Instance Segmentation using MASK R-CNN in Imperfect Conditions

Sai Koneru

May 2020

1 Introduction

Instance segmentation is the task of finding different objects in an image and also providing corresponding masks for each object in the scene. Every pixel in the image is mapped to a class that relates to an object. This can be useful in applications such as self-driving cars or autonomous robots with vision. One problem that may arise in this type of applications is that the image given by the sensors will not be perfect in most cases. Image enhancements in these cases are crucial as it highlights certain features that may be useful for the network to predict. As Convolutional neural networks are already computationally complex, using another network to enhance this image might not be feasible. In this report, an analysis is done using different image processing techniques to see if we can improve the performance and also in what conditions. This gives us insight if any pre-processing operations are needed when the input image is not perfect.

2 Related Work

After the discovery that Convolutional neural networks[5] work well on images, several sophisticated architectures are proposed to solve multiple computer vision problems. Instance segmentation can be split into two different tasks. First, recognize the objects in the scene and Second, compute masks for every object in the scene. To detect the objects R-CNN[3] can be used which is a region-based convolutional neural network. It proposes different region proposals using selective search and then uses a CNN to compute features. Later, these features are then used as input to SVM to classify. As this method requires around 2000 forward passed per image, FAST R-CNN[2] was proposed to combine all of them into a single model using ROI-Pooling. Then FASTER[7] R-CNN was proposed that uses Region Proposal Network instead of selective search to speed up the network. Mask R-CNN[4] adds a fully connected CNN to FASTER R-CNN by which we get a mask for every object that we detected. In [1] the authors discuss how simple pre-processing such as histogram equalization to reduce the

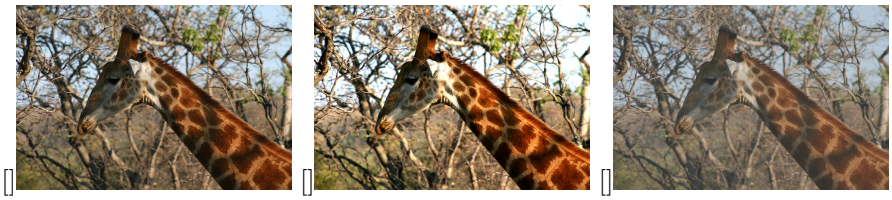

Figure 1: (a) Original (b) Contrast Up (c) Contrast Down

effect of illumination and then smoothing to reduce the noise produced by equalization can improve the performance. In this project, we would like to see is any pre-processing useful in the case of instance segmentation usking MASK R-CNN.

3 Methodology

Image can be corrupted or improper in different ways. It is reasonable to assume that in the case of self driving cars, some of the images may be coupled with noise. We need to take this into consideration that our image can be noisy and our model is able to cope with it and produce higher level of performance. Another reasonable thing to consider is the change in the contrast or brightness of the image. In cases where the car is actually on the road we cannot expect the illumination to perfect all the time. So we need to also take them into account.

To simulate these conditions we first sample around 100 images from the COCO data-set[6]. Then we add Gaussian noise to the image and we can perform experiments on different mean and standard deviation values. After this we randomly increase/decrease the contrast of the image as to replicate the lighting conditions in real time use. The way we increase or decrease the contrast and brightness of the image is shown in Equation 1.

$$J(i,j) = I(i,j) * contrast + brightness \, \forall i,j \tag{1}$$

Here if the contrast parameter is less than 1 there is a decrease in the contrast and increase if it is greater than 1. Similarly, to increase/decrease the brightness the brightness parameter has to be increased/decreased respectively. The effect of this can be seen in Figure 1

Now there are different techniques we can try to see if it increases the performance of the network. First thing we need to try is smoothing the image. There are different ways to smooth the image such as averaging, Gaussian blurring, Median blurring and bilateral filtering. Gaussian blurring can be useful in removing Gaussian noise where bilateral filtering can be useful when we want to have the edges sharp. We can also try sharpening the image as it can helpful in enhancing the image. For the assignment we have tested two different type of sharpening. First, where we apply a Laplacian filter on the image. Second, is to use unsharp masking where we use a Gaussian smoothing filter and then

subtract the original image. As there is also problem with illumination we have to also consider techniques that balance contrast. Histogram Equalization balances the contrast by transforming the intensities of the image into a uniform distribution. As it is done on the complete image and not on the local level, it is better to use Contrast Limiting Adaptive Threshold Equalization(CLAHE) to retain and improve the information in the image.

One problem here is that for some of the techniques mentioned above we cannot pass the RGB image directly. If we use the three channels separately and then join them together after performing operations this will not give a accurate result. To solve this we have to convert the image into HSV format, extract the Value channel, perform operations and convert the image back to RGB.

4 Results

In order to find out if there is any enhancements that we can do given that we know how likely the input is going to be corrupted with, several experiments are conducted to analyze the impact of these techniques.Here we first set a baseline score that we would like to beat without any operations on 100 images. In order to have a more finer analysis, the experiments are conducted under 5 different conditions. First, when there is low noise applied to the image and no contrast/brightness change to the image. Second, when there is high noise applied and no contrast/brightness change to the image. Third, randomly changing contrast to the image and for the last two adding low/high noise with randomly changing contrast of the image.

Table 1: MAP @ IoU = 50

	Baseline	Smoothing Bilateral	Upmask Sharpening	CLAHE	Smoothing Bilateral + Upmask Sharpening	Upmask Sharpening + CLAHE	CLAHE + Smoothing	CLAHE + Smoothing + Unmask Sharpening
Low Noise	0.669	0.642	0.657	0.640	0.636	0.593	0.632	0.606
High Noise	0.639	0.626	0.625	0.604	0.631	0.555	0.603	0.618
Random Contrast Change	0.568	0.498	0.553	0.573	0.512	0.557	0.504	0.499
Low Noise + Random Contrast Change	0.554	0.510	0.541	0.540	0.523	0.544	0.506	0.493
High Noise + Random Contrast Change	0.530	0.505	0.536	0.540	0.502	0.524	0.493	0.488

To evaluate this just the pixel accuracy is insufficient as most of the pixels can belong to one class and this might lead us to inaccurate interpretations. So, in order to evaluate instance segmentation we calculate the mean accuracy when the intersection over union is at 50 percent. The results of several experiments under multiple conditions can be seen in Table 1 and also an example of the output can be seen in Figure 2

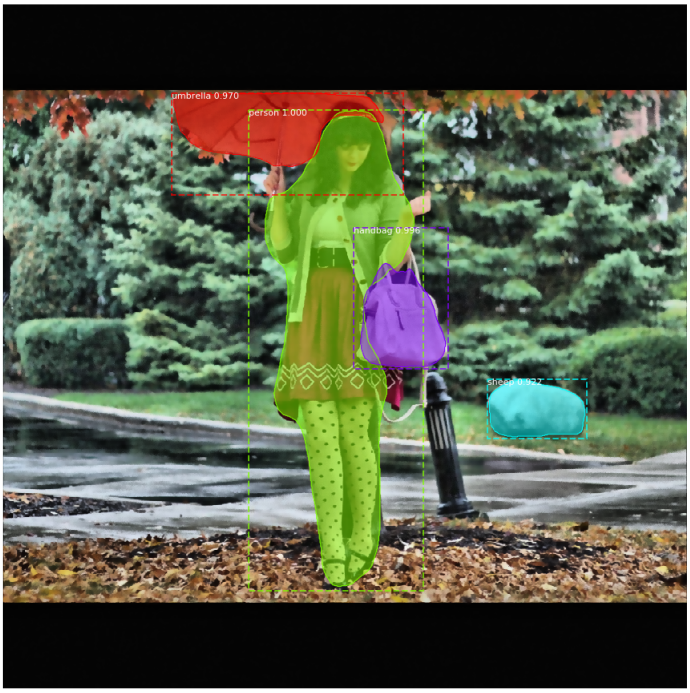

Figure 2: Output of Mask-RCNN

5 Discussion

It is important to note that CLAHE is applied before smoothing as the noise induced by it will be smoothed. In most of the cases the filters or contrast enhancement was unable to beat the baseline score. In cases where there no contrast change is applied it seems that performing smoothing operators only decrease the performance. This might be the case because the convolutions layers in the network are able to handle this noise and smoothing only decreases the information. If we apply sharpening then we will only be increasing the impact of noise and so it also had a score less than the baseline. When there is random contrast change in the input images, it seems that performing contrast enhancement techniques help the model. If we also add noise with random contrast change to the image, then the performance of the model decreases significantly and nothing seems to help in getting a better score unless when there is high amount of noise. In the last case as shown in Table 1, unmask sharpening and CLAHE individually seem to do a better job than combining them together. Even though the score is a little better, it is still not good enough.

6 Conclusion

To conclude, several experiments were conducted under different conditions and analyzed when operating with different enhancement techniques. It seems to be the case that the model is more sensitive to contrast compared to noise. It maybe because there is no inbuilt part in the model that handles contrast unlike noise which can be handled in convolution layers. Simple enhance techniques are not sufficient to make them robust. Advance techniques that uses the activation map of the model in order to enhance based on the local properties are needed to handle images captured in in perfect conditions.

References

[1] Raphaël Feraud, Olivier Bernier, Jean Viallet, and Michel Collobert. A fast and accurate face detector based on neural networks. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, pages 42 – 53, 02 2001.

[2] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[3] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

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

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[7] 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*, pages 91–99, 2015.