strategy to fix this problem.

part-based voting and score system and build an external dataset to do the mask transfer. The results show that the prediction of amodal and in particular invisible masks is a difficult task that needs further research.

References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. IEEE transactions on pattern analysis and machine intelligence, 33(5):898–916, 2011.

[2] M. Bai and R. Urtasun. Deep watershed transform for instance segmentation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2858–2866. IEEE, 2017.

[3] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3992–4000, 2015.

[4] J. Dai, K. He, and J. Sun. Instance-aware semantic segmentation via multi-task network cascades. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3150–3158, 2016.

[5] K. Ehsani, R. Mottaghi, and A. Farhadi. Segan: Segmenting and generating the invisible. arXiv preprint arXiv:1703.10239, 2(3), 2017.

[6] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence, 32(9):1627–1645, 2010.

[7] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In European Conference on Computer Vision, pages 297–312. Springer, 2014.

[8] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 447–456, 2015.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[10] X. He and S. Gould. An exemplar-based crf for multi-instance object segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 296–303, 2014.

[11] A. Kar, S. Tulsiani, J. Carreira, and J. Malik. Amodal completion and size constancy in natural scenes. In Proceedings of the IEEE International Conference on Computer Vision, pages 127–135, 2015.

[12] K. Li, B. Hariharan, and J. Malik. Iterative instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3659–3667, 2016.

[13] K. Li and J. Malik. Amodal instance segmentation. In European Conference on Computer Vision, pages 677–693. Springer, 2016.

[14] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang. Scene graph generation from objects, phrases and region captions. In ICCV, 2017.

[15] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4438–4446. IEEE, 2017.

[16] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan. Proposal-free network for instance-level ob-

ject segmentation. arXiv preprint arXiv:1509.02636, 2015.

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

[18] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431–3440, 2015.

[19] P. H. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In 31st International Conference on Machine Learning (ICML), number EPFL-CONF-199822, 2014.

[20] P. O. Pinheiro, R. Collobert, and P. Dollár. Learning to segment object candidates. In Advances in Neural Information Processing Systems, pages 1990–1998, 2015.

[21] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In European Conference on Computer Vision, pages 75–91. Springer, 2016.

[22] V. Premachandran, B. Bonev, and A. L. Yuille. Pascal boundaries: A class-agnostic semantic boundary dataset. arXiv preprint arXiv:1511.07951, 2015.

[23] B. Romera-Paredes and P. H. S. Torr. Recurrent instance segmentation. In European Conference on Computer Vision, pages 312–329. Springer, 2016.

[24] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229, 2013.

[25] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In European conference on computer vision, pages 1–15. Springer, 2006.

[26] J. Uhrig, M. Cordts, U. Franke, and T. Brox. Pixel-level encoding and depth layering for instance-level semantic labeling. In German Conference on Pattern Recognition, pages 14–25. Springer, 2016.

[27] J. Wang, C. Xie, Z. Zhang, J. Zhu, L. Xie, and A. Yuille. Detecting semantic parts on partially occluded objects. arXiv preprint arXiv:1707.07819, 2017.

[28] J. Wang, Z. Zhang, C. Xie, V. Premachandran, and A. Yuille. Unsupervised learning of object semantic parts from internal states of cnns by population encoding. arXiv preprint arXiv:1511.06855, 2015.

[29] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016.

[30] Z. Zhang, C. Xie, J. Wang, L. Xie, and A. L. Yuille. Deepvoting: A robust and explainable deep network for semantic part detection under partial occlusion. Technical report, Center for Brains, Minds and Machines (CBMM), 2018.