Effective License Plate Detection Using Fast Candidate Region Selection and Covariance Feature based Filtering

Bo-Yuan Feng, Mingwu Ren
School of Computer Science and Engineering, Nanjing Univ. Science & Technology, China
fengboyuannj@gmail.com

Xu-Yao Zhang, Cheng-Lin Liu
NLPR, Institute of Automation, Chinese Academy of Sciences, China
{xyz, liucl}@nlpr.ia.ac.cn

Abstract

This paper presents a new real-time license plate detection method aiming for fast and accurate detection in live videos. Compared with the previous learning based detection schemes which scan multi-scale images with sliding window, our method takes a cascaded scheme. In the first stage, candidate plate regions are detected based on edge density in reduced image of very low resolution for guaranteeing high speed. In the second stage, the candidate regions are verified using a linear SVM classifier with covariance features for high accuracy. Experimental results on two datasets collected from practical traffic surveillance videos indicate the robustness of our method, which is relatively invariant to scaling, rotation, blurring and illumination. This method takes only 10 msec for detection on a 768 × 576 image.

1. Introduction

Automatic recognition of car license plates plays an important role in real-life applications, such as traffic video surveillance, parking lot access-control, automatic toll payment and vehicle tracking [5]. License plate detection is the first and very important stage in the license plate recognition system. However, the size variation, perspective distortion and uneven illumination render the license plate detection a challenging task.

Many papers have reported promising results in the field of license plate extraction. Zheng et al. [18] proposed a fast detection method based on edge distribution. In [19] and [1], texture features extracted by vector quantization (VQ) and sliding concentric windows (SCW) are utilized to locate license plates. The methods of [12,13,15,18] detect plates by taking advantage of the color information. However, the variable lighting conditions of imaging makes the color based methods unstable [2]. In recent years, learning and classification based methods have increasingly applied to license plate detection. The authors of [7,9,16] employ artificial neural networks (ANNs) to locate plates by scanning the entire image with a sliding window. Zhang et al. [17] presented an Adaboost detection framework with both global statistical and local Haar-like features. A fast region covariance descriptor [13] has bee successfully applied to license plate detection by multi-stage neural network classification on multi-scale images [9,16].

In this paper, we present a fast cascaded license plate detection method. Instead of sliding window classification in multi-scale images, we first locate candidate plate regions on low-resolution image using vertical edge density, and then the candidate regions are verified using a linear SVM classifier with covariance features. Our method features efficient detection via fast candidate region location in down-sampled image. Wu et al. [14] employed morphological operations of bottom-hat and morphology gradient to detect the license plate candidates in video with original resolution of 320 × 240, however, they did not intentionally shrink the input frame to accelerate the detection procedure. Whereas, we detect candidate regions in images of 1/8 resolution (size 96x72 when the original image is 768x576), and verify the small number of candidate regions using a classifier in either 1/2 or 1/4 resolution.

Compared with the previous license plate detection methods using covariance feature and neural network classification [9,16], our method has some advantages. First, as mentioned above, we detect candidate regions in images of very low resolution to guarantee high speed. Second, we use a cascaded detection structure, which is more efficient than sliding window classification in multi-scale. Third, we use a linear SVM classifier for candidate region verification, which has better accuracy-complexity tradeoff than multi-layer neural networks as used in [9,16]. Overall, our method yields real-time and accurate license plate detection, and thus, it enables practical application.

The rest of this paper is organized as follows. Section 2 describes the proposed license plate detection method. Section 3 presents the experimental results and Section 4 draws concluding remarks.
2. License Plate Detection Method

The flow of our license plate (LP) detection method is depicted in Fig. 1. It has two main stages: the first stage selects candidate regions in 1/8 resolution image according to the edge density while the edges are detected in 1/2 resolution image; the second stage verifies the candidate regions using covariance features in either 1/2 or 1/4 resolution image. Though our candidate selection and verification modules works in images of fixed resolution, this does not hinder practical video surveillance application since the vehicle can move to a suitable scale in the video, or we can adjust the resolutions according to the imaging condition in application. We can also generalize candidate selection and verification to processing in multi-scale images and still allows real-time detection since our algorithm is extremely fast.

2.1. Fast Candidate Region Selection

In the field of text detection, the edge strength has been widely adopted for their superior performance. Compared with the non-license plate area, the plate regions tend to contain complex textures, thus we assume the areas with dense edges as license plates. However, some non-interested image regions, such as the front inlet grille and bumper of the vehicle, also follow this pattern, which produce a lot of negative candidates. As these regions mainly contain horizontal texture, we compute the density from vertical edges.

As lower image resolution barely impacts the subsequent process but saves considerable time, we firstly reduce by half both the width and height of the input image. The edge detection is implemented by 90° Prewitt (segmented by the average gray value of edge image) and Shenjun operators [11]. To obtain the image with one-pixel wide binary edge, the detection results of above two methods are combined by an logical AND-operation, which means, we consider a pixel as an edge point only if it is labeled as foreground by both operators. Given an input image Fig. 2a, the edge detection result is shown in Fig. 2b.

Since there are many background noise curves in the vertical edge image, we obtain the size of edges by chain code tracing [10], and remove the ones that are extraordinary long or short. An 3 × 3 image dilation is applied to the vertical edge image helping to preserve the edges of license plate from disconnection caused by edge detection (Fig. 2c).

After shrink the binary edge image to 1/8 size, the edge density at point ($x, y$) is obtained by the average gray value of a $M \times N$ block.

$$D(x, y) = \frac{1}{M \times N} \sum_{m} \sum_{n} G(x, y),$$

where $G(x, y)$ denotes the gray level at $(x, y)$ in the binary edge image, the size of block should be larger than the license plate among the dataset. The strides of X and Y axis are fixed to 4 and 2 pixels, respectively. We further speed up the computation by taking advantage of the data structure of integral image.

Fig. 3a shows the edge density map corresponding to Fig. 2a. Generally speaking, the region that containing the highest $T\%$ value in the histogram represents the location of the license plate, however, the sizes of the license plates are variant among different images, and the areas of front inlet grille and headlights may also have high edge density, thus a constant $T\%$ cannot locate all of the license plates in the dataset. Considering the SVM classifier trained with covariance feature will filter candidate plate regions in the following layer, here we have to ensure all of the potential interested areas are extracted, to this end, a multi-threshold strategy is proposed, in which the threshold $T\%$ is enumerated from 1% to $\beta$%. We empirically set $\beta$% as large as 5% to guarantee the region containing license plate is located.
The binarization results are shown in Fig. 3b. Image patches that centred at the geometrical center of connected components in the binary images are extracted. The sampling window is set according to the scale of license plate among the dataset. After shrink these regions to the half or quarter size, we pass them to the second layer. Various sizes of samples along with different resolutions are evaluated in the experiments.

2.2. Covariance Feature based Filtering

The filtering of candidate regions of license plate can be considered as a binary classification problem. The linear SVM verifies the candidate regions using covariance features extracted from either 1/2 or 1/4 resolution image.

2.2.1 Covariance Feature

Given an candidate image region I, we convert each pixel of the image to a d-dimensional feature, and V denotes all of the feature vectors (W × H × d dimension) extracted from I. In [13], a 9-dimensional covariance feature is used for face detection including pixel locations (x, y), RGB gray values and the first and second order derivatives of the intensities with respect to x and y. Since the linear SVM can automatically select the most distinctive features via the training process, we extract as much as 13-dimensional feature for each point of the image region to achieve higher detection accuracy.

\[
V(x, y) = [x \ y \ R(x, y) \ G(x, y) \ B(x, y) \ I_x(x, y) \ I_y(x, y) \ I_x'(x, y) \ I_y'(x, y) \ I_{xy}(x, y) \ I_{x'y'}(x, y) \ \theta_{xy}(x, y) \ \theta_{x'y'}(x, y)]
\]

where W and H mean the width and height of the candidate image region depending on the size of license plate in the dataset. R, G, B represent the RGB gray values, I_x(x, y) and I_y(x, y), I_x'(x, y) and I_y'(x, y) are the first and second order derivatives, respectively.

\[
I_{xy}(x, y) = \sqrt{I_x(x, y)^2 + I_y(x, y)^2}
\]

The covariance of the candidate region is given by

\[
C_V = \frac{1}{S-1} \sum_W \sum_H (V(x, y) - \mu)(V(x, y) - \mu)^T
\]

where S = W × H, \mu is the mean vector of the points in the area. As a result, a \(d \times d = 13 \times 13\) covariance matrix is obtained. Due to symmetry, we only store the left bottom coefficients of the matrix which compose a \((d^2 + d)/2 = 91\) dimensional feature vector. The covariance feature describes correlation of the features in a region, which is comparatively invariant to size, rotation and location of the objects in different images.

Since the covariance features computed from the sub-regions yields more distributional information of the input data, multiple covariance descriptors are combined to deliver a discriminative feature. As shown in the bottom of Fig. 4, we divide the input image to four parts, namely, left-top, right-top, left-bottom and right-bottom, respectively. There are nine strategies [16] (Fig. 4) to concatenate these corners, which increase the feature dimension to \(d \times d \times 9 = 91 \times 9 = 819\).

We provide the extracted covariance feature of a candidate region to the well-trained linear SVM. Fig. 5 shows the classification result, in which the positive candidate window is marked as red while the negative ones are green. The overlapping positive candidates are merged according
2.2.2 Training Instance Selection

We train a linear SVM with covariance feature via libLinear software [6]. For robustness, the positive samples containing some part of background are collected in a small neighborhood around the ground-truth of license plate. A great amount of negative samples can help to enhance the discriminant ability of the classifier, however, it leads to the serious unbalance between positive and negative data.

To reduce the scale of negative samples and keep the generalization of training data, we divide the input training image to six parts (Fig. 5). Since the background regions that surrounding the license plate (regions: 2-5) always contain complex textures including the front inlet grille and headlights, which could severely affect the classification, accordingly, these areas are densely sampled (steps of both directions are 4 pixels for 1/2 resolution image), meanwhile the other parts are sparsely sampled (the stride length equals to 10 pixels). All of the negative samples are collected avoiding the areas of license plate. The colored points in Fig. 5 illustrates the distribution of positive and negative sampling points. By training the linear SVM with a great amount of data which contains various scaling, rotation, partial occlusion, blurring and uneven illumination makes our system robust against the similar distortions in test dataset.

3. Experimental results

In the experiments, we evaluated our algorithm on two datasets collected from practical traffic surveillance videos. The images in both datasets represent various real-world detection circumstances, which are suffered from heavy noise, blurring, various illumination, scale (from 13 to 30 pixels height) and rotation (approximately from $-30^\circ$ to $30^\circ$), which renders the license detection a challenging task. Some license plate samples are shown in Fig. 6. The first dataset contains 301 images, in which 100 images are considered as training samples, 201 images as test samples. Using the training sample selection approach proposed in Section 2.2.2, 28,600 license plate and 261,539 background image patches are generated. As the second dataset consists of 1,867 vehicle images, we selected 867 images for training (21,675 positive and 536,762 negative data), and the others for testing. The image size of the first dataset is $768 \times 576$, and the second is $720 \times 288$. For unified expression, we assume the image height of the second dataset is also $576$, since it removes one field to eliminate the motion blur in interlaced surveillance video. We implemented the experiments with C++, and ran on a PC with Intel(R) Core(TM) i7-2630QM CPU 2.00GHz, and 12GB RAM.

For each image, the performance is measured by the overlap rate between ground truth and detection result.

$$M(D, G) = \frac{\text{overlap}}{\max(D, G)} \in (0, 1),$$

where $D$ and $G$ denote the areas of extraction result and ground truth, respectively. We counted a true-positive if $M(D, G) > 0.7$ and false-positive otherwise.
The histograms of oriented gradients (HOG) feature is introduced for fair comparison of different descriptors. The HOG feature concatenates histogram descriptors of the blocks inside the candidate sample. Each of the blocks \((W_B \times H_B)\) contains several cells \((W_C \times H_C)\), and two neighboring blocks overlap for robustness. For the image of half resolution, we fixed the block and cell size to 16 \times 16 and 8 \times 8; for the 1/4 resolution image, the sizes of the block and cell are 8 \times 8 and 4 \times 4. Each block consists of 4 cells, and the stride (block overlap) equals to 1 cell size. Dividing the gradient angles into 9 bins, \((W_B/W_C) \times (H_B/H_C) \times 9 = 36\) dimensional HOG feature can be extracted from each block.

The size of image samples (candidate window) in the dataset depends on the size of license plate. As the scales of plates in first dataset are similar, each image resolution corresponds to one sample size. For the second database, the height of the license plates vary from 13 to 30 pixels, we separately tested two different sizes of samples for each resolution.

Table I reports the computation cost for the extraction of 91 dimensional covariance feature from the second dataset with different sample resolutions. The window size corresponding to 1/2 and 1/4 resolution image is 95 \times 31 and 47 \times 15, respectively. We found that, due to the huge calculation of covariance descriptor, the feature extraction of low-resolution image patch is almost four times faster than the large one.

<table>
<thead>
<tr>
<th>Sample</th>
<th>Image size</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>95 \times 31</td>
<td>360 \times 288</td>
<td>0.60</td>
</tr>
<tr>
<td>47 \times 15</td>
<td>180 \times 144</td>
<td>0.16</td>
</tr>
</tbody>
</table>

Table 1: Computation cost (msec) of covariance feature.

Given an input image, we first obtained the candidate regions by edge density, and then classified these areas using the linear SVM. Table II compares the final detection accuracy of our cascaded detector on the first dataset in which the covariance and HOG features are extracted from image patches of different resolutions.

Considered the small scale of the first dataset (201 images for testing), and the license plates are about the same size, the detection task is relatively easier than the second one. The highest precision (100%) and recall (99.50%) are produced by the 819 dimensional covariance feature on the 1/4 resolution image. Due to heavy noises, there is only one plate shown in the first image among Fig. 7b that cannot be detected. We list the final license plate detection results on the second dataset with various candidate sample sizes and resolutions in Table III.

<table>
<thead>
<tr>
<th>F Sample</th>
<th>Image size</th>
<th>Dim</th>
<th>Time</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>91 \times 31</td>
<td>384 \times 288</td>
<td>91</td>
<td>10</td>
<td>99.00</td>
<td></td>
</tr>
<tr>
<td>91 \times 31</td>
<td>384 \times 288</td>
<td>819</td>
<td>14</td>
<td>99.00</td>
<td></td>
</tr>
<tr>
<td>45 \times 15</td>
<td>192 \times 144</td>
<td>91</td>
<td>9</td>
<td>98.01</td>
<td></td>
</tr>
<tr>
<td>45 \times 15</td>
<td>192 \times 144</td>
<td>819</td>
<td>11</td>
<td>99.50</td>
<td></td>
</tr>
<tr>
<td>88 \times 32</td>
<td>384 \times 288</td>
<td>1080</td>
<td>12</td>
<td>99.49</td>
<td>98.01</td>
</tr>
<tr>
<td>32 \times 16</td>
<td>192 \times 144</td>
<td>1080</td>
<td>10</td>
<td>97.28</td>
<td>89.05</td>
</tr>
</tbody>
</table>

Table 2: License plate detection accuracy (%) and computational time (msec) on first dataset.

<table>
<thead>
<tr>
<th>F Sample</th>
<th>Image size</th>
<th>Dim</th>
<th>Time</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>95 \times 31</td>
<td>360 \times 288</td>
<td>91</td>
<td>12</td>
<td>96.75</td>
<td>94.92</td>
</tr>
<tr>
<td>95 \times 31</td>
<td>360 \times 288</td>
<td>819</td>
<td>21</td>
<td>97.72</td>
<td>98.30</td>
</tr>
<tr>
<td>67 \times 23</td>
<td>360 \times 288</td>
<td>91</td>
<td>12</td>
<td>96.25</td>
<td>97.21</td>
</tr>
<tr>
<td>67 \times 23</td>
<td>360 \times 288</td>
<td>819</td>
<td>15</td>
<td>97.61</td>
<td>98.10</td>
</tr>
<tr>
<td>47 \times 15</td>
<td>180 \times 144</td>
<td>91</td>
<td>8</td>
<td>96.84</td>
<td>94.72</td>
</tr>
<tr>
<td>47 \times 15</td>
<td>180 \times 144</td>
<td>819</td>
<td>12</td>
<td>97.82</td>
<td>98.90</td>
</tr>
<tr>
<td>33 \times 11</td>
<td>180 \times 144</td>
<td>91</td>
<td>10</td>
<td>97.13</td>
<td>98.11</td>
</tr>
<tr>
<td>33 \times 11</td>
<td>180 \times 144</td>
<td>819</td>
<td>12</td>
<td>95.93</td>
<td>98.51</td>
</tr>
<tr>
<td>96 \times 32</td>
<td>360 \times 288</td>
<td>1188</td>
<td>12</td>
<td>96.46</td>
<td>94.55</td>
</tr>
<tr>
<td>72 \times 24</td>
<td>360 \times 288</td>
<td>576</td>
<td>12</td>
<td>85.78</td>
<td>96.27</td>
</tr>
<tr>
<td>48 \times 16</td>
<td>360 \times 288</td>
<td>1188</td>
<td>8</td>
<td>91.50</td>
<td>97.66</td>
</tr>
<tr>
<td>36 \times 12</td>
<td>360 \times 288</td>
<td>576</td>
<td>7</td>
<td>88.14</td>
<td>91.47</td>
</tr>
</tbody>
</table>

Table 3: License plate detection accuracy (%) and computational time (msec) on second dataset.

Experiments conducted on the second dataset show that the superior performance of covariance feature (819 dimensional), which yields the best accuracy of 97.82% precision and 98.90% recall. However, the HOG feature (1188 dimensional) only reaches the precision and recall of 96.46% and 94.55%. The results provided by the 91 dimensional covariance feature are even better than the HOG feature which involves much higher dimensionality.

The classification result shows that the covariance and HOG features extracted from the large size samples outperform the small ones. We suppose that, for the dataset with different scales of license plates, to produce higher accuracy, the size of the candidate sample (window) should not be smaller the largest license plate collected in the dataset.

Table IV shows that the HOG feature performs inferiorly with low-resolution samples. Conversely, since the covariance feature considers more about the correlation of internal structure instead of textures inside input data, we found that it is relatively insensitive to image resolution. The accuracies obtained by the covariance feature using down-sampled data are comparable with the high resolution ones.
is more, as lower resolution also indicates faster detection speed, the covariance feature that extracted from downsampled data is more suitable to our work for its efficiency and accuracy.

The first cascaded detection layer takes about 7msec to find the candidate plate regions, and the computation speed of the second layer depends on the number of candidate windows. As described in Section 2.1, we can normally obtain about 10 candidate plates for each image, and the total processing time varies from 8 to 12 msec depending on which feature is selected. Since we only compute a small amount of candidate samples, the fast covariance computation method based on integral image is not applied. However, our cascaded detection structure is still twice as fast as the method using integral image, which needs 20 msec for a 640 × 480 image.

Fig. 7 demonstrates some false-positive and false-negative image patches from the second dataset. These false-positive samples contain the character strokes or complex textures that are similar to the pattern of license plate, which lead to misidentification. The false-negative errors are resulted from many reasons, such as noise contamination, variation of background color and distortion caused by the viewpoint changing. We believe larger scale of train samples including various license plate distortion and backgrounds can further promote the detection results.

4. Conclusions

This paper proposed a novel license plate detection method based on the cascaded structure. The plate candidates are located by edge density in low-resolution image and then verified using a linear SVM with covariance descriptor. Compared with the traditional classifier-based license plate detection methods, our scheme guarantees high speed by only identifying several areas instead of scanning multi-scale images. Our algorithm allows real-time detection, since the computation cost of a PAL (768 × 576) survival image is about 10 msec. Experiments on a large scale dataset validate the efficiency of proposed method, in which 97.82% precision and 98.90% recall are achieved. Due to the distinctiveness of the covariance feature, it shows superior performance to the state-of-art HOG feature.

Acknowledgements

This work was supported by National Basic Research Program of China (973 Program) under Grant 2012CB316302 and by the National Natural Science Foundation of China (NSFC) under Grant 61271306 and 60875010.

References