A New Approach for In-Vehicle Camera Traffic Sign Detection and Recognition.

Andrzej Ruta, Yongmin Li
School of Information Systems, Computing & Mathematics
Brunel University
Uxbridge, Middlesesx
UB8 3PH, United Kingdom
{ Andrzej.Ruta,Yongmin.Li}@brunel.ac.uk

Fatih Porikli
Mitsubishi Electric Research Laboratories
201 Broadway, Cambridge
MA 02139, USA
fatih@merl.com

Shintaro Watanabe, Hiroshi Kage, Kazuhiko Sumi
Advanced Technology R&D Center
Mitsubishi Electric Corporation
Amagasaki, Japan

Abstract

In this paper we discuss theoretical foundations and a practical realization of a circular traffic sign detection and recognition system operating on board of a vehicle. To initially detect sign candidates in the scene, we utilize the circular Hough transform with an appropriate post-processing in the vote space. Track of an already established candidate is maintained using a function that encodes the relationship between a unique feature representation of the target object and the affine transformations it is subject to. This function is learned on-the-fly via regression from random distortions applied to the last stable image of the sign. Finally, we adopt a novel AdaBoost algorithm to learn a sign similarity measure from example image pairs labeled either "same" or "different". This enables construction of an efficient multi-class classifier. Prototype implementation has been evaluated on a video captured in crowded street scenes. Good detection and recognition performance was achieved for a 14-class problem which reveals a high potential of our approach.

1 Introduction

Road signs are an inherent part of the traffic environment. They are designed to regulate flow of the vehicles, give specific information, or warn against unexpected road circumstances. For that reason, perception and fast interpretation of signs is critical for the driver's safety. Therefore, when video processing became attainable on a computer machine, automation of the road sign detection and recognition process was found a natural direction to follow.

Different approaches were used in the past for detecting road signs. In the old studies, e.g. [2, 3], as well as in many recent ones, e.g. [9, 12, 15], it was common to employ a heuristic that utilized available prior knowledge about the traffic signs to 1) define how to pre-segment the scene in order to find the interest regions, and 2) define the acceptable geometrical relationships between the sign parts with respect to color and shape. The major deficiency of these methods were a lack of solid theoretical foundations and high parametrization. A more convincing method for detecting road signs was proposed by Bahlmann et al. [11]. They utilized the AdaBoost algorithm [4] and the rejection cascade framework to learn the most discriminative, color-parametrized Haar wavelet filters for road sign representation. In several studies, e.g. [2, 6, 9], the problem of tracking of the observed road signs over time was addressed. However, the proposed frameworks, with the exception of the two-camera system in [6], never went beyond a relatively simple scheme based on a predefined motion model and some sort of geometrical Kalman filtering.

For sign classification, typically a cross-correlation template matching technique was used [2]. Other approaches involved neural networks [9] or kernel density estimation [7] and were shown to offer relatively good classification accuracy. An interesting concept of a trainable, class-specific similarity measure was introduced recently by Paclík et al. [13] who demonstrated a usefulness of this method in solving relatively simple road sign classification problems. A similar approach was further presented by Ruta et al. [15] who adapted it to infer the discriminative sign representations using a single template image per class.

In this paper we present a unified framework for detection and recognition of traffic signs which alleviates the shortcomings of many previous approaches. For initial candidate detection we use a circular Hough transform. It is augmented with a refinement algorithm based on a *Confidence-weighted Mean Shift* clustering of the detector's responses. To track the existing signs, we employ a trainable regression function that compensates the affine distortions, making our detec-

tor pose-invariant and hence more accurate. Finally, we build a traffic sign classifier based on the concept of a trainable similarity. A novel AdaBoost algorithm is utilized to learn a robust sign similarity measure from image pairs labeled either "same" or "different". This measure is further directly used within the *winner-takes-all* classification framework. Good performance of our algorithms is demonstrated in two experiments where our demo implementation is used to recognize road signs in both static images and video stream.

The rest of this paper is divided into five parts. In section 2 our road sign detection method is discussed. In section 3 we develop a pose-invariant sign tracker. Section 4 explains how the concept of trainable similarity is used to construct a robust road sign classifier. In section 5 experimental evaluation of our algorithms is presented. Finally, in section 6 we conclude our work.

2 Sign Detection

Road sign detection is a difficult problem as it involves discriminating a large gamut of diverse objects from a generally unknown background. For simplicity, we consider in this work only the circular traffic signs.

2.1 Candidate detection

In order to detect the new road sign candidates emerging in the scene we utilize a circular Hough transform [1]. Each frame of the input video is first scaled to the size of 360×270 pixels. Then, focus is directed only on the regions of interest (RoI) determined using a quad tree technique as follows. As all the targeted signs have either red or blue rim, we identify the sign boundary gradients with respect to the abovementioned color channels as the key cues to be exploited. To extract these gradients, we propose a simple filter that amplifies the red and blue fragments of the scene:

$$\begin{aligned} f_{RED}(\mathbf{x}) &= \max(0, \min(\tfrac{x_R - x_G}{s}, \tfrac{x_R - x_B}{s})) \\ f_{BLUE}(\mathbf{x}) &= \max(0, \min(\tfrac{x_B - x_R}{s}, \tfrac{x_B - x_G)}{s})) \end{aligned} \quad , \quad (1)$$

where x_R, x_G, x_B denote the red, green and blue components of an input pixel and $s = x_R + x_G + x_B$. Further, from the obtained images we extract the color-specific gradient maps and their integral images [10].

To establish the regions of interest, we first define a minimum amount of red/blue gradient to be contained in a RoI. Then, the entire image is checked against the total color gradient contained using the appropriate integral image [1]. As it is typically far above the defined threshold, the image is subdivided into four quarters and each quarter is independently processed in the same way. The process is stopped either when the current input region contains less gradient than the threshold or upon reaching a predefined number of depth levels (6 in our case). The above-threshold lowest-level regions are clustered and the ultimate RoIs are constructed as bounding rectangles of the found clusters. This way we can very quickly discard the irrelevant fragments of the scene, e.g. sky or asphalt.

Now, in each found RoI a circular Hough transform is run separately on the red and blue gradient map. Instead of risking setting a too high threshold in the vote space, we keep it at a low level, but integrate the multiple hypothetical circles produced using the refinement technique discussed in the next section.

[1] This involves only 4 addition/subtraction operations.

2.2 Detection Refinement

In order to collapse the clouds of redundant hypotheses produced by the detector, we propose to treat the Hough response space as a probability distribution. A kernel density estimation technique is used to model this distribution and its maxima are found using a variant of the mean-shift algorithm [8]. We call it *Confidence-Weighted Mean Shift*.

We first characterize each positive hypothesis with a vector, $\mathbf{x}_j = [x_j, y_j, r_j]$, encoding the circle's centroid position and its radius. In addition, it is assigned a confidence value, q_j, which we relate to the normalized number of votes cast for this circle in the Hough space. Assuming that $f(\mathbf{x})$ is the underlying distribution of $\mathbf{x}$, stationary points of this distribution are found via alternate computation of the mean-shift vector, and translation of the current kernel window by this vector, until convergence (for details, refer to [8]). Our modified mean-shift vector is made sensitive to the confidence of the input points in the following way:

$$\mathbf{m}_{h,G} = \frac{\sum_{j=1}^{n} \mathbf{x}_j q_j g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_j}{h}\right\|^2\right)}{\sum_{j=1}^{n} q_j g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_j}{h}\right\|^2\right)} - \mathbf{x} \ , \quad (2)$$

where $g(\cdot)$ is the underlying gradient density estimator and h is the bandwidth parameter determining the scale of the estimated density. Incorporating the confidence terms q_j in (2) is equivalent to amplifying the density gradients pointing towards the more reliably detected circle locations. The found modes of $\mathbf{x}$ correspond to the new road sign candidates which we need to track in the consecutive frames of the input video. A few examples of such modes are illustrated in Fig. 1.

Figure 1: Output of the Hough circle detector before (upper row) and after (lower row) applying the *Confidence-Weighted Mean Shift* refinement procedure. The transparency of the detected circles in the upper row images correspond to their confidence expressed with the scaled number of votes picked from the Hough voting space.

3 Tracking

To recognize traffic signs from a moving vehicle, it is crucial to have a view-independent object detector. Training such a detector directly exhibits serious difficulties as it requires feature descriptors to be both: highly discriminative and pose-invariant. Our method of solving this problem follows a different strategy and has been shown successful in [16]. Instead of devising a pose-independent feature representation of the target

objects, we learn the application-specific motion model and integrate it with the existing pose-dependent object detector to make it pose-independent.

3.1 Tracking as a Regression Problem

Let $\mathbf{M}$ be an affine matrix that transforms a unit square at the origin in the object coordinates to the affine region enclosing the target object in the image coordinates, and let $\mathbf{M}^{-1}$ be an inverse transform, as shown in Fig. 2.

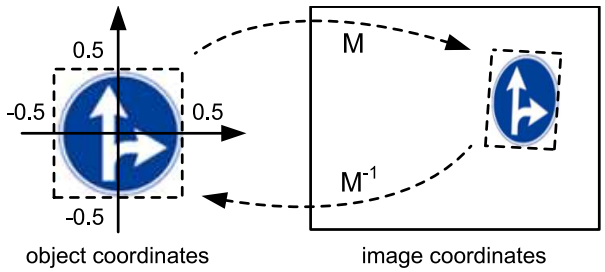


Figure 2: Affine transformation matrix and its inverse.

Our goal is to estimate the transformation matrix $\mathbf{M}_t$, given the observed images up to time t, $I_{0,\ldots,t}$, and the initial transformation $\mathbf{M}_0$. $\mathbf{M}_t$ is modeled recursively:

$$\mathbf{M}_t = \mathbf{M}_{t-1}\Delta\mathbf{M}_t \quad , \tag{3}$$

which means that it is sufficient to estimate only the increment $\Delta\mathbf{M}_t$ corresponding to the motion of the target from time $t-1$ to t in object coordinates. It is determined by an affine, matrix-valued regression function $f : \mathbb{R}^m \longmapsto A(2)$:

$$\Delta\mathbf{M}_t = f\left(\mathbf{o}_t(\mathbf{M}_{t-1}^{-1})\right) \quad , \tag{4}$$

where $\mathbf{o}_t(\mathbf{M}_{t-1}^{-1})$ denotes an image descriptor applied to the previously observed image, after mapping it to the unit rectangle.

The regression function $f : \mathbb{R}^m \longmapsto A(2)$ is an affine matrix-valued function, where $A(2)$ denotes a two-dimensional affine transformation. To learn its parameters, it is necessary to know the initial pose of an object, $\mathbf{M}_0$, and the image I_0 at time t_0. Training examples are generated as pairs $(\mathbf{o}_0^i, \Delta\mathbf{M}_i)$, where $\Delta\mathbf{M}_i$ are random deformation matrices around identity and $\mathbf{o}_0^i = \mathbf{o}_0(\Delta\mathbf{M}_i^{-1}\mathbf{M}_0^{-1})$. The optimal parameters of f are derived on the grounds of the Lie group theory by minimizing the sum of the squared geodesic distances between the pairs of motion matrices: estimated $f(\mathbf{o}_0^i)$, and known $\Delta\mathbf{M}_i$. Details of this method can be found in [16].

3.2 Tracker Architecture

For the task of road sign tracking we utilize the above technique in the way outlined in Fig. 3. Once a candidate sign has been detected for the first time, an instance-specific tracker is initialized with the region corresponding to the bounding rectangle of the found circle, assuming no distortion. At this point a small number of random deformations are generated from the observed image and used for instant training. A map of 6×6 regularly spaced 6-bin gradient orientation histograms is used as an object descriptor. The trained tracker is employed to detect the sign in n subsequent frames, each being used to generate and enqueue m new random deformations.

As in a realistic traffic scenario the scene is often difficult and changes fast, accuracy of the tracker is likely to deteriorate very quickly as a result of contaminating the training examples with the unwanted background fragments. As a remedy, we update the tracker's regression function after each n frames by retraining it on the collected portion of $n \cdot m$ training examples. The updated tracker is then used to re-estimate the pose of the observed sign and the space is allocated for a new portion. Such a periodic update scheme allows us to recover from the misalignments that are likely to occur during the sign tracking. Finally, the track is assumed to be lost when the sign either gets out of the camera's viewfield or when the normalized cross-correlation (NCC) between its current object-coordinate image and the same image recorded at the last update drops below a predefined threshold. The period for which the correlation is below this threshold is extended to several frames to prevent instant track losing due to the short-term occlusions. During this period the last reliable motion matrix estimate, recorded before the NCC fell below the acceptable minimum, is restored and used.

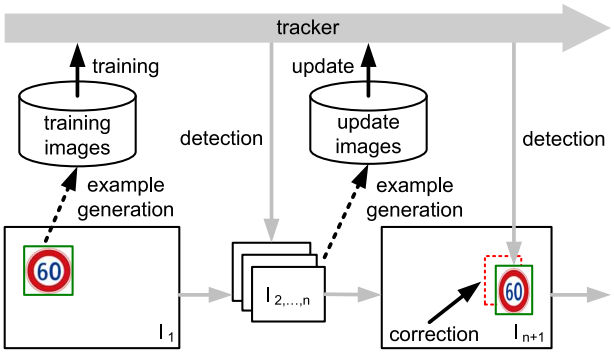


Figure 3: Operation of the proposed road sign tracker over time. The period between the initial training and the first update inclusive is shown.

4 Sign recognition

Recognition of traffic signs is a hard multi-class problem with an additional difficulty caused by the fact of certain signs being very similar to one another. The approach we have adopted, *one-versus-all* (OvA), assumes existence of a single separator between each class and all other classes. It is implemented using a *winner-takes-all* strategy that associates a real-valued score with each class. An example belongs to the class which assigns it the highest score.

Formally, our classifier, $F(\mathbf{x})$, recognizes only two classes: "same" and "different", and is trained using pairs of images, i.e. $\mathbf{x} = (i_1, i_2)$. The pairs representing the same sign type are labeled $y = 1$ (positive), and the pairs representing two different types are labeled $y = -1$ (negative). Real-valued discriminant function F is learned using a modified AdaBoost algorithm [4]. We define F as a sum of image features f_j:

$$F(i_1, i_2) = \sum_{j=1}^{N} f_j(i_1, i_2) \ . \quad (5)$$

Each feature evaluates to:

$$f_j(i_1, i_2) = \begin{cases} \alpha & \text{if } d(\phi_j(i_1), \phi_j(i_2)) < t_j \\ \beta & \text{otherwise} \end{cases} \ , \quad (6)$$

where ϕ_j is a filter defined over a chosen class of image descriptors, d is a generic distance metric that makes sense for such descriptors, and t_j is a feature threshold. Let us denote by W_+^+ the total weight of these positive examples that are labeled positive by the weak classifier (true positives), and by W_+^- the total weight of those that are labeled negative (false negatives). By analogy, let W_-^- and W_-^+ be the total weight of true negatives and false positives respectively. In each boosting round the filter ϕ_j and the threshold t_j are selected so as to minimize the weighted error of the training examples:

$$e_j = W_+^- + W_-^+ \ . \quad (7)$$

The optimal values of α and β are found based on the Schapire and Singer's criterion [5] of minimizing:

$$Z = \sum_{k=1}^{M} w_k e^{-y_k f(x_k)} \ , \quad (8)$$

where M is the total number of training examples. Through several transformations it can be shown that

$$Z = W_+^+ e^{-\alpha} + W_+^- e^{-\beta} + W_-^+ e^{\alpha} + W_-^- e^{\beta} \ . \quad (9)$$

Taking partial derivatives of Z with respect to α and β and setting each to zero determines the optimal values of each parameter to set in a given boosting round:

$$\alpha = \tfrac{1}{2} \log \left(\frac{W_+^+}{W_-^+} \right) \quad \beta = \tfrac{1}{2} \log \left(\frac{W_+^-}{W_-^-} \right) \ . \quad (10)$$

AdaBoost yields a classification decision:

$$l(i_1, i_2) = \text{sgn} F(i_1, i_2) = \text{sgn} \left(\sum_{j=1}^{N} f_j(i_1, i_2) \right) \ . \quad (11)$$

By omitting sign, value of the right-hand-side term can be treated as a degree of similarity of the input images. If one of those images, say i_1, is a prototype of known class k ($i_1 = p_k$), our road sign classifier assigns such a label to the other, unknown image, that satisfies:

$$l(i) = \arg\max_k F(p_k, i) \ . \quad (12)$$

In other words, $l(i)$ is determined from the prototype to which the tested image is the most similar.

To classify a sequence of images, $i_{1,\ldots,t}$, the maximum rule in (12) is applied to the weighted sum of $F(p_k, i_t)$ terms over all images i_t:

$$l(\mathbf{I}_{1,\ldots,T}) = \arg\max_k \sum_{t=1}^{T} q(t) F(p_k, i_t) \ . \quad (13)$$

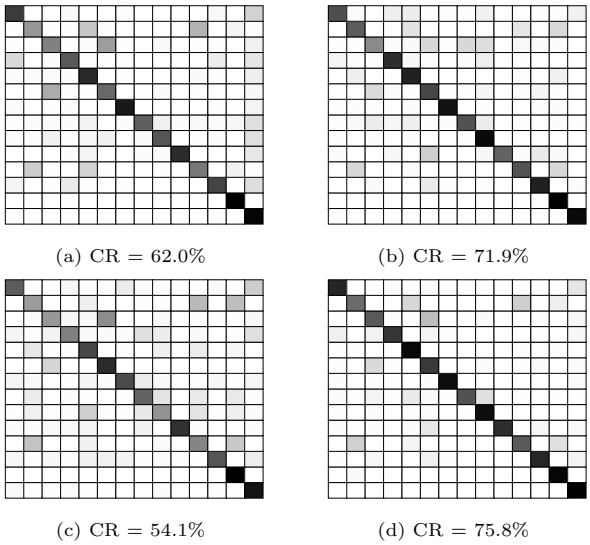
Each i_t denotes an image of a sign in object coordinates, obtained by applying the inverse of the transformation matrix $\mathbf{M}_t$ to the frame at time t. In other words, the possible geometrical deformation of a sign is first compensated by obtaining the full-face view of the target. The resulting image is passed on input of the classifier. Temporal weights $q(t)$ are designed to attach more importance to the most recent observations, which are by definition clearer and hence less ambiguous. Specifically, $q(t)$ is given by:

$$q(t) = b^{t_{last} - t} \ , \quad (14)$$

where t_{last} is the time point when the sign is for the last time seen and $b < 1$ is tuned experimentally from an independent set of image sequences.

5 Experimental Results

We have evaluated our algorithms in the two experiments involving video sequences captured in a cluttered urban traffic environment from a moving vehicle. In the first experiment a sole classifier was tested using 8745 static images of 14 Japanese road signs cropped from these sequences. Each input image contained a single aligned sign and the quality and illumination of the images varied significantly. When constructing the test input pairs, the prototype images were chosen randomly for each class. Exploiting flexibility of the distance measure in (6), three different image descriptors were used within the AdaBoost framework to populate the pool of input features. Obtained results are shown in the confusion matrices in Fig. 4.



(a) CR = 62.0% (b) CR = 71.9%

(c) CR = 54.1% (d) CR = 75.8%

Figure 4: Prediction accuracy of a 100-feature boosted classifier trained using different image descriptors: (a) color-parametrized Haar wavelets [11], (b) histograms of oriented gradients (HOG), (c) 4×4 covariance matrices encoding x and y coordinates and first-order image derivatives [14], (d) Haar and HOG features jointly.

In the second experiment we tested the entire system directly on several image sequences using the best-performing classifier from the static experiment. The minimum radius of the circles captured by the detector was set to 10 pixels and the tracker updated itself every

$n = 10$ frames, generating $m = 6$ new random transformations in each frame. Table 1 illustrates the classification results obtained. As seen, an overall error rate of the classifier did not exceed 15%. Misclassifications were mainly caused by the motion blur erasing relevant image gradients, and by the cumulated reconstruction errors of the tracker. Regarding the other system components, the refined Hough circle detector appeared to be relatively accurate and resistant to clutter. Overall, it missed 12 true signs, mostly due to the insufficient figure-background contrast, and yielded only a few false sign candidates.

Finally, the tracker demonstrated its ability to rapidly compensate small affine sign distortions, which enabled more accurate recognition and real-time system operation [2]. In Fig. 5 screenshots from example synthetic and real image sequences are shown where the detected and tracked traffic sign is marked and the corresponding full-face view of this sign in object coordinates is provided for comparison.

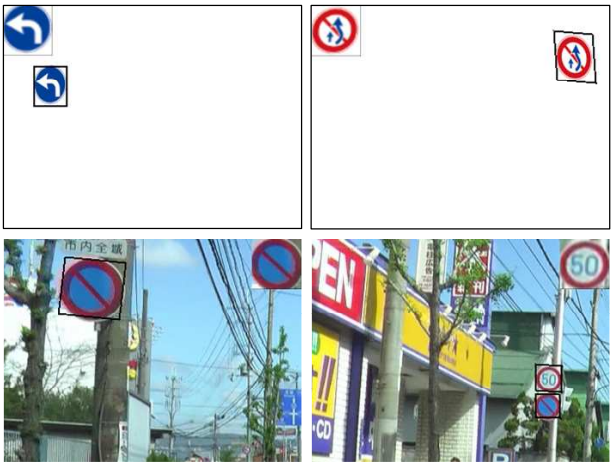


Figure 5: Screenshots from example synthetic and real image sequences where a detected and tracked sign in image coordinates is shown together with its reconstructed frontal view in object coordinates.

Table 1: Classification accuracy in the dynamic experiment. The numbers of correctly classified signs of each class are given against the total numbers of such signs detected in the input sequences.

6/6	4/4	1/1	3/4	1/2	10/10	5/9	2/2	8/9	25/29

6 Conclusions

In this paper we have presented a comprehensive approach to traffic sign detection, tracking and recognition using a car-mounted camera. Our method features a fast and robust tracker that effectively corrects the affine distortions the road signs are subject to. It is learned and periodically updated on the fly using the Lie algebra of the motion group. A detected sign candidate is classified by maximizing its similarity to the class's prototype image. This similarity is estimated by a linear combination of local image similarities and is learned from image pairs using a novel variant of AdaBoost algorithm. Robustness of the traffic sign detector, tracker and classifier is demonstrated in the static and dynamic recognition experiments. In the latter, our prototype implementation is shown to capture and correctly classify most road signs in real time.

[2] Sample videos available at: `http://people.brunel.ac.uk/~cspgaar/MVA2009/`

References

[1] Duda, R.O. and Hart, P.E.: "Use of the Hough transformation to detect lines and curves in pictures", *Communications of the ACM*, vol.15, no.1, pp.11-15, 1972.

[2] Piccioli, G. and De Micheli, E. and Parodi, P. and Campani, M.: "A robust method for road sign detection and recognition", *Image and Vision Computing*, vol.14, no.3, pp.209-223, 1996.

[3] de la Escalera, A. and Moreno, L. E. and Salichs, M. A. and Armingol, J. M.: "Road traffic sign detection and classification", *IEEE Transactions on Industrial Electronics*, vol.44, no.6, pp.848-859, 1997.

[4] Freund, Y. and Schapire, R.E.: "A short introduction to boosting", *Journal of Japanese Society for Artificial Intelligence*, vol.14, no.5, pp.771-780, 1999.

[5] Schapire, R. E. and Singer, Y.: "Improved boosting algorithms using confidence-rated predictions", *Machine Learning*, vol.37, no.3, pp.297-336, 1999.

[6] Miura, J. and Kanda, T. and Shirai, Y.: "An active vision system for real-time traffic sign recognition", In *Proc. of the IEEE Conference on Intelligent Transportation Systems*, pp.52-57, 2000.

[7] Paclík, P. and Novovicova, J. and Pudil, P. and Somol, P.: "Road Sign Classification using the Laplace Kernel Classifer", *Pattern Recognition Letters*, vol.21, no.13-14, pp.1165-1173, 2000.

[8] Comaniciu, D. and Meer, P.: "Mean shift: a robust approach towards feature space analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.24, no.5, pp.603-619, 2002.

[9] Fang, C-Y. and Chen, S-W. and Fuh, C-S.: "Road-Sign Detection and Tracking", *IEEE Transactions on Vehicular Technology*, vol.52, no.5, pp.1329-1341, 2003.

[10] Viola, P. and Jones, M.: "Robust Real-time Face Detection", *International Journal of Computer Vision*, vol.57, no.2, pp.137-154, 2004.

[11] Bahlmann, C. and Zhu, Y. and Ramesh, V. and Pellkofer, M. and Koehler, T.: "A System for Traffic Sign Detection, Tracking and Recognition Using Color, Shape, and Motion Information", In *Proc. of the IEEE Intelligent Vehicles Symposium*, pp.255-260, 2005.

[12] Loy, G. and Barnes, N. and Shaw, D. and Robles-Kelly, A.: "Regular Polygon Detection", In *Proc. of the 10th IEEE International Conference on Computer Vision*, vol.1, pp.778-785, 2005.

[13] Paclík, P. and Novovicová, J. and Duin, R. P. W.: "Building Road-Sign Classifiers Using a Trainable Similarity Measure", *IEEE Transactions on Intelligent Transportation Systems*, vol.7, no.3, pp.309-321, 2006.

[14] Tuzel, O. and Porikli, F. and Meer, P.: "Region covariance: A fast descriptor for detection and classification", In *Proc. of the 9th European Conference on Computer Vision*, pp.589-600, 2006.

[15] Ruta, A. and Li, Y. and Liu, X.: "Towards Real-time Traffic Sign Recognition by Class-specific Discriminative Features", In *Proc. of the 18th British Machine Vision Conference*, vol.1, pp.399-408, 2007.

[16] Tuzel, O. and Porikli, F. and Meer, P.: "Learning on Lie Groups for Invariant Detection and Tracking", In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.1-8, 2008.