DETECT TRAFFIC SIGNS FROM LARGE STREET VIEW IMAGES WITH DEEP LEARNING

by

ZHIFEI DENG

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON
December 2019

To my parents.

ACKNOWLEDGEMENTS

I hold great appreciation to my supervising professor, Dr. Junzhou Huang, who provided great guidance for my thesis. I would also like to thank every lab members in the SMILE lab. Thank you all for your help.

November 11, 2019

ABSTRACT

DETECT TRAFFIC SIGNS FROM LARGE STREET VIEW IMAGES WITH DEEP LEARNING

ZHIFEI DENG, M.S.

The University of Texas at Arlington, 2019

Supervising Professor: Dr. Junzhou Huang

Autonomous driving is about to shaping the future of our life. Self-driving vehicles produced by Waymo or many other companies have demonstrated excellent driving capabilities on the road. However, accidents still happen. Correctly recognising the traffic signs, such as stop signs, is critical for a self-driving vehicle. Failing to recognise the traffic signs could lead to fatal accidents. Meanwhile, computer vision technology has made huge progress since the advent of deep learning, for example, image classification, object detection, and instance segmentation. Efforts have been made in developing faster and more accurate object detection methods. Faster R-CNN stands out as one of the most popular framework for object detection. Although frameworks like Faster R-CNN achieved state-of-the-art results in generic object detection, few endeavours have been made for traffic sign detection.

Detecting traffic signs from street view images is much more challenging than detection of generic objects from natural images. Street view images have high resolution, while traffic sign tends to be small in those images. Complex background in street view images also adds more difficulty in detecting traffic signs. In this thesis,

we proposed a novel two-stage object detection method for solving the challenging problem of detecting traffic signs from large street view images. In the first stage, we detect some less accurate regions which might contain traffic signs. Then we zoom in those candidate regions, and find the exact location of traffic signs in the second stage. The proposed method achieves AP (average precision) of 0.85 on a large street view dataset from an industry partner, which outperforms Faster R-CNN greatly, whose AP is around 0.13. The result reflects the potential of using the two-stage approach to detect small objects from high resolution images.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iv

ABSTRACT v

LIST OF ILLUSTRATIONS ix

LIST OF TABLES x

Chapter Page

1. INTRODUCTION 1

1.1 Object Detection 1

1.2 Traffic Sign Detection 3

1.3 Problems & Challenges 4

1.4 Goal of Thesis 5

2. TWO-STAGE METHOD FOR TRAFFIC SIGN DETECTION 7

2.1 Baseline 7

2.2 Two-stage method for traffic sign detection 7

3. EXPERIMENTAL RESULTS 9

3.1 Dataset 9

3.2 Pre-processing 9

3.2.1 Crop sky and ground 9

3.2.2 Splice image 10

3.2.3 Augment annotation 10

3.3 Post-processing 10

3.4 Experimental setup 10

3.5 Evaluation metrics 11

3.6 Results 11
4. CONCLUSION AND FUTURE WORK 12
REFERENCES 13
BIOGRAPHICAL STATEMENT 15

LIST OF ILLUSTRATIONS

Figure Page

a (a) Object Classification 6

b (b) Object Detection 6

c (c) Semantic Segmentation 6

d (d) Instance Segmentation [5] 6

1.2 Recognition problems related to object detection. (a) Image classification, (b) object detection, (c) semantic segmentation, (d) instance segmentation. 6

LIST OF TABLES

Table Page

CHAPTER 1

INTRODUCTION

1.1 Object Detection

Object detection is a fundamental and challenging problem in computer vision. The goal of object detection is to determine if there are any instances of objects from a pre-defined set of object categories (such as cats, dogs, aeroplanes, and chairs) in an image. If an object instance is present in the image, its spatial location is usually returned, together with associated confidence scores. Spatial location is commonly returned via bounding boxes, i.e., an axis-aligned rectangle tightly encompassing the object. Object can be anything that can be seen and touched. Although there are more than millions of object in our world, the research community is mainly interested in detection of highly structured objects (e.g., faces, aeroplanes, and vehicles) and articulated objects (e.g., humans, dogs, and cats) rather than unstructured scenes (e.g., sky, cloud, and grass) [1]. As a fundamental task in computer vision, object detection forms the basis for scene understanding – make computers understand the world from images and videos. Object detection has been widely used to solve many complex problems, such as face recognition, pedestrian detection, robot vision, intelligent video surveillance, and autonomous driving.

Many problems in computer vision are closely related to object detection. Computer vision sees big breakthrough in object classification/recognition in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) image classification challenge [2]. For object classification, image-level annotation indicating the presence or absence of an object class in the image. For example, "there is a cat in the image" or "there are

some cars in the image". The annotation is a binary label. Thus only the presence or absence of object is of interest. Other information like the quantity of objects or their spatial location will be ignored. Therefore, object classification differs greatly from object detection. For image classification, the computer needs to tell what objects or attributes a image has. For example, whether the animal in the image is a dog or a cat? However, in object detection, the computer also needs to answer the question of "where". That is, where are the objects located in the image? The number of instances is also of interest in object detection – it needs to detect each object instance.

Object detection is also related to semantic image segmentation and instance segmentation. Semantic segmentation also faces an inherent relationship between semantics and spatial location [3]. It assigns a semantic class label to each pixel in an image. Instance segmentation needs to distinguish each instance of the same object class, a task of simultaneously solving object detection and semantic segmentation [4]. Unlike segmentation, object detection includes some background region in the bounding boxes. The difference between object classification, object detection, semantic segmentation, and instance segmentation is illustrated in Figure 1.1a to 1.1d. object classification, object detection, semantic segmentation, and instance segmentation collectively drive the ultimate goal in computer vision: scene understanding.

As a longstanding and challenging problem in computer vision, extensive research has been done for object detection. Early methods were based on geometric representations [6]. It later shifts to the application of statistical classifiers (e.g., SVM [7], and Neural Networks [8]) based on appearance features [9]. Later on handcrafted local invariant features obtained huge popularity, such as Scale Invariant Feature Transform (SIFT) [10] and Histogram of Gradients (HOG) [11]. However, Krizhevsky et al. proposed a deep neural network (AlexNet) in 2012, which achieved ground

breaking results on ImageNet [2]. Since then research focus of many computer vision problems has been on deep learning. The emerging of deep learning also leads to big breakthrough in object detection. For instance, deep learning models have achieved smaller image classification errors than human in the ImageNet challenge. So as in object detection. Deep learning based object detection methods, such as Faster-RCNN and SSD (Single Shot Multibox Detector), made huge progress on the PASCAL VOC dataset. However, there are still big gaps. For examples, Faster R-CNN's AP on PASCAL VOC is about 0.65, which leaves great space for improvement.

In addition, object detection can be divided into two categories: generic object detection and specific category detection. In generic object detection, the computer tries to differentiate many generic object categories, such as flower, aeroplane, ship, chair, etc.; for specific category object detection, only one object category will be of interest, such as human faces. Therefore, traffic sign detection falls into specific category object detection.

1.2 Traffic Sign Detection

Scene understanding is the ultimate goal of compute vision, to make the computer understand the world and take corresponding actions. Google Street View cars are travelling around and take images of the world. The computer takes those images and try to understand our world. As a result, those information can be added to Google Maps to provide better service in people's everyday life. Furthermore, companies like Waymo and Uber take bolder moves of building autonomous vehicles. Undoubtedly, it is extremely import for autonomous vehicles have exact and correct understanding of the roads. Traffic signs are import signals for self-driving. Failure to recognise them correctly could lead to fatal accidents.

1.3 Problems & Challenges

As aforementioned, traffic sign detection falls into specific category object detection. Frameworks like Faster R-CNN can be used to traffic sign detection as well. However, there are several problems of applying methods like Faster R-CNN and SSD to traffic sign detection. Firstly, by nature those methods are designed for generic object detection, instead of specific object detection. For specific object detection, we can explore some domain information of that category to design better detection algorithms. Face detection is a good example.

Besides, traffic sign detection is usually done on street view images with over millions of pixels; while a average PASCAL VOC image is around 600x800. Such high resolution images cannot fit into the memory directly, not even to mention that most neural networks only take fixed size input like 256x256 in AlexNet. So the street view image need to be compress to low resolution. Nevertheless, compressing image with millions of pixels to resolution of 256x256 causes great information loss, which produce big challenge for correctly recognise traffic signs and find their exact location.

While images in the PASCAL VOC dataset usually very simple background, such as in a living room. Street view images have complex backgrounds. They are taken in both urban areas and in the wilderness. In urban areas, traffic signs are mingled with many human made objects, such as store logo, advertisement board, graffiti etc. Images taken in rural areas have a lot of trees and grass. Traffic signs are blended or even partially hidden in the trees and grass. As a result, those complex background adds great difficulty for traffic sign detection.

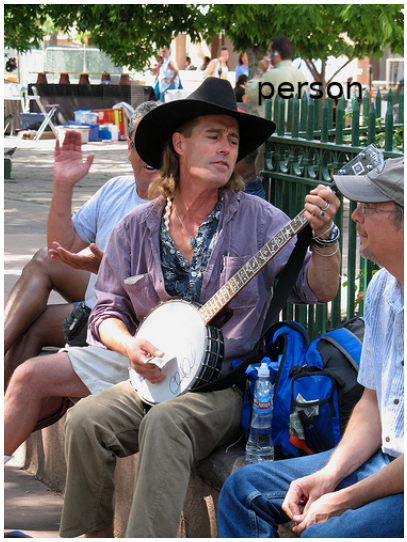
Last but not least, traffic signs only occupy a small region in the large street view image. In a street view image with resolution of 3260x4000, traffic sign could only be of size 40x40, which is about 0.003% of the whole image. After compressing the image, the traffic sign will become even smaller or even lost. More importantly,

as discussed in many literature, many object detection methods perform badly at detecting small objects. DenseNet pointed out that information flow at each layer in the neural network will cause the loss of some detailed information. Therefore, after many layers of information propogation in the neural network, information regarding to small traffic signs could be totally lost. It produces great challenge to recognise and detect small traffic signs in high resolution street view images.

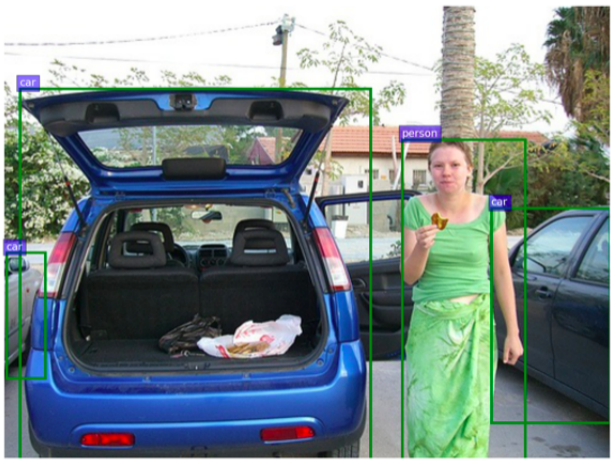
1.4 Goal of Thesis

The goal of this thesis is to push the boundary of traffic sign detection. As discussed before, object detection frameworks like Faster R-CNN are designed for generic object detection. And street view images have much higher resolution than those natural images used in PASCAL VOC. Faster R-CNN is also bad at small object detection. Therefore, we proposed a novel two-stage method to solve the challenging problem of traffic sign detection.

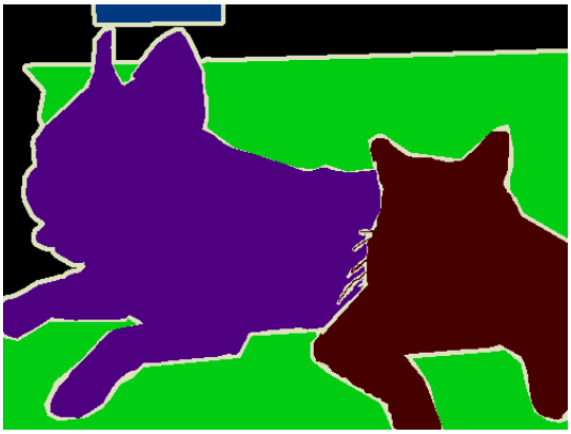
Our method is based on Faster R-CNN. In the first stage we find the rough location of the traffic signs. And then we zoom into that small region in the second stage and find the exact location of the traffic signs. Experiments show that our method beats Faster R-CNN completely, our method obtained AP of about 0.85 on the HERE street view dataset, while Faster R-CNN's AP is only around 0.013.



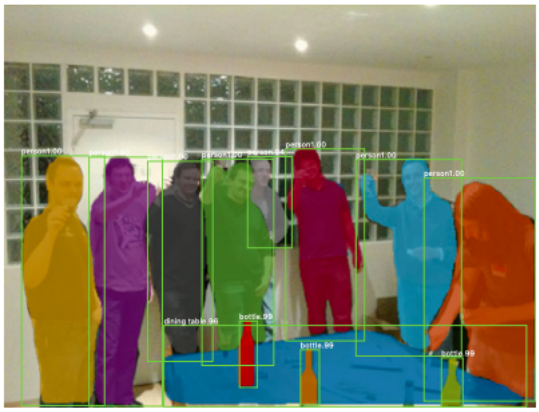
(a) Object Classification



(b) Object Detection



(c) Semantic Segmentation



(d) Instance Segmentation [5]

Figure 1.2: Recognition problems related to object detection. (a) Image classification, (b) object detection, (c) semantic segmentation, (d) instance segmentation.

CHAPTER 2

TWO-STAGE METHOD FOR TRAFFIC SIGN DETECTION

2.1 Baseline

We use Faster R-CNN as our baseline method, which is a strong baseline method for object detection. Nevertheless, we need to tune its anchor size and ratios, because its default settings are for PASCAL VOC dataset and they are not good at catching small objects.

2.2 Two-stage method for traffic sign detection

One main challenge of detecting traffic signs from large street view images is detecting small traffic signs. Unlike the PASCAL VOC dataset, in which objects usually occupy a big ratio of space in the image, street view images have high resolution and the pixel ratio of small traffic signs can be as low as 0.000019%. On the other hand, most neural networks expect fixed size input. Directly feeding the large street view image into the neural network is not feasible. Either because of the memory limit or it leads to enormous parameters. As a result, we need to resize the street view image. However, after resizing the image, small traffic signs will become smaller or even unrecognisable. What's worse, after many layers of information propagation in a deep neural network, information of small traffic signs could be total lost.

Because of this reason, Faster R-CNN is not good at detecting small objects. But it achieves a good AP on PASCAL VOC dataset. Our two-stage method is built on top of Faster R-CNN. We try to convert the challenging problem of detecting traffic signs to easier problems which Faster R-CNN can solve. In the first stage,

we run Faster R-CNN to detect a large region which could potentially contain traffic signs. Thus we avoid the problem of detecting small objects in this stage. In the second stage, we train another Faster R-CNN model to detect traffic signs on the region produced in the first stage. As a result, even if the traffic signs are small, they still occupy a large portion in that region. Similarly, we don't have to deal with the problem of detecting small traffic signs in this stage.

CHAPTER 3

EXPERIMENTAL RESULTS

3.1 Dataset

During the work of this thesis, we collaborate with an industrial partner, who collected a large street view image dataset. Each segment is represented by 1,000 consequtive car locations. The car takes images every 6 meters, and thus each location is 6 meters from the previous one. Therefore, a segment shows images from approximately 6,000 meters of a drive path. At each location four pictures are taken - one by each front, left, rear, right cameras.

There are about 80,000 street view images in this dataset. The image resolution is 3264 x 4000. In particular, some small traffic signs only have 16 x 16 pixels. The images are taken both in the wilderness and in urban areas. Thus they have complex background. For example, for images taken in the wilderness, some traffic signs are partially hidden in the trees or grass, or they are taken in severe conditions, such as rain or strong glare.

This dataset also contains various types of traffic signs, such as speed limit and turn signs. The signs also come in a large range of size. Except for the small signs, which are of our main concern, it also has some large signs and medium signs.

3.2 Pre-processing

3.2.1 Crop sky and ground

The images have high resolution and it is problematic for our neural network. After plotting the heat map of traffic signs' location, we find that it is very unlikely

that traffic signs will appear at the very top or bottom of images. Because the top and bottom is sky and ground respectively. So we can crop them out to make the image smaller.

3.2.2 Splice image

After cropping the image, its resolution is still high. As discussed earlier, if we resize the image, information of small signs could be totally lost. Thus we split the image vertically into five smaller images. Then we can run traffic sign detection on each smaller image separately.

3.2.3 Augment annotation

In the first stage we need to detect large regions which might have traffic signs. So we need to enlarge the original annotation – making the original bounding much larger if the traffic sign is small. If the traffic sign is large, we can keep its original annotation or make it slightly larger.

3.3 Post-processing

Because we run traffic sign detection on the five smaller images individually, we need to merge the detection results. There is some overlap between two neighbouring smaller images. So that a traffic sign may appear in both images. We can use non-maximum suppression (NMS) to eliminate duplicate bouding boxes.

3.4 Experimental setup

We split the dataset evenly – 40,000 images for training and 40,000 images for testing.

3.5 Evaluation metrics

The models are evaluated using average precision (AP).

3.6 Results

For our baseline model, AP is 0.0137, while the two stage method obtains AP of 0.794.

CHAPTER 4

CONCLUSION AND FUTURE WORK

In this thesis we demonstrated the effectiveness of our two-stage approach for traffic sign detection. It works especially well for detecting small traffic signs from large street view images. The two-stage approach significantly outperforms directly applying Faster R-CNN.

However, there is still some space for improvement. Firstly, we may combine two stages into a single neural network to facilitate efficient training. Meanwhile, the two-stage approach can be connected with the attention mechanism. In the first stage the neural network pays attention to some regions which probably contain traffic signs. Then it zooms into that region. Thus we can add the attention mechanism to the two stage approach. Furthermore, instead of using Faster R-CNN, we can also experiment with SSD and YOLO.

REFERENCES

[1] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, Matti Pietikäinen. "Deep Learning for Generic Object Detection: A Survey", CoRR, vol. abs/1809.02165, 2018.

[2] Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.

[3] Evan Shelhamer, Jonathan Long, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. IEEE Trans. Pattern Anal. Mach. Intell., April 2017.

[4] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, Hartwig Adam. MaskLab: Instance Segmentation by Refining Object Detection with Semantic and Direction Features. CoRR, vol. abs/1712.04837, 2017.

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick. Mask R-CNN. CoRR, vol. abs/1703.06870, 2017.

[6] Jean Ponce, Martial Hebert, Cordelia Schmid, Andrew Zisserman. Towards category-level object recognition. Springer, 2006.

[7] Edgar Osuna, Robert Freund, Federico Girosi. Training Support Vector Machines: an Application to Face Detection. CVPR, 1997.

[8] Henry A. Rowley, Shumeet Baluja, Takeo Kanade. Neural Network-Based Face Detection. PAMI, 1998.

[9] Hiroshi MuraseShree K. Nayar. Visual learning and recognition of 3-d objects from appearance. IJCV, 1995.

[10] David G. Lowe. Object Recognition from Local Scale-Invariant Features. ICCV, 1999.

[11] Navneet Dalal, Bill Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005.

[12] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. NIPS, 2012.

BIOGRAPHICAL STATEMENT

Zhifei Deng received his Bachelors of Engineering in Software Engineering from Nankai University, Tianjin, China. He also obtain his Master's degree in Machine Learning from University College London, London, UK.