Robust License Plate Detection Using Covariance Descriptor in a Neural Network Framework

Fatih Porikli
Mitsubishi Electric Research Labs

Tekin Kocak
Polytechnic University

Abstract

We present a license plate detection algorithm that employs a novel image descriptor. Instead of using conventional gradient filters and intensity histograms, we compute a covariance matrix of low-level pixel-wise features within a given image window. Unlike the existing approaches, this matrix effectively captures both statistical and spatial properties within the window. We normalize the covariance matrix using local variance scores and restructure the unique coefficients into a feature vector form. Then, we feed these coefficients into a multi-layer neural network. Since no explicit similarity or distance computation is required in this framework, we are able to keep the computational load of the detection process low. To further accelerate the covariance matrix extraction process, we adapt an integral image based data propagation technique. Our extensive analysis shows that the detection process is robust against noise, illumination distortions, and rotation. In addition, the presented method does not require careful fine tuning of the decision boundaries.

1 Introduction

Even though license plate detection systems started emerging in late 80s [9], only recently it became a mainstream computer vision application by gaining popularity in security and traffic monitoring systems. Often the extracted information is used for enforcement, access-control, and flow management, e.g. to keep a time record for automatic payment calculations or to fight against crime. Still, robust detection of different types of license plates in varying poses, changing lighting conditions and corrupting image noise without using an external illumination source presents a challenge.

Existing approaches diversify from rule based deterministic methods [5]-[4] to more elegant training based classifiers [6]-[2]. A simple approach is based on the detection of the license plate boundaries [5]. The input image is first processed to amplify the edge information using a customized gradient filter. Then, Hough transformation is applied to detect parallel line segments. Coupled parallel lines are considered as license plate candidates. Another approach uses gray level morphology [4]. This approach focuses on local appearance properties of license plate regions such as brightness, symmetry, orientation, etc. Candidate regions are compared with a given license plate image based on the similarity of these properties.

Classifier based methods learns different representations of the license plates. In a color texture based approach [6], a license plate region is assumed to have discriminatory texture properties, and a support vector machine (SVM) classifier is used to determine whether a candidate region corresponds to a license plate or not. Only the template of a region is fed directly to the SVM to decrease the dimensionality of the representation. Next, LP regions are identified by applying a continuously adaptive mean-shift algorithm to the results of the color texture analysis. A recent algorithm [2] imposes the detection task as boosting problem. Over several iterations, the AdaBoost classifier selects the best performing weak classifier from a set of weak ones, each acting on a single feature, and, once trained, combines their respective votes in a weighted manner. This strong classifier is then applied to sub-regions of an image being scanned for likely license plate locations. An optimization based on a cascade of classifiers, each specifically designed using the false positive and false negative rates, helps to accelerate the scanning process. In addition to single frame detection techniques, there exists methods that take advantage of video data [8] by processing multiple frames at the same time.

One main drawback of all the above methods is that their performance highly depend on the strong assumptions they made on the appearance of the license plates. Most methods cannot handle in-plane and out-plane rotations, and incapable of compensating imaging noise and illumination changes. Enlarging the training dataset with rotated training samples often deteriorates performance and increases false positive rate.

Many different image representations, from aggregated statistics to textons to appearance models, have been used

Proceedings of the IEEE International Conference
on Video and Signal Based Surveillance (AVSS'06)
0-7695-2688-8/06 $20.00

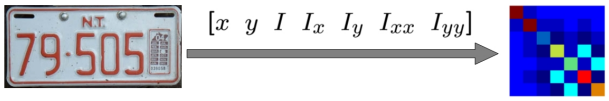

Figure 1. Covariance matrix generated for 7 features.

for license plate detection. As a general rule, a region descriptor should be invariant to geometric and radiometric distortions but competent enough to distinguish license plates from the background clutter under uncontrolled conditions. Histograms are popular representations of image statistics. They are easy to compute, however, they disregard the spatial arrangement of feature values and do not scale efficiently to higher dimensions. Appearance models provide spatial discrimination, but they are highly sensitive to the pose, scale and shape variations. Texture representations are often computationally expensive and scale dependent.

We developed a feature covariance matrix descriptor that captures not only the appearance but also the statistical properties of image regions. This descriptor has low dimensionality in comparison to many other approaches and it is invariant to in-plane rotations. In general, a single covariance matrix extracted from a region is sufficient to match the region in different views and poses. Covariance matrix of any region has the same size, thus it enables comparing any regions without being restricted to a constant window size.

In the following sections, we explain how we construct the covariance matrix and train the neural network classifier. Then, we give several examples of license plate detection, present performance graphs and comparison results with a state-of-art weighted orientation histogram approach.

2 Covariance Descriptor

We denote one dimensional, unit normalized intensity image as I. The method can also be generalized to other type of images, e.g. multi-spectral. Let F be the $M \times N \times d$ dimensional feature image extracted from I as

$$F(x,y) = \Phi(I,x,y) \tag{1}$$

where the function Φ can be any pixel-wise mapping such as color, image gradients $I_x, I_{xx}, ..$, edge magnitude, edge orientation, filter responses, etc. This list can be extended by including higher order derivatives, texture scores, radial distances, angles and temporal frame differences in case video data is available.

For a given rectangular window $W \subset F$, let $\{\mathbf{f}_k\}_{k=1..n}$ be the d-dimensional feature vectors inside W. Each feature vector $\mathbf{f_k}$ represents a pixel (x,y) within that window. Since we will extract the mutual covariance of the features, the windows can actually be any shape not necessarily rectangles. Basically, covariance is the measure of how much two variables vary together. We represent each window W with a $d \times d$ covariance matrix of the features

$$\begin{aligned} \mathbf{C}_W &= \begin{bmatrix} c_W(1,1) & \cdots & c_W(1,d) \\ \vdots & \ddots & \\ c_W(d,1) & & c_W(d,d) \end{bmatrix} \\ &= \frac{1}{n-1}\sum_{k=1}^{n}(\mathbf{f_k}-\mu)(\mathbf{f_k}-\mu)^{\mathbf{T}} \end{aligned} \tag{2}$$

where μ is the mean vector of all features. The diagonal coefficients represent the variance of the corresponding features. For example, the i^{th} diagonal element represents the variance for the i^{th} feature we measure. The off-diagonal elements represent the covariance between two different features. Fig. 1 shows a sample covariance matrix for a given image.

We construct the feature vector f_k using two types of mappings; spatial features that are the functions of the pixel coordinates, and appearance attributes, i.e., color, gradient, etc., that are obtained from the pixel color values. As spatial features, the coordinates may be directly associated with the appearance features;

$$\mathbf{f}_k = [x \;\; y \;\; I(x,y) \;\; I_x(x,y) \;\; ... \,] \tag{3}$$

or using polar coordinates

$$\mathbf{f}_k = [r(x',y') \;\; \theta(x',y') \;\; I(x,y) \;\; I_x(x,y) \; ... \,] \tag{4}$$

where

$$(x',y') = (x - x_0, y - y_0) \tag{5}$$

are the relative coordinates with respect to window center (x_0, y_0), and

$$r(x',y') = (x'^2 + y'^2)^{\frac{1}{2}} \tag{6}$$

is the distance from the (x_0, y_0) and

$$\theta(x',y') = \arctan\left(\frac{y'}{x'}\right) \tag{7}$$

is the orientation of the pixel location.

Note that, using Euclidean coordinates makes the covariance descriptor strictly rotation variant. Even though using the distance r from the window center as the coordinate feature provide rotation invariance, it destroys the connectivity along the lines passing through the origin. In other words, polar coordinates is blind towards the pattern modifications in case such modifications are on the same radial circle as illustrated in Fig. 2.

Proceedings of the IEEE International Conference
on Video and Signal Based Surveillance (AVSS'06)
0-7695-2688-8/06 $20.00 © 2006 IEEE

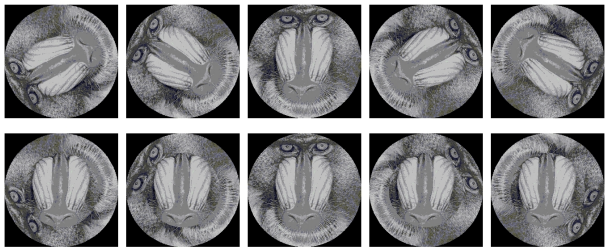

Figure 2. Top: rotation, **bottom** distortion (only outer rim is rotated).

To make covariance representation rotation invariant, we use a frequency transform function g of coordinates as

$$\mathbf{f}_k = [g(x',y') \;\; r(x',y') \;\; I(x,y) \;\; I_x(x,y) \; ... \;] \quad (8)$$

We define the frequency transform function g as

$$g(x',y') = e^{i\left(2\pi \frac{r(x',y')}{r_{max}} + \theta(x',y')\right)} \quad (9)$$

where r_{max} is a normalizing constant corresponding to the maximum radius of the window W. Since the frequency transform feature gives complex covariances, we take the magnitude of the covariance coefficients when we construct the covariance matrix.

We computed the covariance distances for the rotated and distorted images in Fig. 2 where the top row corresponds to rotated images and the bottom row shows the images that only the pixels at the same radial distance are rotated. The top row in Fig. 2 shows the distances of the rotated covariance matrices constructed using the frequency transform and the bottom row corresponds to polar coordinates. As visible, the radially symmetric feature r generates almost identical results for the rotated and distorted images, In other words, it fails to differentiate the distortion from rotation even though it is rotation invariant. On the other hand, the covariance responses of the frequency transform feature changes according to the amount of the distortion. In addition, it is rotation invariant.

3 Detection Algorithm

We impose the license plate detection as a classifier based binary recognition problem. We adapt an off-line trained neural network to determine whether a given image region corresponds to a license plate or not.

In training phase, we compute pixel-wise image features including spatial gradients and moments. We model each positive (license plate) and negative (non-license plate) training image via a covariance descriptor introduced in [10] and feed this information into a neural network with

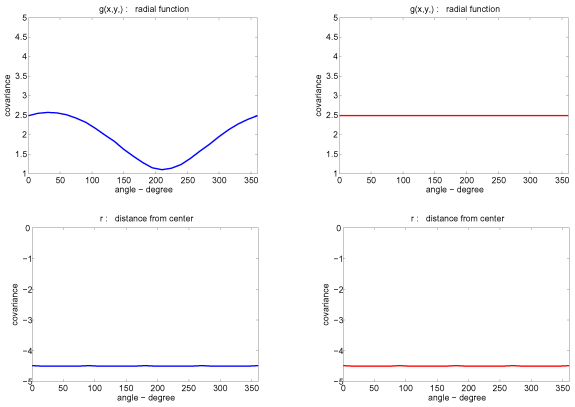

Figure 3. Top: rotation, **bottom** distortion.

+1,-1 labels respectively. We employ a feed-forward back-propagation type of neural network with three internal layers. Neural networks are made of nodes that their state can be described by activation values. Each node generates an output signal based on its activation. Nodes are connected to each other very specifically, each connection having an individual weight. Each node sends its output value to all other nodes to which they have an outgoing connection. The node receiving the connections calculates its activation by taking a weighted sum of the input signals. In other words, each node acts as a single perceptron that has a linear decision surface. The output is determined by the activation function based on this activation. Networks learn by changing the weights of the connections. We use a three layer network to impose nonlinear decision boundaries while preventing from overfitting to the training data. The number of inputs is same as the number of unique coefficients. Each layer has a weight matrix, a bias vector, and an output vector. A graph of this network is shown in Fig. 4.

In detection phase, we scan the image at different scales and test each scanning window whether it corresponds to a license plate or not. Normally, such an exhaustive search would be slow. Thus, we adapt an integral image based fast covariance matrix extraction method [7] to achieve the search in linear time. We restructure the unique coefficients of the covariance matrix into a vector form and send this vector to the trained neural network to compute the activa-

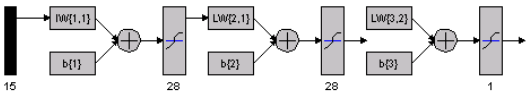

Figure 4. A three-layer, 28-input feed-forward back-propagation neural network used for all features.

Proceedings of the IEEE International Conference
on Video and Signal Based Surveillance (AVSS'06)
0-7695-2688-8/06 $20.00 © 2006 IEEE

Figure 5. Sample detection result by scanning windows.

tion function score. For example, for a 7×7 covariance vector, there are 28 unique coefficients. The sign of the score indicates the estimated class, e.g. positive for license plates and negative for non-license plates. The network uses a non-linear sigmoid transfer function, which calculates its output as

$$tansig(z) = \frac{2}{1 + e^{-2z}} - 1 \qquad (10)$$

to make sure that the output remains within a specified range, which is $[-1, 1]$. A sample detection result is given in Fig. 5. This detection is done by scanning each image using a fixed size window. For unknown license plate size, the scanning can be executed in multiple image resolutions keeping the minimum detection window constant.

We want to emphasize that the covariance matrices do not lie in Euclidean space, thus, their distance can not be computed by simple matrix subtraction. Instead, it is required to find the generalized eigenvalues of two matrices [3]. The proposed neural network based framework successfully eliminates this costly operation by converting problem into comparison of separate covariance values structures into a feature vector form.

4 Experiments

We used only a sparse set of positive examples, 300 license plate samples, and 3000 non-license plate samples to train the neural network. We assessed the performance using non-overlapping datasets consist of 300 positive and 3000 negative images. We aimed to reflect the real-life condition that the database is unbalanced. Sample positive training images are shown in Fig. 6.

We tested different feature combinations for the covariance descriptor as given in Table 1. To make a fair evaluation, we compared our results with the weighed orientation histogram descriptor that is proven to outperform other features in human detection [1]. This descriptor computes an orientation histogram using pixels gradient orientations. The bin value is incremented by the gradient magnitude. The histogram is then assigned as the feature vector. In addition, we collected the gradient responses to each image in

Figure 6. Positive samples (gray level images are used).

several bins, and filtered out the desired information in respective bins to focus mainly on the vertical and horizontal features using a set of positive samples.

In all cases, we used the same dimensionality, i.e. 28 unique coefficients in covariance descriptors and 28 bins for orientation histograms. We computed feature vectors for each sample and feed these features into the same neural network to find out which method is more efficient for the same set of samples.

The training and test data consist of manually extracted 105×32 license plate images captured from a camera mounted on a street lamp pole overlooking a stop sign. We observed that with the higher resolution data the performance of the orientation histogram method remains same. Whereas, its performance severely degrades as the resolution of the samples becomes smaller. Thus, histogram approach is not suitable for low-resolution data.

Figure 7 shows the performances of the covariance descriptors and orientation histogram. These ROC curves obtained from the test data by applying thresholds in the range $[-0.9, 0.9]$ to the likelihood score computed by the neural network. As visible, the covariance features with the same dimensionality produce much better detection results than the orientation histogram method. Even though its dimension is almost half i.e. 15 vs. 28, the feature vector obtained from 5×5 covariance descriptor gives as similar results as the orientation histogram. Moreover, orientation histogram method is highly sensitive to the selected likelihood threshold value of the neural network; the range of the true positive and true negatives varies significantly as the threshold changes. As shown, all of the tested covariance descriptors provided comparably more consistent results indicating they are less sensitive to the selected threshold value.

Table 1. List of Features

Features	Number
$g, r, I, \lvert I_x\rvert, \lvert I_y\rvert, \lvert I_{xx}\rvert, \lvert I_{yy}\rvert$	C_1
$x, y, I, \lvert I_x\rvert, \lvert I_y\rvert, \lvert I_{xx}\rvert, \lvert I_{yy}\rvert$	C_2
$x, y, I, \lvert I_x\rvert, \lvert I_y\rvert$	$C_{5\times5}$
$histogram$	H_ϕ

Proceedings of the IEEE International Conference
on Video and Signal Based Surveillance (AVSS'06)
0-7695-2688-8/06 $20.00 © 2006 IEEE

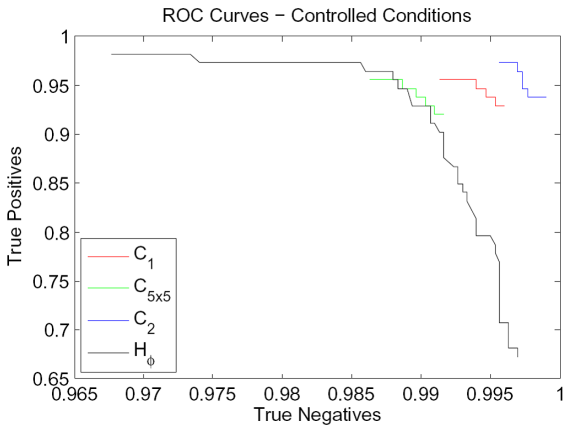

Figure 7. Performance graphs under controlled lighting and noise free conditions for accurately aligned images.

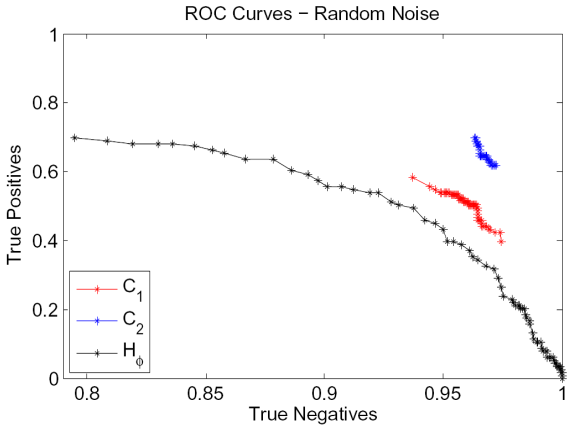

Figure 8. ROC curves for added noise $N(0, 0.04)$.

In another simulation, we contaminated the training and test data with Gaussian noise $N(0, 0.04)$, i.e. zero-mean, 0.04 variance, which is a severe contamination. The ROC curves are given in Fig. 8. As seen, the true positive rates are significantly decreased for all descriptors, however, covariance descriptors still provided more accurate results.

We tested the sensitivity for in-plane rotations in Fig. 9. We randomly rotated both training and tested images within the $[-20, 20]$ degrees and retrained the neural network. We observed degradation of the performance for the orientation histogram H_phi and frequency transform covariance descriptor C_2. Persistently, the covariance descriptor using Euclidean coordinates C_1 gave identical results by outperforming the other methods. One possible explanation is that the license plates are composed of characters and the rotation of the training data, in a way, corresponds to the shuffling of these characters around the window centers by keeping the covariance response of C_1 almost intact.

To analyze the robustness against the illumination

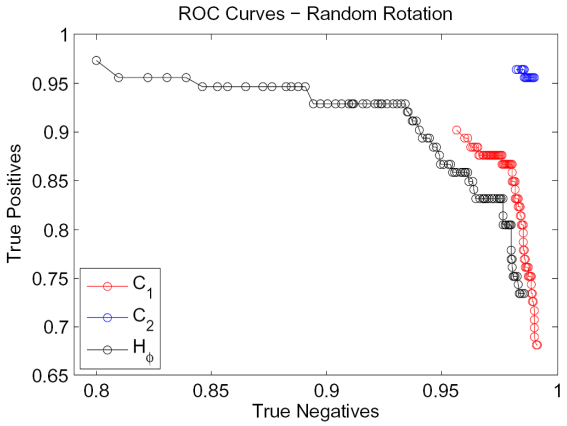

Figure 9. ROC curves for random rotation in the range of $[-20, 20]$ degrees around the window center. In other words, the training and test images are not aligned.

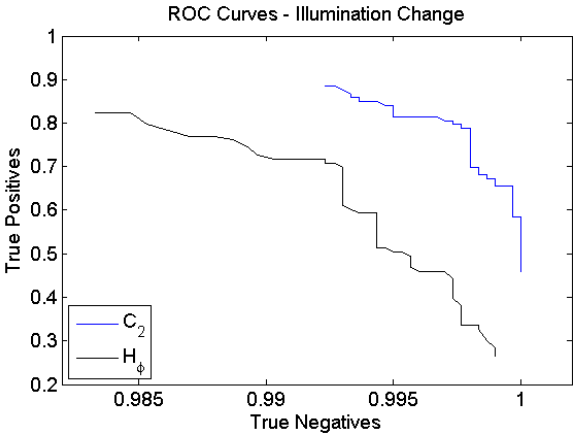

Figure 10. Intensity values of the test images are distorted.

changes, we distorted the intensity values of the test samples and applied them to the original neural networks that were trained by the original data. By doing this, we wanted to simulate the real-life conditions where the intensity of the acquired images may vary due to the external lighting conditions. Even though we compensate for the illumination changes when we construct the orientation histogram, it still failed to achieve its performance when it was tested with the undistorted data as shown in Fig. 10. Yet, the covariance descriptor is proven to less sensitive towards the illumination changes due to its intrinsic property that it doe not use the intensity mean but the intensity variance. This enables the covariance descriptor to automatically normalize for the varying illumination conditions. To provide a comprehensive comparison, we also show the above results in the Fig. 11.

As a final test, we tested the performance of the covari-

Proceedings of the IEEE International Conference
on Video and Signal Based Surveillance (AVSS'06)
0-7695-2688-8/06 $20.00 © 2006 IEEE

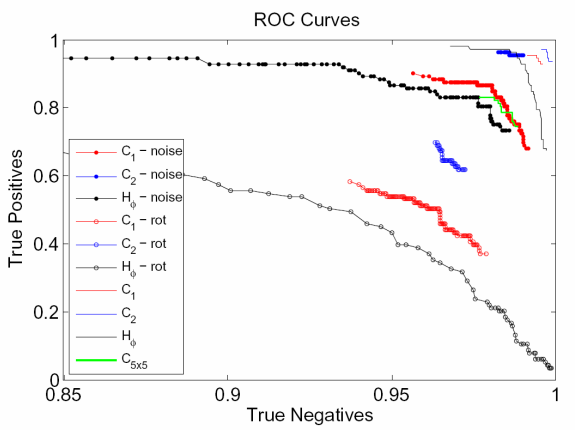

Figure 11. ROC curves of various cases are shown altogether.

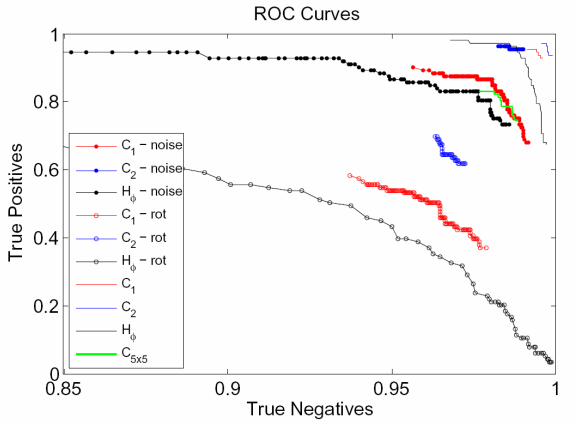

Figure 12. Comparison of C_1 and C_2.

ance descriptors C_1 and C_2 that contains the Euclidean coordinate an frequency transform features respectively. We randomly rotated only the test images and applied these images to the original neural networks of trained for these descriptors. As shown in Fig. 12, the frequency transform feature provided much better results. This indicates that the frequency transform is a more competent representation in cases that the amount of the rotation is not embedded in the training data, which is what generally happens in actual applications.

The detection time of each frame depends on the number of candidate windows scanned and tested. Our current implementation runs at 100,000 windows per second, which is suitable for most real-time detection applications. Furthermore, the integral image based covariance construction [7] significantly accelerates the extraction process. Computation of all 7×7 covariance matrices takes only 20msec for a 640×480 image.

5 Conclusions

We presented a license plate detection method based on a novel descriptor. The covariance descriptor effectively captures both spatial and statistical properties of target patterns. The covariance representation is robust against in plane rotations of license plates, and severe image noise. Since the mean is filtered, it is almost invariant to severe illumination changes. Our tests show that the neural network framework provides adaptability to out plane rotations as well. Once the training is done, the detection process is very fast. We compared our algorithm with a state-of-art detection method based on the weighted histograms of orientation. Our results show that the proposed method outperforms this approaches in all cases.

References

[1] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* San Diego, CA, 2005. 4

[2] L. Dlagnekov. Car license plate, make, and model recognition. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* San Diego, CA, 2005. 1

[3] W. Förstner and B. Moonen. A metric for covariance matrices. Technical report, Dept. of Geodesy and Geoinformatics, Stuttgart University, 1999. 4

[4] J. Hsieh, S. Yu, and Y. Chen. Morphology based license plate detection from complex scenes. 2002. 1

[5] V. Kamat and S. Ganesan. An efficient implementation of the hough transform for detecting vehicle license plates using dsp. In *Proceedings of Real-Time Technology and Applications*, pages 58–59, 1995. 1

[6] K. Kim, K. Jung, and J. Kim. Color texture-based object detection: an application to license plate localization. In *Lecture Notes in Computer Science, Springer*, pages 293–309, 2002. 1

[7] F. Porikli and O. Tuzel. Fast extraction of covariance matrices using integral images. In *Proceedings of Int'l Conference on Image Processing, ICIP*, 2006. 3, 6

[8] C. Rahman, W. Badawy, and A. Radmanesh. A real time vehicle license plate recognition system. In *Proceedings of the IEEE on Advanced Video and Signal Based Surveillance, AVSS03*, 2003. 1

[9] V. Sgurev, G. Gluhchev, and D. Mutafov. Recognition of car registration numbers. In *Problems of Engineering Cybernetics and Robotics*, volume 27, pages 79–85, 1987. 1

[10] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. In *Proc. 9th European Conf. on Computer Vision,* Graz, Austria, volume 2, pages 589–600, 2006. 3