Principal Visual Word Discovery for Automatic License Plate Detection

Wengang Zhou, Houqiang Li, Yijuan Lu, *Member, IEEE*, and Qi Tian, *Senior Member, IEEE*

Abstract—License plates detection is widely considered a solved problem, with many systems already in operation. However, the existing algorithms or systems work well only under some controlled conditions. There are still many challenges for license plate detection in an open environment, such as various observation angles, background clutter, scale changes, multiple plates, uneven illumination, and so on. In this paper, we propose a novel scheme to automatically locate license plates by principal visual word (PVW), discovery and local feature matching. Observing that characters in different license plates are duplicates of each other, we bring in the idea of using the bag-of-words (BoW) model popularly applied in partial-duplicate image search. Unlike the classic BoW model, for each plate character, we automatically discover the PVW characterized with geometric context. Given a new image, the license plates are extracted by matching local features with PVW. Besides license plate detection, our approach can also be extended to the detection of logos and trademarks. Due to the invariance virtue of scale-invariant feature transform feature, our method can adaptively deal with various changes in the license plates, such as rotation, scaling, illumination, etc. Promising results of the proposed approach are demonstrated with an experimental study in license plate detection.

Index Terms—Clustering, geometric context, object detection, principal visual word (PVW).

I. INTRODUCTION

LICENSE plate detection plays a key role in intelligent transportation systems. It can be applied in vehicle management, such as security control, traffic monitoring, automatic vehicle ticketing, and so on. Recently, this topic has attracted more attention in privacy protection as a lot of images and videos containing private information such as license plates are shared on the Internet. For instance, in Google Street View, one important task is to blur license plates in order to protect privacy.

License plates detection is widely considered a solved problem, with many systems already in operation. Nevertheless, the existing algorithms or systems work well only under some controlled conditions. For instance, some systems require sophisticated video capture hardware, possibly combined with infrared strobe lights (PVW), or require that the images be taken with little distortion from view-point changes. Although many reported results are very good, with even perfect accuracy on their test datasets, it is still a challenging task to detect license plates in open environment.

Generally, a license plate detection system has to solve two problems: where a license plate is located and how big it is. Usually, the candidate position of characters in the license plate is first identified, and the bounding box of the license plate is determined later. There are many challenges in license plate detection in an open environment, such as various observation angles from cameras, background clutter, differently sized license plates, poor image quality from uneven lighting conditions, and multi-plate detection. Typical instances can be found in Figs. 9 and 10.

Recently, the bag-of-words (BoW) model [1] based on local invariant feature [2], [3] has attracted a lot of attention in the computer vision and multimedia community. Due to the merits of scale-invariant feature transform (SIFT), i.e, the invariance property in rotation, scale, and illumination, it has been widely applied in object recognition [2], video tracking [4], content-based image retrieval [1], [5], and especially partial-duplicate image search [6], [7].

Observing that a specific character in different license plates can be considered as duplicates of each other, we bring in the idea of BoW model based on local feature for license plate detection. Since visual words generated from unsupervised clustering [5] are sensitive to noisy features from image background, it is desirable to yield the principal (discriminative and descriptive) visual words that correspond to each unique character in the license plate. Besides, these principal visual words (PVW) are expected to contain geometric context, which can be used to deduce the size of the corresponding character.

Motivated by the above discussion, we formulate license plate detection as a visual matching problem. For each character, we collect SIFT features falling into the character region and generate PVW by unsupervised clustering. The amount of PVW for each plate character is determined automatically.

W. Zhou and H. Li are with the Electrical Engineering and Information Science Department, University of Science and Technology of China, Hefei 230027, China (e-mail: zhwg@mail.ustc.edu.cn; lihq@ ustc.edu.cn).

Y. Lu is with the Computer Science Department, Texas State University, San Marcos, TX 78666 USA (e-mail: yl12@txstate.edu).

Q. Tian is with the Computer Science Department, University of Texas, San Antonio, TX 78249 USA (e-mail: qitian@cs.utsa.edu).

Besides SIFT descriptors, each visual word contains some geometric information, such as orientation, ratio of scale to character height, and relative position in the character region. Those geometric clues will be used to filter false feature matches and estimate the character and plate size. In testing, every valid match votes a support for plate location, and all supports are unified to discover potential license plates. Due to the invariance virtue of SIFT feature, our method can adaptively deal with various changes of license plate, such as distortion from observation views, scaling, and illumination. Multiple license plates in a single image can also be automatically detected.

We discuss our framework based on SIFT features [1]. Due to the relatively expensive time cost in feature extraction, our approach is suitable for applications without strong requirement of real-time efficiency, such as blurring license plate for privacy protection in Google Street View images. On the other hand, the SIFT feature adopted in our method can also be substituted by its invariant features, such as speeded up robust feature (SURF) [3], which is demonstrated as discriminative but much more efficient in implementation.

In this paper, we experiment on two types of license plates. It is straightforward to extend it to detection of other types of license plates in other regions or countries of the world. Our main contributions include the following.

1) We propose an automatic method to discover the PVW for each character in license plate. Each PVW is characterized with both local descriptor and some other geometric clues.
2) Based on visual matching with PVW, we propose an efficient scheme to accurately locate the image patch containing license plate.

II. Related Work

In the literature, many license plate detection algorithms have been proposed. Although license plate detection has been studied for many years, it is still a challenging task to detect license plates from different angles, partial occlusion, or multiple instances.

License plate detection investigates an input image to identify some local patches containing license plates. Since a plate can exist anywhere in an image with various sizes, it is infeasible to check every pixel to locate it. Generally, it is preferable to extract some features from images and focus only on those pixels characterized by the license plate. Based on the involved features, traditional license plate detection methods can be classified into three categories: color-based, edge-based, and texture-based. In what follows, we will review the related work in each category.

Color-based approaches are based on the observation that some countries have specific colors in their license plates. It is intuitive to extract license plates by locating their colors in the images. In [8] and [9], a test image is checked with a classifier of color model. Then, candidate regions from the classification results are verified with some post-processing to locate the plates. In [10] and [11], a color interval is determined from a mapping function to label potential regions of license plate.

In [12], the collocation of license plate color and character color is used to generate an edge image. Then, it checks neighbors of pixels with a value within the license plate color range to find candidate license plate regions. In [13], the color of each pixel in the image is identified using characteristic function. A series of morphological operations are used to merge the same plate color pixels into separate candidate areas. The license plate is then extracted from candidates using prior knowledge of its position in the image. In [14] and [15], color images are segmented by the mean shift algorithm into candidate regions, which are subsequently classified as with or without plate. A feature combination of rectangularity, aspect ratio, and edge density is exploited to determine the candidate regions. To address the effect of illumination variation, [16] proposes a fuzzy logic method for color recognition of license plates. Extracting the license plate using color information has the advantage of detecting inclined and deformed plates. However, it will be very sensitive to various illumination changes and suffer from false positive especially when other parts of testing images have the same license plate color.

Edge-based approaches are the most popular, with reliable performance in license plate detection. Generally, as a prior, license plate is characterized by a rectangular shape with a specific aspect ratio, and can be extracted by checking all possible rectangles in the image. In [17], a hybrid license plate extraction algorithm is presented based on the edge statistics and morphology for monitoring the highway ticketing systems. In [18] and [19], observing that the change of brightness in the license plate region is more remarkable and more frequent than elsewhere, edge detection combined with some morphological operations are exploited to find rectangles of interest for license plate location. In [20], the Hough transform is used to detect boundaries of license plates. In [21], a series of gray level morphological operations are applied to the input image to detect vertical edges of license plates. Some predefined rules are applied to extract the regions of license plates. A hybrid method for license plate extraction is proposed in [22]. It detects lines in the edge map and applies a weight assignment scheme to obtain an edge density map to select candidate plate regions. Extraction of license plate based on vertical edge detection and mathematical morphology is proposed in [23]. It first roughly determines the plate region and then makes use of the outer shape feature and inner shape feature to accurately detect the plate. Although edge-based methods are reliable in many cases, it may be distracted by some objects with rich texture and similar shape. Besides, when license plates undergo changes from observation, edge-based methods may partially detect or miss them.

Methods in the third category are focused on texture features. In [24], a covariance descriptor is employed with a neural network to detect license plates. In [25], an intensity saliency map is used to segment out the characters on a license plate, and then a sliding window on these characters is applied to compute some saliency-related features to detect license plates. Some learning-based methods treat license plate detection as a binary classification problem. [26] applied a support vector machine (SVM) classifier on color texture feature to detect license plates. In [27], modified from a
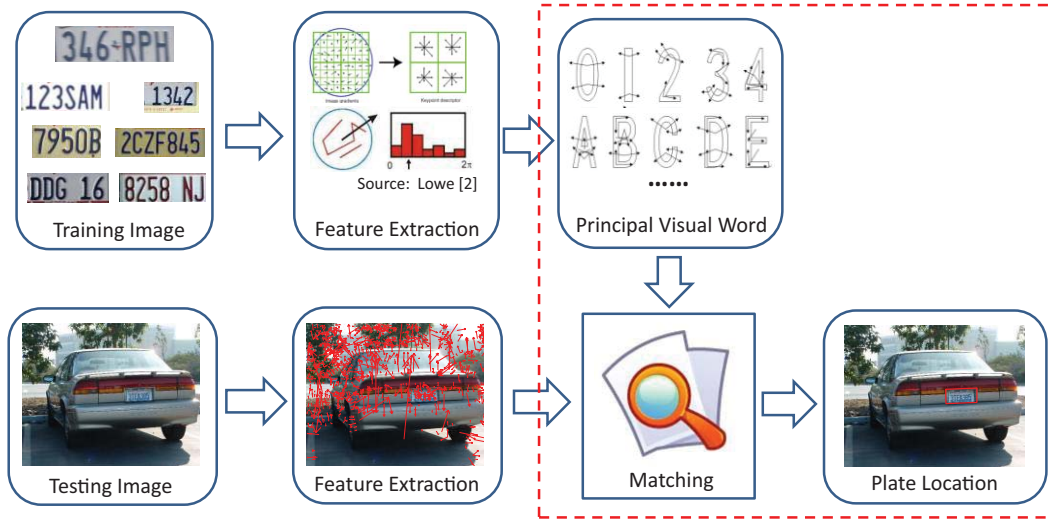
Fig. 1. Framework of license plate detection. We focus on the three components in the red dashed box.

face detection algorithm, a license plate detection algorithm using adaptive boosting (AdaBoost) on Harr-like features is proposed. In [28], with color information, a mean-shift based method is used to locate license plate. In [29], sliding concentric windows is used for license plate localization. Based on the "local" irregularity in the image, the method uses image statistics such as standard deviation and mean value as a "heuristic" for possible plate localization. In [30]–[33], AdaBoost is combined with Haar-like features to obtain cascade classifiers for license plate extraction. The Haar-like features are commonly used for object detection. Using the Haar-like features makes the classifier invariant to the brightness, color, size, and position of license plates. However, in open environment, license plates may undergo changes from rotation and observation views, and it is difficult to obtain enough comprehensive samples to cover such variations.

Different from the above three kinds of traditional locating methods, some other approaches based on local features have been proposed recently. In [34], license plate regions are first selected from maximally stable extremal regions (MSER) [35] detection results with some criteria. Then, the single character regions are removed and the borders of plate regions are determined. In [36], license plates are detected with a fusion of MSER and SIFT-based unigram classifier trained with core vector machine. The above two methods rely too much on the MSER detection result which is not reliable in open environment images as demonstrated in our study. In [37], a two-stage scheme is proposed to detect license plates. It first detects characters with Adaboost-based classifier. Then, SIFT descriptors are used with SVM to filter false positives.

Distinguished from the above methods, our approach is motivated from the BoW model [1] based on local invariant feature [2]. In BoW, a visual codebook is trained by clustering feature samples [5], and any new local feature is assigned to the visual word with the closest distance. Then, images can be compactly represented with a "bag" of visual words for efficient comparison. One drawback of the traditional visual word is that only SIFT descriptors are used for codebook construc-

tion, with the geometric clues ignored. In fact, these geometric clues, such as orientation, characteristic scale, and relative position, are very critical for object location. In this paper, we fully explore the invariant properties of local SIFT features to identify those repeatable local features, that is, PVW, in license plate characters. Our PVW are characterized by both SIFT descriptors and some invariant geometric context. With PVW, license plate in a novel image can be automatically and accurately located.

III. OUR APPROACH

As illustrated in Fig. 1, the framework of our approach consists of three key components: PVW generation, visual word matching, and license plate location. In PVW generation, we collect sufficient feature samples from labeled training images and generate some PVW for each character in license plate via clustering [38].

In visual word matching and license plate locating, we compare the extracted SIFT features of the test image with all discovered PVW, and locate the license plate based on the matching results. Let us denote the PVW set as $\{D, G\} = \{(d_i, g_i), i = 1, \ldots, N\}$, where d_i denotes appearance descriptor, and g_i denotes the geometric clues, N denotes the visual word number. Once the PVW of an object category is discovered, we can use it for detection in a new image. Given features $F = \{f_i\}$ for a test image, the probability that the test image corresponds to a sign of interest is

$$p(O|d_i, g_i) = \frac{p(d_i, g_i|O) \cdot p(O)}{p(d_i, g_i)} \propto p(d_i, g_i|O) \quad (1)$$

where $p(O)$ is prior of plate. The likelihood $p(d_i, g_i|O)$ is deduced as

$$p(d_i, g_i|O) = \sum_j p(d_i, g_i|f_j, O) \cdot p(f_j|O) \quad (2)$$

where $p(d_i, g_i|f_j, O)$ is modeled by matching feature f_j to the descriptor d_i of the PVW.
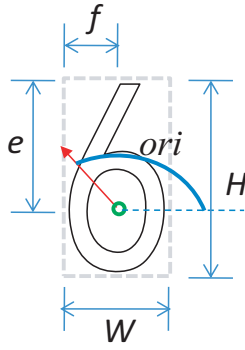
Fig. 2. Illustration of a PVW (red arrow) in the character "6."

Consequently, by searching for the local maxima of the likelihood function given by (1) for all PVW, we can find the initial hypotheses for license plate location. Some other prior heuristics can also be imposed to remove potential false positives.

In the following subsections, we will discuss PVW generation and local feature matching to extract license plate in detail.

A. PVW Generation

There are a certain number of sorted characters in license plates, each with the same format, but maybe undergoing illumination change or affine transformation. Since SIFT feature is invariant to changes in scale and rotation, and robust to illumination change and affine distortion [2], some repeatable and distinctive SIFT features to each character exist, called PVW. As shown in Fig. 2, a PVW is denoted as $V(\text{des}, \text{ori}, \text{rat}, \text{pos})$, where des is the 128-D SIFT descriptor, ori is the SIFT orientation ($-\pi \leq \text{ori} < \pi$), rat $= H/s$ (s is the SIFT scale), and pos $= (f/W, e/H)$ is a 2-D vector denoting the relative position of the key point in the character region. Both des and ori are originated from the standard SIFT features [2] des captures the local visual appearance with a concatenation of 8-D orientation histograms from 4 by 4 sub-patches around local interest point. ori denotes the dominant directions of local gradients around a key point. Relative to ori and des is represented to achieve invariance to image rotation changes [2].

Ideally, for a feature with high repeatability in a certain character, rat shall be identical. Given this specific SIFT feature with scale s, we can estimate the corresponding character height as rat $\cdot s$. Given an image patch of the character with height, v, we can also derive the scale of the SIFT feature as v/rat.

We collect many training images, each containing one or more license plates. License plates in the training images are all upright, with little affine distortion. Each character in the license plate is annotated and all SIFT features in each character region are extracted. Usually, many noisy features also exist. To discover the PVW of each character, we need to cluster the local features of each character and discover the most representative cluster centers as the PVW.

In this paper, we choose affinity propagation [38] to perform clustering. The most important merit of affinity propagation

is that it does not need to pre-specify the cluster number, which can be found automatically in the clustering process. In affinity propagation, a similarity matrix of samples shall be defined. We first give the distance metric, which will be used to define the similarity metric. The distance between two feature samples V^j and V^k is defined in (3)

$$d_{j,k} = \alpha \cdot D_d + \beta \cdot D_o + \gamma \cdot D_r + \delta \cdot D_p \qquad (3)$$

where α, β, γ, and δ are constant weighting factors, D_d, D_o, D_r, and D_p are the distance of descriptor, orientation, height-scale ratio, and position, respectively, and are defined as follows:

$$D_d = \frac{1}{\sigma} \sqrt{\sum_{i=1}^{128} (\text{des}_i^j - \text{des}_i^k)^2} \qquad (4)$$

$$D_o = \frac{1}{\pi} \cdot \min\left(\left|\text{ori}^j - \text{ori}^k\right|, 2\pi - \left|\text{ori}^j - \text{ori}^k\right|\right) \qquad (5)$$

$$D_r = \frac{1}{N} \left|\text{rat}^j - \text{rat}^k\right| \qquad (6)$$

$$D_p = \frac{1}{2} \sqrt{\sum_{i=1}^{2} \left(\text{pos}_i^j - \text{pos}_i^k\right)^2} \qquad (7)$$

where σ and N are normalization factors to make sure that both D_d and D_r range from 0 to 1.

The similarity metric is a decreasing function of the distance metric. There are many choices for it. In our implementation, the pair wise similarity between two feature samples V^j and V^k is defined as

$$s_{j,k} = -\left(d_{j,k}\right)^n, \quad (n > 0). \qquad (8)$$

In affinity propagation, the diagonal elements in the similarity matrix are referred to as exemplar preference, which will influence the number of identified clusters. Generally, without any priori, we set it as the median of the input similarities, as recommended in [38].

After clustering, we need to discover the most representative clusters. For each cluster, we count the number of image patches which contain at least one feature falling into the cluster. Then an image-number histogram is built. To select those representative clusters, a threshold thresh shall be specified on the histogram. Any cluster with image number above thresh will be selected. In each selected cluster, the PVW are defined as the average of all samples falling into that cluster. In our experiments, we set thresh $= 0.6 \cdot$Num, where Num is the total sample number of the specific character. Fig. 3 illustrates the feature clustering results of three characters: "0," "6," and "9." In each character, the PVW are highlighted in red color on the patch with its geometric information: ori, rat, and pos.

The PVW of characters from "0" to "9" are shown in Fig. 4, while those of characters from "A" to "Z" excluding "I" and "O" are illustrated in Fig. 5. There are no PVW of character "I" and "O" as these two characters are not found in any training plate.

B. Visual Word Matching

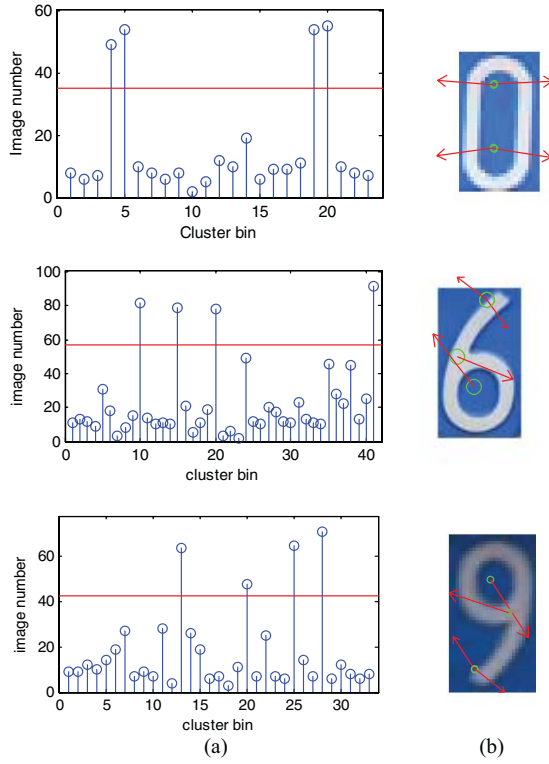Given a test image, we will discover those characters with features matched to the PVW. We first extract SIFT features

Fig. 3. (a) Number of images in each feature cluster of digit "0," "6," and "9." The red horizontal line denotes the threshold for selecting PVW. (b) Discovered four PVW for each character. Each red arrow denotes a PVW with a green circle as its location.
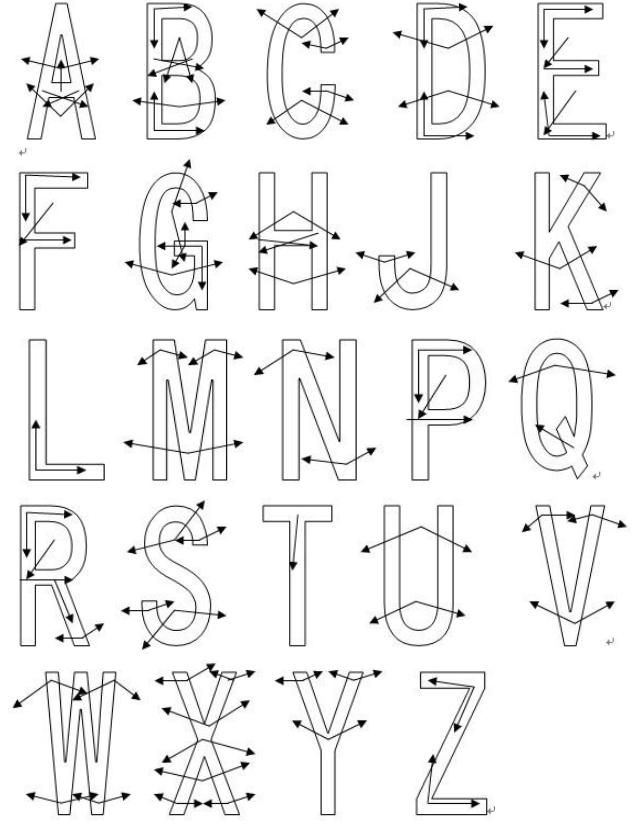

Fig. 5. PVW of each letter in Chinese license plate, from "A" to "Z," excluding "I" and "O." Each arrow denotes a PVW.

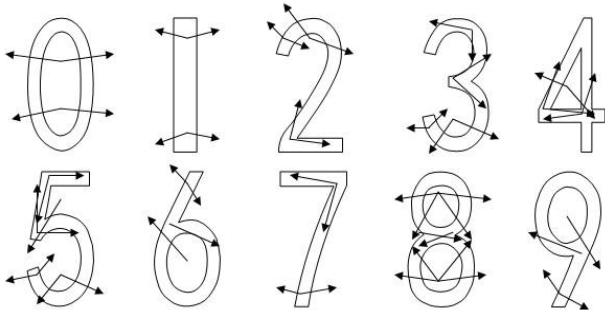Fig. 4. PVW of each digit in Chinese license plate. Each arrow denotes a PVW.

from the test image. Then each SIFT feature $F(\text{des, ori, scl})$ is compared with the PVW of each character. A feature is considered as a candidate match if the minimum descriptor distance to a certain PVW of a certain character is less than a constant threshold T

$$\{d^*, t^*\} = \min_{d,t} \arg \left\| \text{des}^F - \text{des}_t^d \right\|_2 \le T \qquad (9)$$

where des^F denotes the descriptor vector of a test SIFT feature, des_t^d denotes the descriptor vector of the t-th PVW in the d-th plate character. In standard SIFT features, descriptors are all normalized to be a constant const. In our experiments, we set $T = 0.5 \cdot$ const.

Each candidate match is recorded as $C(x, y, \text{angle, height}, \text{pos})$, where x and y denote spatial position of the test SIFT feature in the image plane, angle is the rotation angle from the

test feature to the matched visual word, height $= \text{rat}_{t^*}^{d^*} \cdot \text{scl}^F$ is the estimated height of the corresponding license plate, pos $= \text{pos}_{t^*}^{d^*}$ denotes the relative position in character.

In some cases, multiple license plates may exist in one image. However, the character features located in a plate must be close to each other, with distance less than the plate width, which can be derived from the estimated plate height of each character feature. Based on this observation, clustering can be performed on the locations of matched features in an incremental way. Matched features of each cluster correspond to a candidate license plate. Before locating the plate, false positive matches can be filtered based on consistency of orientation difference and estimated plate height. The details are discussed below.

The above descriptor matching based on descriptor comparison alone still incurs some false positives. The consistency of orientation difference can help to remove false positives. For all candidate features, a histogram of the orientation difference angle for all candidate matches is constructed. False positive candidate will be removed if its orientation difference is far away from the histogram peak. Similar operation can also be performed in the estimated plate heights for the remained candidate matches to further remove false positives.

Fig. 6 illustrates a matching instance. In Fig. 6(a), many SIFT features are extracted. After visual word matching and filtering, character features are identified, as shown in Fig. 6(b). The corresponding histogram of orientation difference is illustrated in Fig. 7.
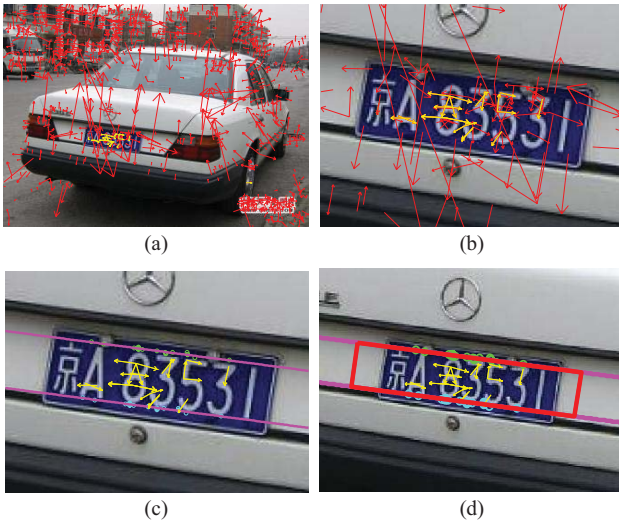
Fig. 6. Instance of license plate locating. (a) SIFT features detected from image shown by red arrow. (b) Identified character features of interest, highlighted in yellow. (c) Detected upper and lower bounding lines in local view. (d) Detected license plate patch highlighted in red bounding box.
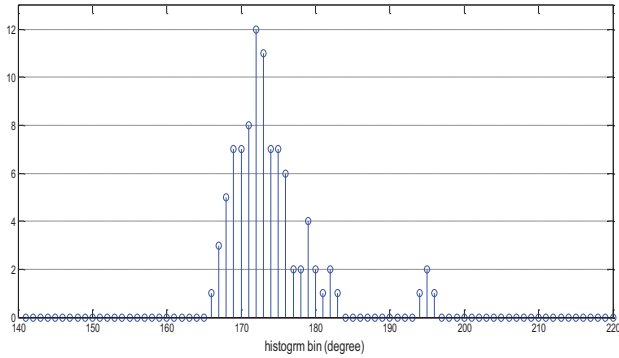

Fig. 7. Histogram of orientation difference between matched SIFT features. The histogram has been convolved with the template vector $(1, 2, 1)^T$ to suppress noise. Bins far from the principal histogram peak are considered false matches.

C. License Plate Locating

Once the character features in the test image are identified, we can make use of the geometric context of the matched PVW to locate the license plate. A bounding box will be estimated to encompass license plate by determining the upper, lower, left, and right bounding lines sequentially.

We first estimate the upper and lower bounding lines of license plate in images. Specially, each matched feature $C(x, y, angle, height, pos)$ estimates a point (x_{up}, y_{up}) of the upper bounding line by (10) and (11)

$$x_{up} = x - pos(2) \cdot h \cdot (\cos\theta - \sin\theta) \quad (10)$$
$$y_{up} = y - pos(2) \cdot h \cdot (\sin\theta + \cos\theta) \quad (11)$$

where h and θ are the median of height and angle of all valid matched features C, respectively, $pos(2) = e/H$. The origin is assumed to be at the upper-left corner of the image plane.

Then, for all upper bounding points, we estimate a line with linear regression. Similarly, we can also determine the lower bounding points (x_{down}, y_{down}) with (12) and (13),

Fig. 8. Sample results of license plates accurately detected by three approaches. Red bounding box: our approach. Green: HLPE. Yellow: LPE. Cyan: ESM.

and estimate the lower bounding line of license plate. An instance is shown in Fig. 6(c)

$$x_{down} = x + (1 - pos(2)) \cdot h \cdot (\cos\theta - \sin\theta) \quad (12)$$
$$y_{down} = y + (1 - pos(2)) \cdot h \cdot (\sin\theta + \cos\theta). \quad (13)$$

After that, we can also roughly estimate the left and right bounding lines. In license plate, the ratio of plate width to height is constant. When the plate height h is estimated, we can obtain the plate width w. Since license plate must cover all matched key points of SIFT feature, the interval between the left bound and the most right key point of matched feature shall be no less than w, so is that of the interval between the right bounding line and the most left key point.

Consequently, we can determine the minimal bounding box containing the license plate, as shown in Fig. 6(d). Although some background patch is also included, they can be removed with some other information, such as edge map. However, this is not our focus in this paper.

IV. EXPERIMENTAL RESULTS

Testing Dataset: To evaluate the proposed approach, we select two datasets, that is, LP dataset [39] and Caltech Cars 1999 [40]. The first dataset [39] is built by ourselves, containing 410 Chinese license plate images. Of them, 160 license plate images are downloaded from the Internet while another 250 images are taken by the authors. The second dataset contains 112 images with resolution of 896 × 592, each contains a U.S. license plate with a cluttered background, such
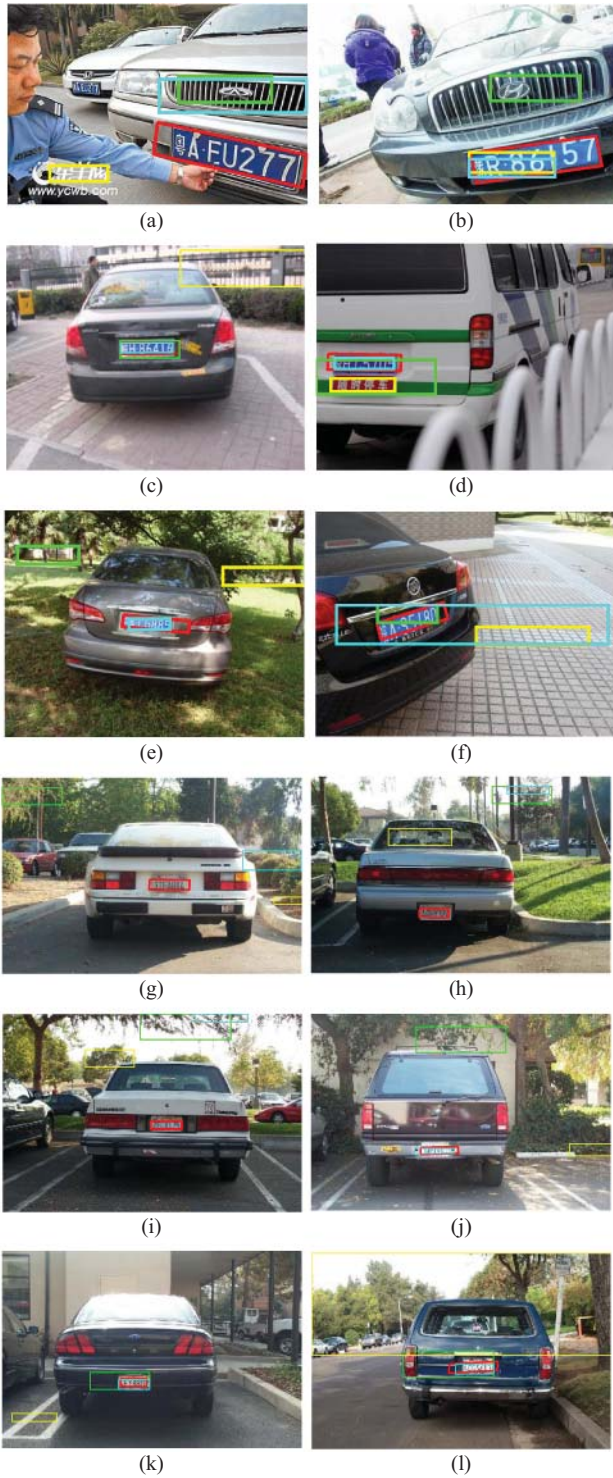
Fig. 9. Sample results of license plate detection by three approaches. Red bounding box: our approach. Green: HLPE. Yellow: LPE. Cyan: ESM.

Fig. 10. Sample detection results of license plates with different observation views. Red bounding box: our approach. Green: HLPE. Yellow: LPE. Cyan: ESM. No bounding box means no plates are detected by the corresponding method.

as trees or grass. In this dataset, the plate character height ranges from about 16 to 23. Some instances are shown in Figs. 9 and 13.

License plates in the first dataset exhibit different conditions, such as background clutter, rotation, and scale change, inhomogeneous illumination. The testing images do include many confusing elements, for example, road signs, written advertisements, logos, etc. Typical samples can be found in Figs. 8(d)–(f), 9(a) and (b), 11(c), 12(a) and (b), 13(b) and (d), and 14(d).

Training Dataset: Since the plate characters in the LP dataset [41] exhibit different formats from those in the Caltech Cars 1999 dataset [42], different training datasets shall be prepared to train the corresponding PVWs. To train the Chinese PVWs for LP dataset, we collected another 220 license plate images, in which license plates were of little affine distortion. To discover the PVWs for the plate characters in Caltech Cars dataset, another 364 U.S. license plate images were collected from the Web.

Features: In our feature extraction, we adopted the standard implementation of SIFT [1]. The Difference-of-Gaussian detector was used for key point detection and a 128-D orientation histogram along with orientation and scale was extracted to describe the local patch around each key point. There were 540 and 2868 SIFT features on average per image in the first and second dataset, respectively.

Fig. 11. Sample detection results of license plates with blur. Red bounding box: our approach. Green: HLPE. Yellow: LPE. Cyan: ESM. No bounding box means no plates are detected by the corresponding method.

Fig. 12. Sample detection results of license plates with occlusion. Red bounding box: our approach. Green: LLPE. Yellow: LPE. Cyan: ESM. No bounding box means no plates are detected by the corresponding method.

A. Evaluation

Comparison Methods: In the experiments, we compare our approach with other three algorithms of license plate detection. The first method [22] detects lines in image and constructs a weighted edge map, based on which color information is used to remove noise regions. We denote this algorithm as "HLPE." The second one [23] extracts license plate in a coarse-to-fine way by vertical edge detection and mathematical morphology. This algorithm is denoted as "LPE." The third comparison algorithm [17] denoted as "ESM" extracts license plate based on the edge statistics and morphology. These three comparison algorithms and our approach are all implemented in MATLAB. The experiments were performed on a PC with 4-G memory and 2.53-GHz CPU.

Metric: To evaluate the performance in terms of detection accuracy, we categorize the detection results into three levels: high level–true: license plate is totally encompassed by the bounding box and $A \cap B / A \cup B \geq 0.5$ [25], where A is the detected region and B is the ground truth region, low level–false: the license plate is totally missed by the bounding box, middle level–partial: the remaining results excluded by the above two types. In the detection results, the "false" type considers both false positives and false negatives (i.e., missed detection results), which are considered in combination for
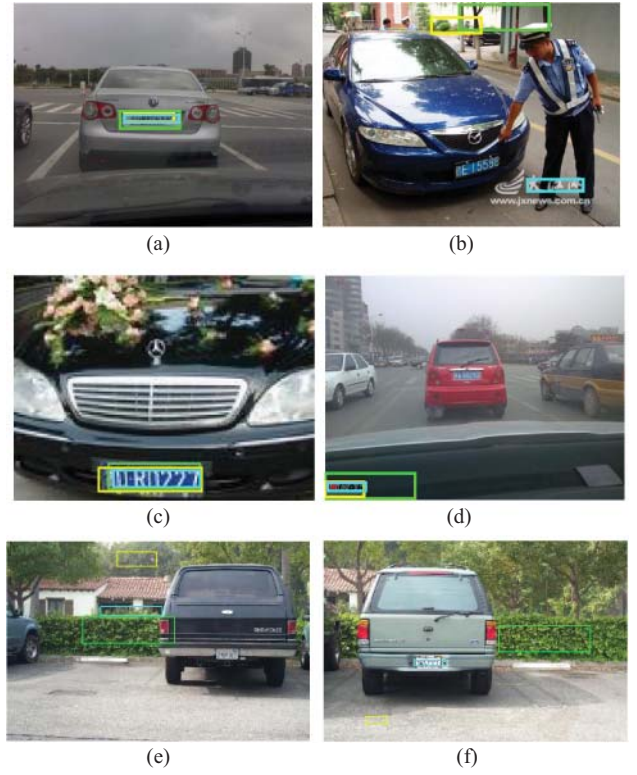

Fig. 13. Sample detection results of license plates with very low resolution. Red bounding box: our approach. Green: HLPE. Yellow: LPE. Cyan: ESM. No bounding box means no plates are detected by the corresponding method.

comparison of all three algorithms. The false positive rate is defined as the ratio of false detection results to the total number of detection results.

Accuracy: The detection results of the four approaches on two datasets are shown in Tables I and II, respectively. Our approach outperforms all of the three comparison approaches in accuracy for both datasets. On the first dataset, it achieves a 93.2% "true" detection rate, which is higher than HLPE by 12.4%, LPE by 8.6%, and ESM by 18.6%. Accordingly, the detection rate of our approach in "partial" and "false" is much lower than the two comparison algorithms. Specially, our approach makes a false positive rate of only 1.0%, which is much smaller than all the other three approaches.

On the second dataset, the "true" detection rate of our approach is 84.8%, which is higher than HLPE by 23.2%, LPE by 26.8%, and ESM by 16.1%. All the "true" detection rates of four approaches are lower than that on the first dataset. This is because the plates in the second dataset are of smaller size, and the background is much more cluttered. The false positive rate of our approach is also much lower than that of the three comparison approaches.

Efficiency: We investigate the time efficiency from two aspects. The first one is feature extraction time, and the second one is detection time after feature extraction. Our approach is based on SIFT feature, whose extraction time cost is larger than that of edge maps, as used in the other three comparison approaches. The detection time cost of our approach is proportional to the SIFT feature amount of image, while the

TABLE I
COMPARISON IN ACCURACY AND EFFICIENCY ON LP DATASET [39]

Approach	Accuracy			False positive rate	Time cost per image (second)	
	true	partial	false		Feature extraction	Detection
HLPE [22]	80.8%	6.6%	12.6%	12.6%	0.28	2.39
LPE [23]	84.6%	1.5%	13.9%	7.6%	0.01	0.11
ESM [17]	74.6%	7.3%	18.1%	17.9%	0.56	0.12
Our approach	93.2%	0.3%	6.5%	1.0%	1.36	0.22

TABLE II
COMPARISON IN ACCURACY AND EFFICIENCY ON CALTECH CARS DATASET [40]

Approach	Accuracy			False positive rate	Time cost per image (second)	
	true	partial	false		Feature extraction	Detection
HLPE [22]	61.6%	9.8%	28.6%	28.6%	2.56	36.30
LPE [23]	58.0%	1.8%	40.2%	29.5%	0.07	1.39
ESM [17]	68.7%	2.7%	28.6%	25.9%	4.20	2.80
Our approach	84.8%	1.8%	13.4%	4.5%	6.13	1.06

detection efficiency of the three comparison approaches are determined by the complexity extent of image texture.

On the first dataset, LPE performs the best in terms of detection efficiency. It takes only 0.11 s in average to process one image. ESM takes a comparable time cost of 0.12 s. As a contrast, HLPE costs 2.39 s, about 21 times the cost of LPE. Our approach also takes about twice the time of LPE.

On the second dataset, our approach is the most efficient in the detection stage. With more SIFT features per image, our approach takes 1.06 s on average, which is about four times more than that on the first dataset. Due to the complex background, all three comparison approaches take more time cost than our approach in the plate detection stage. The detection efficiency of HLPE is comparable to our approach, while ESM takes 1.7 s longer than our approach. HLPE is the most time-consuming and costs as much as 36.3 s in average.

B. Sample Results

When license plates in images are upright with little affine distortion and in good illumination condition, all the four methods are capable of accurately detecting the plates, as shown in Fig. 8. However, some challenges will greatly impact the detection accuracy. In the following, we will discuss these challenges and give sample license detection results of images with background noise, various observation views, blur, occlusion, low resolution, and multi-plate detection, respectively.

Background Noise: Since the three comparison algorithms are based on image edge map, they are apt to be attracted to the patches with rich texture or irrelevant text appearing in images. Fig. 9 gives 12 sample results of the three approaches. As can be seen, our approach is more stable and robust to the background noise.

Observation Views: In open environment, there are various observation views from cameras, which will make the edge map-based methods difficult to accurately extract the whole plate. However, benefiting from the invariance property of SIFT feature, our approach can effectively address that difficulty when the observation angle is within some tolerance range. Some examples are shown in Fig. 10 (a)–(f). When the observation angle incurs severe affine distortions, the corresponding key points of local features may be difficult to be detected, or the SIFT descriptors become out of shape. In such a situation, we may discover very few features corresponding to the character visual words, and consequently fail to detect the license plates. Fig. 10(g) and (h) shows two examples of that case.

It will be interesting to quantitatively study the tolerability of our PVW approach under various observation views. Since our PVW is derived from SIFT, the essence of the issue is SIFT's tolerability of affine distortion from observation views, which has been well studied by Mikolajczyk *et al.* [40].

Blur: Blurring can weaken the edge response of license plate and may pose a challenge for edge-map based methods. However, it casts relatively weak impact on SIFT. In other words, our method is more robust to blurring than edge-based approaches. Some sample detection results are shown in Fig. 11.

Partial Occlusion: Our approach detects license plate based on the local features of characters. Therefore, it is robust to partial occlusion of the plate. Even with partial occlusion of the license plates, the characters in the observed regions will provide clues to estimate the location of the plate. Fig. 12 shows two examples of occlusion. It can be observed that the plates are accurately detected by our approach.
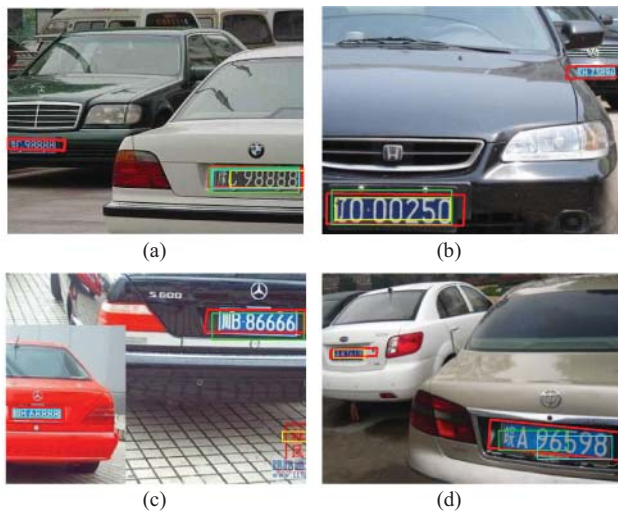
Fig. 14.   Sample detection results of our approach in images with two license plates. Red bounding box: our approach. Green: HLPE. Yellow: LPE. Cyan: ESM. No bounding box means no plates are detected by the corresponding method.

Low Resolution: When the license plate in the image is in good resolution, say, more than 25 pixels in character height, our approach works pretty well in localization. However, when the resolution of license plate in image is too low, it will be difficult for our approach to locate the license plate. It is observed from our experiments that when the height of characters in a license plate is smaller than 15 pixels our approach may miss the plate. Fig. 13 illustrates some failure examples with low resolution of license plate. Although edge-map-based approaches can make it sometimes, the low resolution problem is still a challenge to the three baseline algorithms.

Multi-Plate Detection: As discussed in Section III-B, based on incremental clustering of matched feature points, our approach can automatically detect multiple license plates. Fig. 14 shows the detection results of our approach on images, each with two license plates of different sizes. As a contrast, the three comparison algorithms can easily miss one or both plates in each image.

V. CONCLUSION

In this paper, we proposed an automatic approach for license plate detection in open environment. Our approach is based on PVW discovering and visual word matching. We identify the PVW of each character. With the invariance merit of SIFT feature, our approach is effective in dealing with various observation angles, scale change, and illumination variation, etc. Besides, our approach can detect multiple plates in an image automatically. Promising results were shown on two evaluation datasets. An additional virtue of our approach is that plate recognition is partially addressed when locating license plates. As demonstrated in the experiments, our approach is superior when the license plates have various observation views.

The weakness of our approach is that it may fail when the license plate resolution is too low, or when the distortion

from the observation angle is too severe. This is because these two conditions make it difficult to detect the SIFT features matching the PVW of related characters.

Our method is not limited to SIFT feature. Some other feature, such as SURF [3] which is much more efficient in implementation, can substitute SIFT. In our future work, we will test other local affine invariant features, such as SIFT descriptors based on affine invariant regions [39], to better address severe affine distortion of license plate.

In our experiments, we used two types of license plate images. It is straightforward to apply our method to license plates of other regions or countries of the world, just by learning another set of PVW of the specific place.

In our future work, we will evaluate our approach on infrared images for license plate detection. We will also explore the potential of our approach in detection of logos and trademarks.

REFERENCES

[1] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vision*, Oct. 2003, pp. 1470–1477.

[2] D. Lowe, "Distinctive image features from scale-invariant key points," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vision*, 2006, pp. 404–417.

[4] H. Zhou, Y. Yuan, and C. Shi, "Object tracking using SIFT features and mean shift," *J. Comput. Vision Image Understand.*, vol. 113, no. 3, pp. 345–352, Mar. 2009.

[5] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2006, pp. 2161–2168.

[6] O. Chum, M. Perdoch, and J. Matas, "Geometric min-hashing: Finding a (thick) needle in a haystack," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2009, pp. 17–24.

[7] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search," in *Proc. 18th Int. Conf. Multimedea*, Oct. 2010, pp. 511–520.

[8] X. Shi, W. Zhao, and Y. Shen, "Automatic license plate recognition system based on color image processing," in *Proc. Int. Conf. Comput. Sci. Appl.*, 2005, pp. 307–314.

[9] M. Zayed, J. Boonaert, and M. Bayart, "License plate tracking for car following with a single camera," in *Proc. IEEE Int. Conf. Intell. Trans. Syst.*, Oct. 2004, pp. 719–724.

[10] S. Yohimori, Y. Mitsukura, M. Fukumi, N. Akamatsu, and N. Pedrycz, "License plate detection system by using threshold function and improved template matching method," in *Proc. IEEE Ann. Meet. Fuzzy Inform.*, Jun. 2004, pp. 357–362.

[11] S. Yoshimori, Y. Mitsukura, M. Fukumi, N. Akamatsu, and R. Khosal, "License plate detection system in rainy days," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom.*, Jun. 2003, pp. 972–976.

[12] Y.-Q. Yang, J. B. R.-L. Tian, and N. Liu, "A vehicle license plate recognition system based on fixed color collocation," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Aug. 2005, pp. 5394–5397.

[13] W. Zhu, G. Hou, and X. Jia, "A study of locating vehicle license plate based on color feature and mathematical morphology," in *Proc. Int. Conf. Signal Process.*, Aug. 2002, pp. 748–751.

[14] W. Jia, H. Zhang, X. He, and M. Piccardi, "Mean shift for accurate license plate localization," in *Proc. IEEE Conf. Intell. Trans. Syst.*, Sep. 2005, pp. 566–571.

[15] W. Jia, H. Zhang, and X. He, "Region-based license plate detection," *J. Netw. Comput. Appl.*, vol. 30, no. 4, pp. 1324–1333, 2007.

[16] F. Wang, L. Man, B. Wang, Y. Xiao, W. Pan, and X. Lu, "Fuzzy-based algorithm for color recognition of license plates," *Pattern Recognit. Lett.*, vol. 29, no. 7, pp. 1007–1020, 2008.

[17] H. Bai and C. Liu, "A hybrid license plate extraction method based on edge statistics and morphology," in *Proc. Int. Conf. Pattern Recognit.*, Aug. 2004, pp. 831–834.

[18] D. Zheng, Y. Zhao, and J. Wang, "An efficient method of license plate location," *Pattern Recognit. Lett.*, vol. 26, no. 15, pp. 2431–2438, 2005.

[19] F. Faradji, A. H. Rezaie, and M. Ziaratban, "A morphological-based license plate location," in *Proc. IEEE Int. Conf. Image Process.*, Sep.–Oct. 2007, pp. 57–60.

[20] V. Kamat and S. Ganesan, "An efficient implementation of the Hough transform for detecting vehicle license plates using DSP'S," in *Proc. Real-Time Technol. Appl. Symp.*, 1995, pp. 58–59.

[21] J. Hsieh, S. Yu, and Y. Chen, "Morphology-based license plate detection from complex scenes," in *Proc. Int. Conf. Pattern Recognit.*, 2002, pp. 176–179.

[22] W. Le and S. Li, "A hybrid license plate extraction method for complex scenes," in *Proc. Int. Conf. Pattern Recognit.*, 2006, pp. 324–327.

[23] Y. Qiu, M. Sun, and W. Zhou, "License plate extraction based on vertical edge detection and mathematical morphology," in *Proc. Int. Conf. Comput. Intell. Software Eng.*, Dec. 2009, pp. 1–5.

[24] F. Porikli and T. Kocak, "Robust license plate detection using covariance descriptor in a neural network framework," in *Proc. Int. Conf. Video Signal Based Surveillance*, 2006, pp. 1–107.

[25] K. Lin, H. Tang, and T. S. Huang, "Robust license plate detection using image saliency," in *Proc. Int. Conf. Pattern Recognit.*, Sep. 2010, pp. 3945–3948.

[26] K. I. Kim, K. Jung, and J. H. Kim, "Color texture-based object detection: An application to license plate localization," in *Proc. Int. Workshop Pattern Recognit. Support Vector Mach.*, Aug. 2002, pp. 293–309.

[27] L. Dlagnekov and S. Belongie, "Recognizing cars," Dept. Comput. Sci. Eng., UCSD, San Diego, Tech. Rep. CS2005-083, 2005.

[28] W. Jia, H. Zhang, X. He, and M. Piccardi, "Mean shift for accurate license plate localization," in *Proc. Intell. Trans. Syst.*, Sep. 2005, pp. 566–571.

[29] C.-N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, "A license late-recognition algorithm for intelligent transportation system applications," *IEEE Trans. Intell. Trans. Syst.*, vol. 7, no. 3, pp. 377–392, Sep. 2006.

[30] H. Zhang, W. Jia, X. He, and Q. Wu, "Learning-based license plate detection using global and local features," in *Proc. Int. Conf. Pattern Recognit.*, 2006, pp. 1102–1105.

[31] H. Zhang, W. Jia, X. He, and Q. Wu, "A fast algorithm for license plate detection in various conditions," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, Oct. 2006, pp. 2420–2425.

[32] W. Le and S. Li, "A hybrid license plate extraction method for complex scenes," in *Proc. Int. Conf. Pattern Recognit.*, 2006, pp. 324–327.

[33] Y. Lee, T. Song, B. Ku, S. Jeon, D. K. Han, and H. Ko, "License plate detection using local structure patterns," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveillance*, Aug.–Sep. 2010, pp. 574–579.

[34] W. Wang, Q. Jiang, X. Zhou, and W. Wan, "Car license plate detection based on MSER," in *Proc. Int. Conf. Consumer Electron. Commun. Netw.*, Apr. 2011, pp. 3973–3976.

[35] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," *Image Vision Comput.*, vol. 22, no. 10, pp. 761–767, 2004.

[36] H. W. Lim and Y. H. Tay, "Detection of license plate characters in natural scene with MSER and SIFT unigram classifier," in *Proc. IEEE Conf. Sustainable Utilizat. Devel. Eng. Technol.*, Nov. 2010, pp. 95–98.

[37] W. T. Ho, H. W. Lim, and Y. H. Tay, "Two-stage license plate detection using gentle Adaboost and SIFT-SVM," in *Proc. First Asian Conf. Intell. Inform. Database Syst.*, 2009, pp. 109–114.

[38] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[39] *License Plate Dataset.* (2011) [Online]. Available: http://home.ustc.edu.cn/~zhwg/download/LicensePlate_dataset.rar

[40] *Caltech Plate Dataset.* (2003) [Online]. Available: http://www.vision.caltech.edu/Image_Datasets/cars_markus/cars_markus.tar

[41] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors," *Int. J. Comput. Vision*, vol. 1, no. 60, pp. 63–86, 2004.

[42] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *Int. J. Comput. Vision*, vol. 65, nos. 1–2, pp. 43–72, 2005.

Wengang Zhou received the B.S. degree in electronic information engineering from Wuhan University, Wuhan, China, in 2006. He is currently pursuing the Ph.D. degree in signal and information processing with the Electrical Engineering and Information Department, University of Science and Technology of China, Hefei, China.

His current research interests include large-scale multimedia information retrieval and multimedia and computer vision.
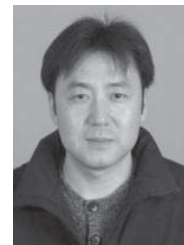
Houqiang Li received the B.S., M.Eng., and Ph.D. degrees from the University of Science and Technology of China (USTC), Hefei, China, in 1992, 1997, and 2000, respectively, all in electronic engineering.

He is currently a Professor with the Department of Electronic Engineering and Information Science, USTC. His research has been supported by the National Natural Science Foundation of China, State High-Tech Development Plan of China 863 Program, Microsoft, Nokia, and Huawei. His current research interests include video coding and communications, image and video analysis, and computer vision.

Dr. Li is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and is on the Editorial Board of the *Journal of Multimedia*. He has served on technical and program committees, organizing committees, and as a Program Co-Chair, Track Chair, and Session Chair for over ten international conferences.

Yijuan Lu (M'05) received the Ph.D. degree in computer science from the University of Texas, San Antonio, in 2008.

She is an Assistant Professor with the Department of Computer Science, Texas State University, San Marcos. Her current research interests include multimedia information retrieval, computer vision, machine learning, data mining, and bioinformatics.

Dr. Lu is a member of the Association for Computing Machinery. She was a Best Paper Candidate in the Retrieval Track of the Pacific-Rim Conference on Multimedia in 2007. She was the recipient of the Prestigious HEB Dissertation Fellowship in 2007 and was in the Star of Tomorrow Internship Program of Methicillin-Resistant *Staphylococcus aureus* in 2007.

Qi Tian (SM'04) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, in 2002.

He is currently an Associate Professor with the Department of Computer Science, University of Texas, San Antonio. His research projects have been supported by the Army Research Office, Department of Homeland Security, HP Laboratory, San Antonio Life Sciences Institute, CIAS, and Citizens Advice Scotland. His current research interests include multimedia information retrieval and computer vision.

Dr. Tian is a member of the Association for Computing Machinery (ACM). He is a Guest Co-Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, the *Journal of Computer Vision and Image Understanding,* the *ACM Transactions on Intelligent Systems and Technology*, and the *EURASIP Journal on Advances in Signal Processing*. He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and is on the Editorial Board of the *Journal of Multimedia*. He has served as a Program Chair, Session Chair, Organization Committee Member, and TPC for over 120 IEEE and ACM conferences, including ACM Multimedia, SIGIR, the International Conference on Computer Vision, and the International Conference on Acoustics, Speech, and Signal Processing.