Comparing Multiple Instance Learning Video Segmentation Techniques

Alex Wong
Boston University
awong1@bu.edu

Wei-Hsiang Lin
Boston University
shawnlin@bu.edu

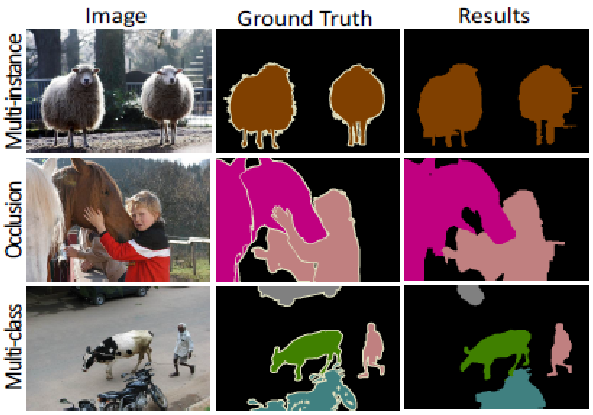

Figure 1. Semantic Segmentation with Peak Response Mapping

1. Introduction

In this paper, we experiment different techniques in the area of Multiple Instance Learning. We are interested in this particular area, because labeled data are expensive and Multiple Instance Learning tries to solve this particular problem. Furthermore, we plan to apply the techniques on video frames to see how well it performs.

Multiple Instance Learning (MIL for short below) refers to a type of supervised learning problem where the model learns to predict instance labels given supervision of whether the image contains the object(s). This type of problem, also referred to as weakly supervised learning problem, targets at training object segmentation models with only image-level labels.

Recent works of instance or class segmentation achieved promising results in several task domains such as medical imaging, traffic monitoring, etc. However, typical supervised training regime for these tasks requires extremely large amount of labeling effort since the model are supervised on pixel-level ground truth. On the other hand, target objects are statistically more likely to be heavily piled, occluded or silhouetted in some of the benchmark datasets [5, 8, 1]. This increases the difficulty of producing accurate boundaries for complex image scenes.

2. Related Works

Following the success of applying convolutional networks (Convnets) to solve classification problem [3], other computer vision tasks which requires model to perceive local information, such as object detection and image segmentation, involves Convnet structure to produce state-of-the-art results.

2.1. Fully Convolutional Multi-class Multiple Instance Learning

Pathak's work [4] pioneered in applying Fully Convolution Network (FCN) on weakly supervised semantic segmentation tasks. Their work shown that after substituting fully connected layer for 1x1 convolutional layers, the model will be able to generate class activation maps, or heat maps that indicates the area that are more likely to contain object of interest. They also proposed a trivial MIL loss for such procedure to teach the model with image-level class labels.

An obvious drawback of FCN [4] is that it performs poorly when segmenting multiple occluded objects. [7] took on the challenge and proposed an additional layer called Peak Stimulation Layer that enables the FCN model to output multiple object attention values for each class. Traditional classification loss functions are used but apportioned to all peak responses, therefore model can reflect on each object even when occluded together, and produce more accurate segmentation boundary as shown in Fig. 10.

2.2. Adaptive Noisy-And Global Pooling Layer

Kraus's work [2] to experiment global aggregation layers proposed Noisy-And pooling layer. This type of pooling layer reduces the negative impact of outlier objects. The underlying idea is that it activates a class probability when the mean of the instance level probabilities is above a certain threshold. The Noisy-AND pooling function is defined as follow:

$$P_i = g_i(p_{ij}) = \frac{\sigma(a(p_{ij} - b_i)) - \sigma(-ab_i)}{\sigma(a(1 - b_i)) - \sigma(-ab_i)} \qquad (1)$$

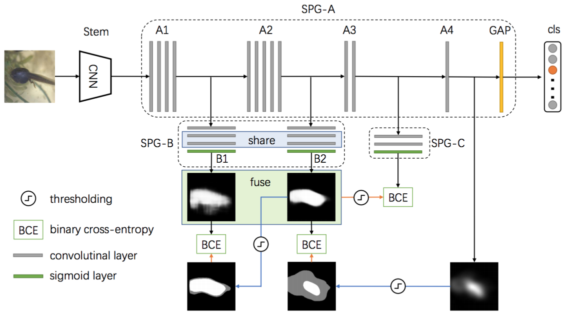

Figure 2. Architecture for the Self-Produced Guidance model

2.3. Self-Produced Guidance

[6] propose generating attention maps in a stage-wise manner, and let generated masks serve as supervision for each other. Zhang argues that previous methods focused on making sure the accuracy of the peak confidence position, that they lose track of tightening object boundary in the attention map. [6] introduced thresholding policy for the stage-masks and only let the foreground pixels backpropagate its gradient. Architecture of the Self-Produced Guidance method is shown at Fig. 2

3. Model

In [6], models are trained on images and their ground truth of class presence to produce attention maps that strongly correlates to visual cues of the object.

3.1. FCN Architecture and Peak Stimulation

The fully connected layers in CNN are substituted by 1x1 convolution layers to construct a fully convolutional network, the output of which is called class response maps. Such model preserves spatial information therefore is adaptable to input image sizes and suitable for pixel level predictions.

An additional peak stimulation layer is added after the last convolutional layer. Denote the class response maps as $M \in R^{C\times H\times W}$, with C the number of object classes and $H \times W$ the size of the class response maps. M is the input to the stimulation layer and the output is class-wise confidence score $s \in R^{C}$. For the map M^c of class c, define peaks as the maximum of windows of size r, which is set to 3 suggested by original works. The peak location is denoted as $P^c = \{(i_1, j_1), (i_2, j_2), ..., (i_{N^c}, j_{N^c})\}$, where N^c is the number of peaks in class c. A kernel $G^c \in R^{H\times W}$ calculates s as

$$G^c_{x,y} = \sum_{k=1}^{N^c} f(x - i_k, y - j_k) \tag{2}$$

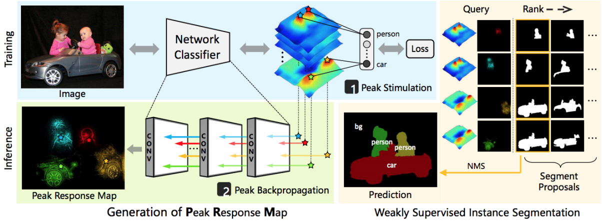

Figure 3. Architecture of Peak Response Maps (PRM).

where $0 \leq xH$, $0 \leq yW$, $(x - i_k, y - j_k)$ is the coordinate of the k-th peak. f is a Dirac Delta sampling function which aggregates the peak values to the features. Then we have

$$s^c = M^c G^c = \frac{1}{N^c} \sum_{k=1}^{N^c} M^c_{i_k, j_k} \tag{3}$$

The computation of class-wise confidence uses only peak values and the gradient in back propagation is

$$\delta^c = \frac{1}{N^c} \cdot \frac{\partial L}{\partial s^c} \cdot G^c \tag{4}$$

where L is the classification loss.

3.2. Proposed Method

Comparing to the ImageNet [5] dataset which [4, 6] evaluated on, video scenes in the robotic recycling scene contains more objects that are heavily occluded with each other, which makes it harder to segment out the objects by simply adopting MIL (Multiple Instance Learning) methods. To overcome the occluding nature of the robotic recycling dataset, we plan to adopt the architecture as proposed in [6], and further experiment the Noise-AND pooling function proposed in [2].

We will start by reproducing the baseline methods in [6]. An illustration of the baseline architecture is shown at Fig. 4. A pre-trained backbone FCN is used to extract visual feature of the video frame, the output feature maps are then fed into a 1x1 conv layer to produce class activation maps. A peak stimulation layer is constructed on top of the class activation maps so that we not only keep track of the global maximum class activations, but also the top K local maximum ones. The output of the peak stimulation layer, or the peak response maps, are then used to calculate actual object boundaries. For different peak response maps within the same class, we apply Non Maximum Suppression (NMS) algorithm to merge similar object proposal, and re-rank the generated maps according to confidence.

Figure 4. Example of the KITTI dataset.

Method	Accuracy	IOU Score
Noisy-And	0.75	N/A
FCN	0.934	0.603
SPG	0.996	0.691

Table 1. Results on KITTI dataset.

4. Experiments

4.1. Dataset

We chose two datasets to evaluate the MIL techniques. The datasets we chose are the KITTI surveillance dataset [1] and AiSkyEye VisDrone2019 dataset [8]. We plan to evaluate our model and baseline using the Mean Average Precision (MAP) metric for the object instance predictions.

4.1.1 KITTI Dataset

The KITTI dataset contains the image sequences for multiple videos. In the KITTI dataset, we are interested in identifying pedestrians.

In the pre-processing step, we crop the original image into a square of 224x224. The exact region to crop is decided randomly. The crop regions are then classified into two classes; 'has_pedestrian' if the cropped image contains a pedestrian and 'no_pedestrian' otherwise. The train-validation-test data distribution is 3540/505/1013.

4.1.2 AiSkyEye Dataset

The AiSkyEye dataset also contains the image sequences for multiple videos. However, in this dataset, we are interested in identifying cars.

The pre-processing step is very similar to that of the KITTI dataset. The train-validation-test data distribution is 21806/1632/5578.

4.1.3 Noisy-And Global Pooling Layer Implementation

While the Noisy-And global pooling layer is a pooling layer, we chose to experiment it by implementing it to the same neural network mentioned in [2].

5. Results and Evaluation

We evaluated the performance of our implementation of the SPG model, along with two other baselines, on the two

Method	Accuracy	IOU Score
Noisy-And	0.65	N/A
FCN	0.857	0.491
SPG	0.883	0.606

Table 2. Results on AiSkyEye dataset.

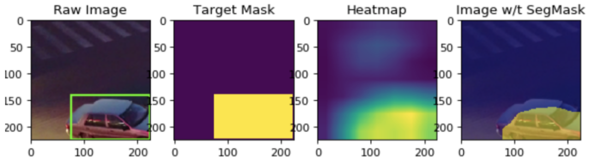

Figure 5. Example of result of FCN heatmap on AiSkyEye

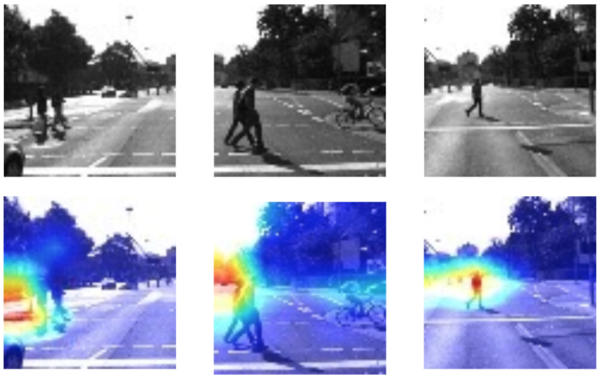

Figure 6. Example of result of Noisy_And layer on KITTI

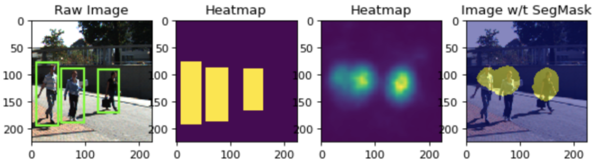

Figure 7. Example of good result of SPG on KITTI

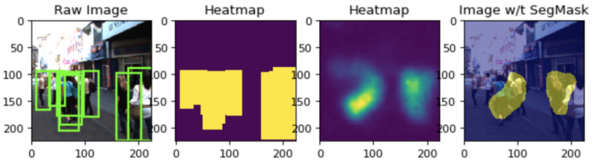

Figure 8. Example of bad result of SPG on KITTI

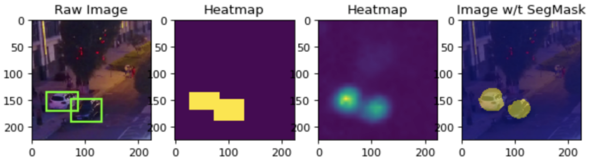

Figure 9. Example of good result of SPG on AiSkyEye

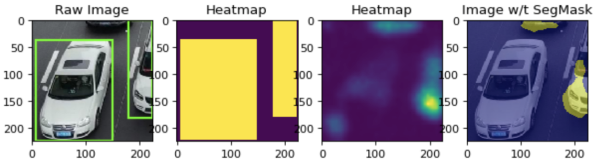

Figure 10. Example of bad result of SPG on AiSkyEye

datasets in [1] and [8]. The results are shown in Table 1 (for KITTI dataset) and Table 2 (for AiSkyEye dataset).

5.1. Accuracy

In terms of accuracy in the KITTI and AiSkyEye dataset, the SPG model performed the best, followed by FCN, then Noisy-And. SPG achieved 99.6% accuracy for KITTI dataset and 88.3% accuracy for AiSkyEye dataset. We believe the 11.3% accuracy difference to be a cause of the lack of quality and annotation accuracy in the AiSKyEye dataset. When we manually look into the AiSkyEye dataset, we found many inconsistency, for example: sometimes the object of focus (car) is there, but it is not annotated, while other times, the object of focus (car) is not there, but the annotation says it is there.

The Noisy-And global pooling layer performed significantly worse than the other two models. We believe the reason for this is because the Noisy-And pooling layer is originally designed for cells, where the presence of a class affects the majority of the cell. As a result, the mean of instance level probabilities to surpass a certain threshold is more likely. In our datasets, however, the presence of the object of focus does not necessarily affect the majority of the image.

5.2. IOU Score

In terms of IOU score in the KITTI and AiSkyEye dataset, SPG performed the best, followed by FCN. It's important to note that the Noisy-And global pooling layer does not have an IOU score because it is a classification problem; IOU score is specifically for segmentation problems.

Since SPG has a better accuracy, It is no surprise that SPG has a better IOU score as well. We believe a better IOU score is the cause to achieving better accuracy, as the data is detected more precisely.

5.3. Example of Heatmap

The heatmap generated from the models are shown in Figure 4 to 9.

Figure 4 and Figure 5 are heatmaps generated by FCN and Noisy_And layer respectively. Since the focus is on SPG, we did not include the poor results and picked good examples to show. As we can see, even in the worse-performing method, the Noisy-And, the model is able to identify the approximate region of the object of focus.

Figure 6 to 9 are heatmaps generated by SPG. For each dataset, we picked a good and a bad example to display. As shown in Figure 6, the model is able to identify 3 pedestrians and their region very well. However, as shown in Figure 7, the model is unable to map the individual instances well when there are many instances close to each other.

Figure 8 is a good result of SPG on AiSkyeye. Again, the heatmap can pick the approximate region of the car, however, there are times when the model barely picks up any signal from a very object car in the image, as shown in Figure 9.

6. Conclusion

In conclusion, we feel that we achieved very good results in the Multiple Instance Learning techniques mentioned in this paper. We were able to reproduce Noisy-And Global Pooling Layer and FCN baseline methods, and compare it to our implementation of SPG. We trained our model against two different datasets containing video sequences. In each dataset, the object of focus is different. Regardless, we achieved very good results in both datasets.

In the future, we want to dig deeper in the segmentation section and achieve the same (or better) results on multiple class datasets (as we only have 2 classes in our modified datasets). Futhermore, we want to adapt the techniques in this paper to achieve near-real time video segmentation to make our methods more practical.

References

[1] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1, 3, 4

[2] O. Z. Kraus, J. L. Ba, and B. J. Frey. Classifying and segmenting microscopy images with deep multiple instance learning. *Bioinformatics*, 32(12):i52–i59, 2016. 1, 2, 3

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[4] D. Pathak, E. Shelhamer, J. Long, and T. Darrell. Fully convolutional multi-class multiple instance learning. *arXiv preprint arXiv:1412.7144*, 2014. 1, 2

[5] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 1, 2

[6] X. Zhang, Y. Wei, G. Kang, Y. Yang, and T. Huang. Self-produced guidance for weakly-supervised object localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 597–613, 2018. 2

[7] Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, and J. Jiao. Weakly supervised instance segmentation using class peak response. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3791–3800, 2018. 1

[8] P. Zhu, L. Wen, X. Bian, L. Haibin, and Q. Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018. 1, 3, 4