Technology Exploration Report

Summary: This document will detail the different techniques used when detecting traffic signals in an image and the corresponding methods that are going to be used for it. Some of the tools that will be used during the process will be detailed so that this is recorded for future possible applications.

	NAME	***FUNCTION***	***DATE***
AUTHOR(S)	*Cadix Martín, Víctor*	***Technology leader***	28/10/2019
	Rosado Junquera, Pablo J.	***Validation leader***	28/10/2019
	Casa Rodríguez, Alejandro	***Programming chief***	29/10/2019
APPROVAL	*De la Quintana Béjar, Hugo*	***Executive officer***	31/10/2019
	Rosado Junquera, Pablo J.	***Validation leader***	31/10/2019
AUTHORIZATION	*De la Quintana Béjar, Hugo*	***Executive officer***	02/10/2019

ARCHIVING CARTOUCHE

REFERENCE:	--	***ISSUE***	2.1	***DATE***	02/10/19
TITLE:	***Technology Exploration Report***				
SUBJECT:	***Computer Vision***				
TYPE:	***Technical Report***				
LANGUAGE:	***English***				
CANCEL PREVIOUS ISSUE:	***Yes***				

Team

DEPARTMENT / ROLE	***NAME***
Technology leader	*Cadix Martín, Víctor*
Programming chief	*Casa Rodríguez, Alejandro*
Executive Officer	*De la Quintana Béjar, Hugo*
Validation leader	*Rosado Junquera, Pablo J.*
Documentation	*---*

Revision record

ISSUE	*DATE*	*EFFECT ON*		*REASON FOR REVISION*
		PAGE	PARAGRAPH	
1.0	02/10/19	All	All	First Issue
2.0	30/09/19	6-9	5,6	More content added
2.1	02/10/19	10	8	References revised and other fixes
		3		Added changelog
		9	7	Methods and scores for the dataset
		9	8	Targets added

Table of Contents

1. Introduction

There are **three typical steps** in object detection methods:

1- A model or algorithm is used to generate ROI (Region of Interest) across the image. Those are bounding boxes of different sizes.
2- Features are extracted for each bounding box and evaluates which objects (if any) are present in the proposed regions.
3- In the post-processing overlapping boxes are combined into a single ROI.

2. Regions of Interest

2.1. Region proposal

At the beginning, the "*selective search*" algorithm used to be considered to generate object proposals. Other methods can extract regions with more complex features or patterns, like a deep learning model or using brute force. The key is the fact that the more regions there are, the better the detection will be, but it comes with a trade-off, which is the time it takes to process the image.

2.2. Anchor box

We can put some assumption on the shapes of bounding boxes. For example, if we want to detect humans, we should search humans with some vertical rectangular boxes. The anchor boxes are fed to the network, before training and prediction, as a list of some numbers, which is a series of pairs of width and height.

When using anchor boxes, you can evaluate all object predictions at once, making real-time object detection systems possible. Besides, anchor boxes eliminate the need to scan an image with a sliding window that computes a separate prediction at every potential position. Examples of detectors that use a sliding window are those that are based on aggregate channel features (ACF) or histogram of gradients (HOG) features.

3. Evaluation of objects

3.1. Feature extraction

It is known as the process of getting a fixed set of visual features from a non-predefined size image. Traditional approaches as histogram, filter-based methods or the more recent deep learning methods have the objective of extracting the relevant features for the task.

For the classification stage, deep learning methods as deep CNN are commonly used. Here, the process of **transfer learning** is the strategy, where a pretrained network is used and retrained for the new task. This allows reaching a faster training as the first layers of the network (the ones used for the feature recognition) are already trained. The only ones that need to be retrained are the top layers, used for the classification part. [1]

3.2. Histogram of oriented gradients (HOG)

This method is a feature descriptor, based on the computation of gradient orientation in localized portions of the image. The strategy is similar to SIFT and Shape Context but is computed on a dense grid of uniformly spaced cells and use overlapping local contrast for further improve the accuracy. There are also several examples of its use in traffics signs recognition [2].

3.3. Scale Invariant feature transform (SIFT)

It describes local features. SIFT key points of objects are first extracted from a set of reference images and saved in a database. Then, an object is recognized in a new image by individually comparing each feature from the new image to this database. A candidate is found when the algorithm gets a match on its features, computed through the Euclidean distance of vectors.

4. Post-Processing

4.1. Non-max suppression (NMS)

Is the method of selecting the bounding box with the highest probability from a set of near/overlapped boxes.

5. Machine Learning approaches

5.1. Support Vector Machine (SVM)

This strategy is widely used in classification problems. The main objective of the support vector machine algorithm is finding a hyperplane in an N-dimensional space that distinctly classifies the data points. It is usually used on top of the HOG algorithm for the classification part [3]. [Traffic Sign Recognition – How far are we from the solution]. It is also used in literature [4] to classify features obtained from a CNN used for feature extraction.

5.2. Viola-Jones framework

This algorithm is mainly used with face recognition, so we have decided not to consider it in this work.

5.3. Shape context

Commonly used in shape matching, this algorithm extracts the contours of a shape, and then pick some of those points to define vectors. These are defined by connecting each point to the others, as the descriptor.

6. Deep Learning approaches

These approaches can solve end to end detection and are typically based on convolutional neural networks. (Black box)

6.1. R-CNN

The method uses selective search to extract just 2000 regions from the image, the region proposals. Thus, the main difference is that instead of trying to classify a huge number of regions, it just works with a defined amount. [5]

Selective Search:

1. Generate initial sub-segmentation, generating many candidate regions.
2. Use an algorithm to recursively combine similar regions into larger ones.
3. Use the generated regions to produce the final candidate region proposals.

Problems:

1. It still takes a huge amount of time to train the network as it must classify 2000 region proposals per image.
2. Consequently, it cannot be implemented in real time as it takes around 47 seconds for each test image.

3. The selective search algorithm is a fixed algorithm. Therefore, no learning is happening at that stage. This could lead to the generation of bad candidate region proposals.

6.2. Fast R-CNN

The approach is similar to the R-CNN algorithm [6]. However, instead of feeding the region proposals to the CNN, it feeds the input image to the CNN to generate a convolutional feature map. The regions of proposals are identified from the feature map (Region Proposal Network). The main advantage is that it is no longer needed to compute 2000 regions, with a faster prediction achieved. It also exists the Faster R-CNN, an even better version of this algorithm.

6.3. SSD – Single Shot MultiBox Detector

This algorithm solves the object localization and classification in a single forward pass of the network. For the training phase, the error of the box position and size (compared to the ground truth) is calculated with the Intersection over Union ration (IoU). Also, it uses NMS for the post processing. The method achieved records in terms of performance and precision for object detection tasks, scoring over 74% mAP (mean Average Precision) at 59 frames per second on standard datasets [7].

Having MultiBox on multiple layers results in a better detection as well, due to the detector running on features at multiple resolutions. However, SSD confuses objects with similar categories, a phenomenon probably caused by locations that are shared for multiple classes. It produces worse performance on smaller objects, as they may not appear across all feature maps.

6.4. YOLO

It uses a single convolutional network to predict the bounding boxes and the class probabilities for these boxes [8]. The key aspects include:

1- Uses anchor boxes.
2- Supress the boxes with low probability.
3- For each class use non-max suppression to generate the result.

Its consequent versions are the state of art in object detection, allowing for real time object detection. In terms of speed, YOLO is orders of magnitude faster (45 frames per second) than other object detection algorithms [9].

YOLO sees the entire image during training and test time so it implicitly encodes contextual information about classes as well as their appearance. The method imposes strong spatial constraints on bounding box predictions since each grid cell only predicts two boxes and can only have one class. This spatial constraint limits the number of nearby objects that our model can predict.

The main limitation of YOLO algorithm is that it struggles with small objects within the image. (Same as SSD).

7. Algorithm performance

In this section we will briefly discuss about recent techniques used on the BTSD and its scores, based on literature [10]. At first, in [9] a domain transfer learning combined with a genetic algorithm is proposed for the optimization of the pretrained CNN for the BTSC (classification) achieving 99.16% accuracy.

Other strategies, such as the one used in [10], use a Haar cascade detector for the extraction of ROIs. Then a deep learning algorithm is used for the verification of the ROIs and, finally, a SVM is used for the classification part. The average precision rate of the algorithm is 98.81%.

The case of [11] includes a CNN made of residual convoluting blocs and hierarchical skip connections. They score 99.33% Accuracy in German sign recognition benchmark (GTSRB) and 99.17% accuracy in BTSC benchmark. Finally, in [4] it can be seen a CNN-SVM algorithm with a score of 98.6% accuracy.

8. Methodology

We will divide the detection and classification phases, so different techniques can be applied to each part. For the detection, we will use traditional computer vision techniques as HOG combined with clustering for image segmentation. The estimation of the regions of interest will be made from that information.

Once a ROI is generated, it will be transferred to a convolutional Neural Network for the classification part. For the CNN a pretrained network will be used in order to decrease the learning time.

It is also interesting to split detection and classification because there is a larger number of images of traffic signs for the classification problem than images for the detection part. This will allow to train our model on a wider variety of images.

The choice is made for purely didactic reasons, since the Visio De-Sign team considers it to be one of the best ways to apply the knowledge acquired in the Computer Vision course of the Master in Automation and Robotics at the UPM.

8.1. Target

The objective of this work is to create a system that is capable of recognizing one or several signals in an image taken through a camera installed in a car and treat them to facilitate its recognition.

Likewise, we expect to obtain success rates of at least 80% in the recognition of images in good conditions, and 50% in adverse conditions (rain, night ...).

9. References

[1] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Trans. Knowl. Data Eng.*, pp. 1345–1359, 2010.

[2] M. Mathias, R. Timofte, R. Benenson, and L. Van Gool, "Traffic sign recognition 2014; How far are we from the solution?," *2013 Int. Jt. Conf. Neural Networks*, pp. 1–8.

[3] C. Yao, F. Wu, H. Chen, X. Hao, and Y. Shen, "TRAFFIC SIGN RECOGNITION USING HOG-SVM AND GRID SEARCH School of Electronic and Information Engineering , Beijing Jiaotong University , Beijing , China," *2014 12th Int. Conf. Signal Process.*, pp. 962–965, 2014.

[4] Y. Lai, N. Wang, Y. Yang, and L. Lin, "Traffic Signs Recognition and Classification based on Deep Feature Learning," pp. 622–629, 2018.

[5] M. R-cnn, P. Doll, and R. Girshick, "Mask R-CNN."

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN : Towards Real-Time Object Detection with Region Proposal Networks," pp. 1–14.

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg, "SSD : Single Shot MultiBox Detector."

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection."

[9] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger."

[10] A Novel Genetically Optimized Convolutional Neural Network for Traffic Sign Recognition: A New Benchmark on Belgium and Chinese Traffic Sign Datasets.

[11] Deep Learning Traffic Sign Detection, Recognition and Augmentation.

[12] Total Recall: Understanding Traffic Signs using Deep Hierarchical Convolutional Neural Networks