中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-10-003

Recognition of Blurred License Plate Images

Pei-Lun Hsieh, Yu-Ming Liang, Hong-Yuan Mark Liao

April, 28 2010 || Technical Report No. TR-IIS-10-003
http://www.iis.sinica.edu.tw/page/library/LIB/TechReport/tr2010/tr10.html

Recognition of Blurred License Plate Images

Pei-Lun Hsieh [#1], Yu-Ming Liang [*2], Hong-Yuan Mark Liao [#3]
[#] Institute of Information Science
Academia Sinica, Nankang, Taiwan
[1] peilunhsieh42@gmail.com [3] liao@iis.sinica.edu.tw
[*] Department of Computer Science and Information Engineering
Aletheia University, Taiwan
[2] benson0409@gmail.com

Abstract—We propose a systematic way to perform blurred license plate image recognition. Our method only uses one license plate image and there is no need to perform character segmentation. There are three main steps involved in the proposed system. First, we perform character position identification and corresponding character list estimation using single-character templates. Then, the position of special symbol on a license plate is estimated. Finally, we expand the templates from single-character templates to multiple-character templates for refining the recognition results. The experiment results show that our method is superb in recognizing characters of blurred license plate images.

1. Introduction

The main purpose of this study is to design a systematic method to recognize the characters on a blurred license plate image. In many modern cities, it is a trend to set up video camcorders to monitor the traffic and potential criminal activities. Through video recording, violations of traffic rules and criminal activities are no longer undetected. Nevertheless, in consideration of the expense, the resolution of most video camcorders mounted at road intersections or important locations of buildings is very low. In addition, since these video clips are taken in outdoor environments, the quality of them is seriously influenced by the weather condition and very often they are blurred. Under these circumstances, recognizing the characters on a blurred license plate image becomes a very important issue for crime scene investigation. Many forensics-related research institutes in the world have devoted much effort in the past decades on this important problem. However, a systematic way that can either solve the problem or tell the investigators that the problem is not solvable is still deficient. In this paper, we shall develop a systematic way to solve the above-mentioned problem.

In the past two decades, a large number of algorithms have been proposed to recognize the characters on a license plate [1-7][9][14]. In [1], Anagnostopoulos *et al*. did an excellent survey on license plate recognition. They divided a complete license plate recognition procedure into three steps: (1) license plate detection, (2) character segmentation, and (3) character recognition. Among the numerous existing license plate recognition methods, Bai and Liu [5] put their emphasis solely on license plate detection. Since the intensity value detected in the interior region of a license plate is not stable, they computed the statistics of edges instead to extract the valid information from a license plate. As to the character segmentation step, it is well known that appropriate segmentation of the characters on a license plate image is a very difficult task because these characters are usually blurred and slant if they were taken by a video camcorder. In [4], Shi *et al*. did an excellent job on license plate character segmentation based on vertical and horizontal projections of binarized pixels. In [6], Nomura *et al*. tackled similar problem using a morphology-based approach. As to the character recognition issue, a number of methods are commonly used, such as statistical classifiers [7], multilayered feedforward neural networks [8], and template-based approaches [9].

In recent years, super-resolution-based image reconstruction [10][13][14] has been proposed as an alternative to tackle the license plate recognition problem. Unlike conventional pattern recognition-based approaches that perform recognition directly on a grabbed image frame, the first step of a super-resolution-based approach is to improve the quality of a grabbed image frame. In [10], Park *et al*. presented a comprehensive survey on super-resolution image reconstruction. Recent research on super-resolution usually splits the whole process into three steps. First, an image registration technique [11] is applied to register on input image sequence. Then, subpixel motion information is used to interpolate the high-resolution grid [12]. Finally, an observation model [13] and a probability model are applied to super-resolve them together. In [14], Suresh *et al*. used Markov random field and gradual nonconvexity optimization to generate a high-resolution license plate image from a sequence of low-resolution images.

However, both of the above-mentioned approaches have their own problems when dealing with blurred license plate images. For a pattern recognition-based approach, the accuracy of recognition highly depends on the result of character segmentation. However, a license plate image extracted from a low-resolution video clip is usually too blurred to detect the boundary of any two neighboring characters on the plate. As to a super-resolution-based approach, the current trend is to super-resolve multiple low-resolution images to a high-resolution one. However, a license plate image is usually extracted from a moving vehicle. License plate images taken at different time instances may have different perspective effect because they are located at different positions in the field of view. This will make the registration task even more difficult. Another drawback of applying a super-resolution-based approach is that it will "modify" the content of a target image. In a forensics-related field such as license plate image recognition, to "modify" the content of a potential evidence is inappropriate.

In this paper, we propose a systematic way to perform recognition of blurred license plate images. The proposed method only uses one license plate image and there is no need to perform character segmentation. The framework is composed of three main steps. First, after manually cropping the license plate image from a blurred license plate image, we perform character position identification and corresponding character list estimation using single-character templates. Then, the position of special symbol on a license plate is estimated. Finally, we expand the templates from single-character templates to multiple-character templates for refining the recognition results. The experiment results show that our method is indeed superb in recognizing characters of blurred license plate images.

The remainder of this paper is organized as follows. In Section 2, we introduce the three main steps in our system. The experimental results using real traffic video frames are detailed in Section 3. concluding remarks will be drawn in Section 4.

2. The Proposed Approach

It is known that all characters printed on a license plate are single font and of fixed size. Therefore, it is reasonable to propose a template-based license plate recognition approach to deal with a blurred license plate image. Since the purpose of this study is to provide possible license plate numbers of a suspected vehicle, we can manually crop the license plate region from a suspected vehicle. Since most of the detected vehicles are driven on a smooth road way and therefore we can make an assumption that the grabbed video frames are not seriously influenced by rotation or the perspective effect. Fig. 1 shows an example of cropping a license plate region from a suspected vehicle image. The proposed approach consists of three main steps: 1) determine potential character positions and estimate the most probable character set based on single-character templates; 2) determine the position of any special symbol on a license plate; and 3) perform license plate recognition based on multiple-character templates. The details of the three proposed steps are reported as follows.

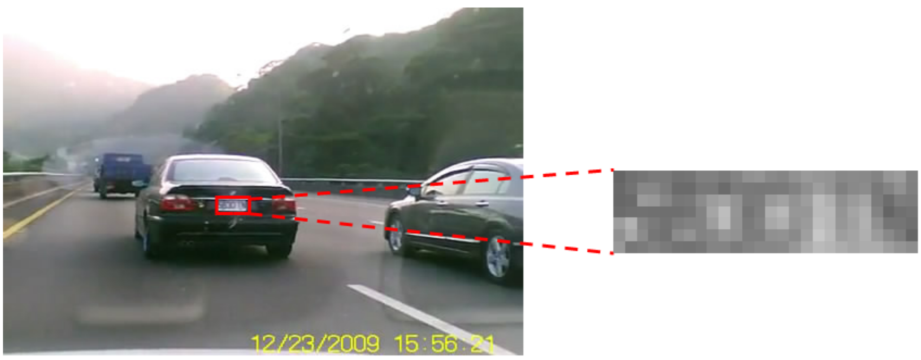


Fig. 1 An example of license plate region cropping.

2.1 Character Position and Character Set Estimation

After cropping the license plate region from a target video frame, the next step is to estimate the potential character positions. In Taiwan, the number of characters on a license plate is six. These characters can be letters or digits. There is always a hyphen sitting in-between the letter-digit region (the length is 2) and the pure-digit region (the length is 4). The letter-digit region can be two letters, one letter followed by one digit, or one digit followed by one letter. The pure-digit region, on the other hand, consists of four pure digits. The order of the letter-digit region and the pure-digit region can be reversed. Therefore, the position of the hyphen can be close to the left (the letter-digit region first and then four digits) or to the right (the pure-digit region first and then the letter-digit region) of a license plate.

To estimate the potential positions of the characters on a license plate, we apply the sliding window algorithm to achieve the goal. Before applying the sliding window algorithm, we need to collect the template of every possible character. Therefore, we collect the templates of letters (from A to Z) and digits (from 0 to 9) directly from high-resolution license plate images. We then normalize these extracted character templates into the same size and same intensity level. To apply the sliding window algorithm correctly, we need to calculate the average intensity difference as well as the size difference between each character template and the target license plate image and then compensate it. First, we normalize the height of every character template and make it equal to the height of a target license plate image. Then, we let the width of the target license plate image equal to six and one third times of the width of a normalized character template (in Taiwan the hyphen sitting in-between the letter-digit region and the pure-digit region is one third times of width of a character template). The six times of width is to fit the six characters and the one third time of width is to fit the hyphen existing on a license plate. After normalizing the size, the next step is to normalize the mean and the standard deviation of the intensity histograms between the target license plate image and every character template. Let $\boldsymbol{C}$ be the set of all possible characters on a license plate and I_t be the target license plate image. Let μ_{It} and σ_{It} be the mean and standard deviation of all pixel intensities in I_t, respectively. On the other hand, for each character template image, I_c, we can calculate its mean, μ_{Ic}, and standard deviation, σ_{Ic}, respectively. We need to compensate the intensity value of every pixel in I_c and make it become T_c as follows:

$$T_c(x,y)=(I_c(x,y)-\mu_{It})*\frac{\sigma_{It}}{\sigma_{Ic}}+\mu_{Ic}. \tag{1}$$

Fig. 2 shows an example of normalizing source character images into character template images based on the size and intensity obtained from a cropped low-resolution image.

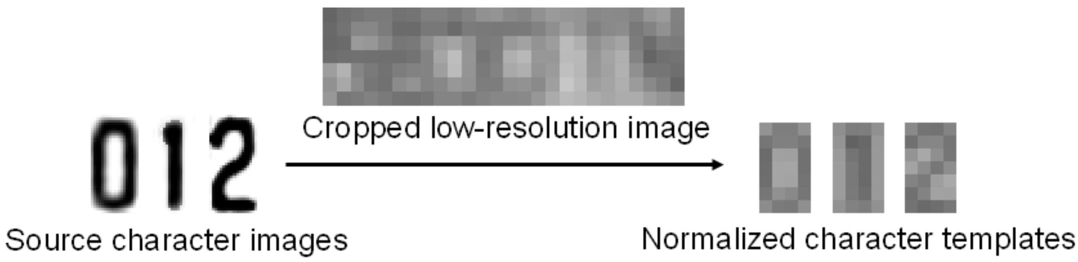


Fig. 2 An example of normalizing source character images into character template images.

After all the character templates are generated, we use every normalized character template, T_c, as a sliding window to search for all potential character positions on a license plate image. Let the size of T_c be $\mathrm{M}\times\mathrm{N}$. The degree of similarity between T_c and the subimage it covers during the sliding process can be measured by the L1-distance:

$$d_c(x_p)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left|T_c(x,y)-I_t(x_p+x,y)\right|,\ \ c\in\boldsymbol{C}, \tag{2}$$

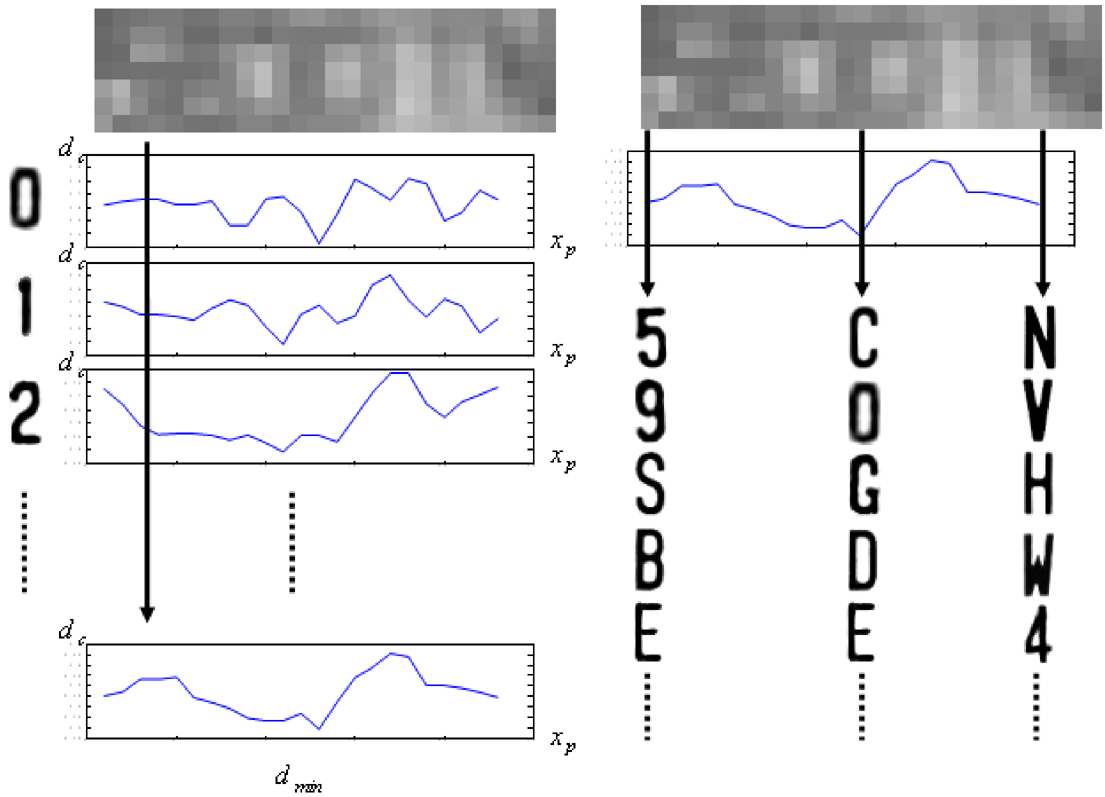
where x_p represents the starting x-axis position of a sub-image in the license plate image. Under these circumstances, if one slides a specific character template along the x-axis of a license plate image, the distance between this character template and all sub-images of the license plate image will form a distance curve as shown in Fig. 3(a). Fig. 3(a) shows the three distance curves generated by sliding the three character templates “0”, “1”, and “2”. Since there are in total 36 characters (0 to 9, and A to Z) in the character set, we can line up the 36 generated distance curves and determine a minimum distance curve. This minimum distance curve is determined by choosing the minimum distance value among the 36 distance values at all x_p positions. To determine the minimum distance at a specific position x_p, we use the following equation:

$$d_{min}(x_p)=\min_{c\in \boldsymbol{C}} d_c(x_p). \tag{3}$$

The bottom of Fig 3(a) shows the determined minimum distance curve (a scaled up version) of a specific license plate. Since every spot on the minimum distance curve means the most similar template, we need to find out the local minima along this curve because these positions should correspond to candidate character positions. We can determine these positions based on the following equation:

$$X_{cand}=\{x_{cand}|d_{min}(x_{cand})\le d_{min}(x)\,,\,where\,|x_{cand}-x|<M/2\}, \tag{4}$$

where M denotes the width of a character template.



(a) (b)

Fig. 3 Candidate character positions and corresponding characters estimation. (a) The calculated similarity curves and the minimum distance curve. (b) The estimated candidate positions and part of their corresponding characters.

After all possible candidate character positions are estimated, the corresponding character with the minimum distance for each candidate position x_{cand} can be found. However, since we are dealing with degraded images (seriously blurred), the minimum distance calculated at a specific position may not really reflect the truth. Therefore, we take the top few closest characters instead of taking only one. The cutoff threshold for screening qualified candidate characters is determined as follows. For a candidate character position x_{cand}, we calculate the average of all the obtained distances and use it as the cutoff threshold. Some estimated candidate positions and part of their corresponding characters are shown in Fig. 3 (b).

2.2. Treatment of Special Symbols

In the previous stage, we use single-character templates as sliding windows to determine potential character positions and the sets of possible characters associated with them. Under these circumstances, the best way to increase the fidelity of recognition results is to extend from single-character template to multiple-character template. However, to apply this extension directly without considering potential existence of special symbols in-between characters is inappropriate. In many countries, special symbols are placed on a license plate and these symbols do occupy spaces. Therefore, if one uses multiple-character template to perform recognition, he/she has to deal with these special symbols because they may influence the recognition outcome. A special symbol may be located at different positions of a license plate. For example, the license plate of North Carolina state and Kentucky state contains six characters. These six characters are split into two groups, one group has two characters and the other group contains four characters. There is a hyphen sitting in-between these two groups. As we have mentioned in Section 2.1, the license plates in Taiwan use a hyphen to separate the letter-digit region and the pure-digit region.

To confirm the position of a hyphen, we use single-character templates that cover all 36 characters and that only cover 10 digits (0 to 9), respectively, to generate their corresponding minimum distance curves. We use a real case to explain how this process works for identifying the position of a hyphen. The top row of Fig. 4 shows a cropped license plate image. The second and the third rows of Fig.4 show the minimum distance curves generated by sliding single-character templates that cover all 36 characters and all 10 digits, respectively. For those positions that have consistent local minimum in both curves (as indicated by the long vertical arrows in Fig. 4), we can judge they are digits. For the local minimum detected at the left side of the second row in Fig. 4, it is not detected in the minimum distance curve shown in the third row. This indicates it did not pass the pure-digit templates but passed the pure-letter templates. Using this strategy, we are able to tell the hyphen of this license plate should be close to the left hand side. Its position should sit between the second and the third character of the license plate. If one cannot find any character through the above process, he/she has to check whether there is any digit identified at the third or the fourth character position from left. If there is one or two digits found at these positions, we check its (or theirs) starting position. If an identified digit falls in the above section and its starting position at x-axis is an integer times of the width of a character template, then we judge the hyphen of this license plate is at the right hand side (it is a 4-2 combination). On the other hand, if this starting position is an integer times plus one third time of the width of a character template, then we judge this hyphen is at the left hand side (it is a 2-4 combination). If the above two rules cannot help come up with any correct judgement, then it means: (1) a targeted license plate does not have distinguishable letters in the first two-character or the last two-character sessions; (2) this license plate does not have distinguishable digits in the third and the fourth character positions. Under these circumstances, we need to perform exhausted search by considering both cases (i.e., both 2-4 and 4-2 cases) in the more sophisticated recognition process proposed in the next section. In this paper, the special symbol issue is discussed based on the example of Taiwan's license plates. However, similar methodology can be applied to any kind of license plates in the world if there is any special symbols existing on a license plate.

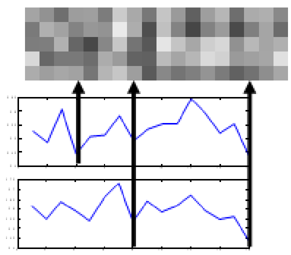


Fig. 4 The minimum distance curves generated by all 36 characters (middle row) and 10 pure digits (bottom row) are different.

2.3. Recognition Based on Multiple-Character Templates

In this section, we describe how to use multiple-character template to further improve the results generated by executing single-character template recognition. Since the license plate images are seriously blurred most of the time (due to the low-resolution nature of a video frame), the estimation made by executing single-character template-based recognition may not be correct. The erroneous estimations may be caused by several reasons. First, some license plate characters are seriously deteriorated and thus no character templates can pass the cutoff threshold. Under these circumstances, no possible character templates are chosen at these positions. Second, at some positions that have chosen possible characters, the order of these characters may be incorrect due to the low-resolution nature of a license plate image. Therefore, we propose the use of multiple-character template to increase the fidelity of recognition results.

Fig. 5 shows the hierarchical structure of multiple-character template-based recognition. Since the license plate in Taiwan has six characters, at the bottom level we illustrate six nodes that correspond to all six possible character positions. The second level from bottom in Fig. 5 shows all possible combinations of two-character templates and the third level are four possible three-character templates. We use an example illustrated in Fig. 6 to explain how multiple-character template recognition works. Fig. 6 shows three detected character positions (they were detected by single-character templates). The list of characters associated with each position (position 2, 3, and 6, respectively, from left to right) indicates an ordered list of candidate characters (from top (most probable) to bottom (least probable)). If one wants to extend from the single-character template at position 3 to the second level, then the two two-character templates he/she needs to use are the position 2-3 template and the position 3-4 template. Since we have found potential characters within the range of the first two characters and that of the 3-4 characters, we can judge the location of the hyphen based on the rules described in Section 2.2. In this case, the starting position of the third character is two and one third times of the width of a character template away from the left end. Therefore, we know immediately the hyphen should be between the second and the third characters of the license plate. If we check the potential character list of the second character, it also supports the above assertion because its top two character are letters. Under these circumstances, we can list all possible two-character template as follows. For the template of position 1-2, since the character at position 1 was not identified in the single-character template check, we use all 36 characters to respectively combine with the 11 characters ("V", "W", …, "Q") which were identified as the candidate characters in the first round check. Therefore, we have in total 396 (36x11) possible two-character templates for position 1-2. The possible two-character templates for position 2-3 can be deduced as follows. In this case, since the hyphen is located between the second and the third characters, we need to insert the template of hyphen (1/3 with of a character template) when generating the two-character template. On the other hand, since these two positions are associated with their own candidate character lists, we have in total 33 (11x3) possible two-character templates. They are V-1, V-4, V-7, W-1, W-4, W-7, B-1, B-4, B-7, and so on. An example of two-character template at position 2-3 is shown in Fig. 7. The normalization process of size and intensity is the same as the process stated in Section 2.1. As to all possible two-character templates for position 3-4 we can deduce as follows. Since position 4 was not matched with any candidate characters in the single-character test, we need to include all digits as possible characters (we know this position belongs to the pure-digit region). Therefore, we use "1", "4", and "7" to respectively combine with all 10 digits (0 to 9) to form 30 (3x10) possible two-character templates. The other possible two-character templates can be deduced in the same way. For position 3-4, we have in total 30 distances calculated. We computed the average of these distances and use it as the cutoff threshold to screen out those have larger distances. Fig. 8 shows 10 two-character templates survived in the screening process for position 3-4. As to the three-character templates, four-character templates, and five-character templates, we can use the same methodology to deduce them. Finally, these possible templates will be ranked based on their distances with the sub-images of the original license plate image.

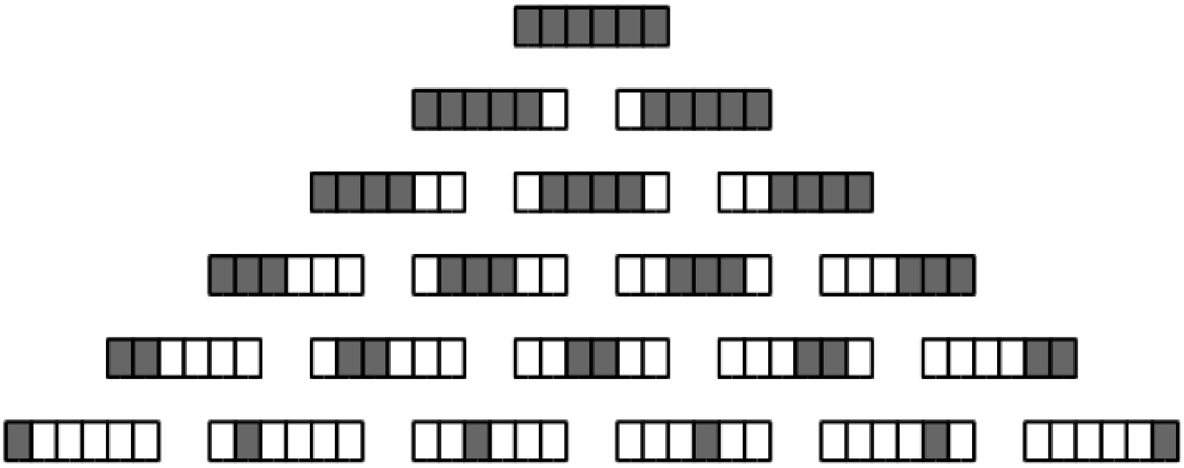


Fig. 5 The hierarchical structure of the multiple character templates.

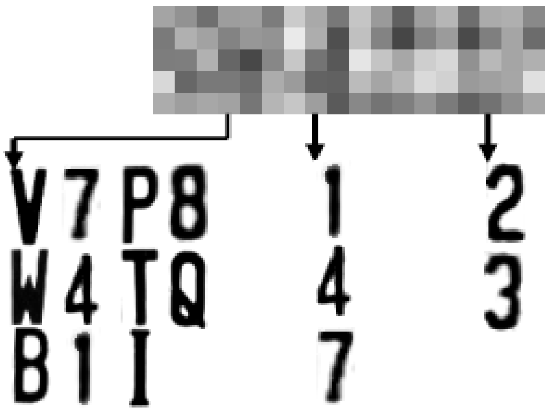


Fig. 6 An example of three detected character positions and the list of characters associated with each position.

Fig. 7 An example of two-character template with a hyphen in-between.

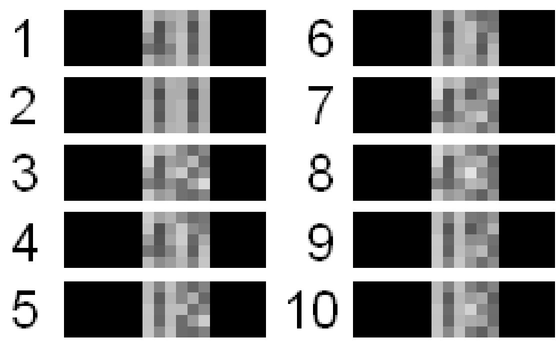


Fig. 8 The 10 two-character templates survived in the screening process for position 3-4.

3. Experimental Results

To test the effectiveness of the proposed method, we made use of Taiwan's license plates to conduct experiments. Fig. 9 shows three typical license plates used in Taiwan. Fig. 10 shows three blurred license plate images that were used in our experiments. We know the ground truth of these blurred license plate images by closer look on these vehicles. A set of more clear license plate images in contrast to Fig. 10 are shown in Fig. 11. The ground truth numbers of these license plate were 8P-1311, SV-4232, and 5830-TN. In the three blurred license plate images, the height of the cropped license plate image was at most 7 pixels. Under these circumstances, many conventional license plate recognition systems cannot even process them [1]. With this poor resolution, on the other hand, even the human visual system can hardly recognize any character from these license plate images.

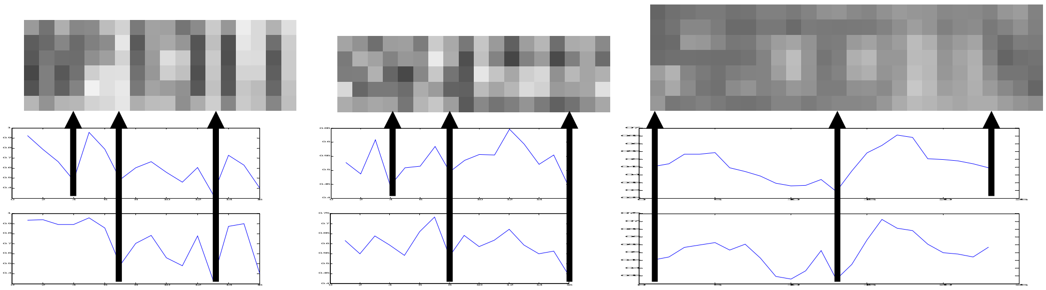
Fig. 8 Some typical license plates used in Taiwan

Fig. 10 Three blurred sample license plate images used in the experiments. (From top to bottom) The actual characters of the license plates are 8P-1311, SV-4232 and 5830-TN. (From left to right) Low-resolution vehicle image, and the cropped license plate region.

Fig. 11 The closer look on the three vehicles in Fig. 9.

In the first step of the experiments, we used the set of character templates that contained all characters (A-Z, 0-9) and that contained only digits (0-9), respectively, to calculate the minimum distance curves of the three test license plate images. Fig. 12 shows the results of this experiment. The first row shown in Fig. 12 are the three blurred license plate images cropped directly from Fig. 10. The middle row and the bottom row of Fig. 12 show the minimum distance curves and the potential character positions determined by using all 36 character templates and 10 digit templates, respectively. Comparing the minimum distance curves shown in the second row and the third row of

Fig. 12, one can easily separate the digit-letter region and the pure-digit region. The lists of partial candidate characters associated with each detected character position are shown in Fig. 13 (each list with priority from top to bottom).



(a) (b) (c)

Fig. 12 Results of potential character position estimation. The first row shows the three license plate images cropped from Fig. 8. The middle row shows the minimum distance curve and candidate character positions determined by 36 character templates. The bottom row shows the minimum distance curves and candidate character positions determined by 10 digit templates.

(a) (b) (c)

Fig. 13 The results of corresponding candidate character estimation on each potential position.

From Fig. 13 it is known that even after potential character position estimation, there are still some positions with no characters detected. Under these circumstances, we used the detected characters to determine where the special symbol was located and then proceeded to the multiple-character recognition process. Fig. 14 shows how the proposed three-step license plate recognition process operated and the corresponding hierarchical structure established when handling the license plate “8P-1311.” The hierarchical structure shown in Fig. 14 was built from all the possible single-character templates (bottom nodes) to all the possible two-character templates (all nodes one level up), and finally to all the possible six-character templates (on the top of the structure).

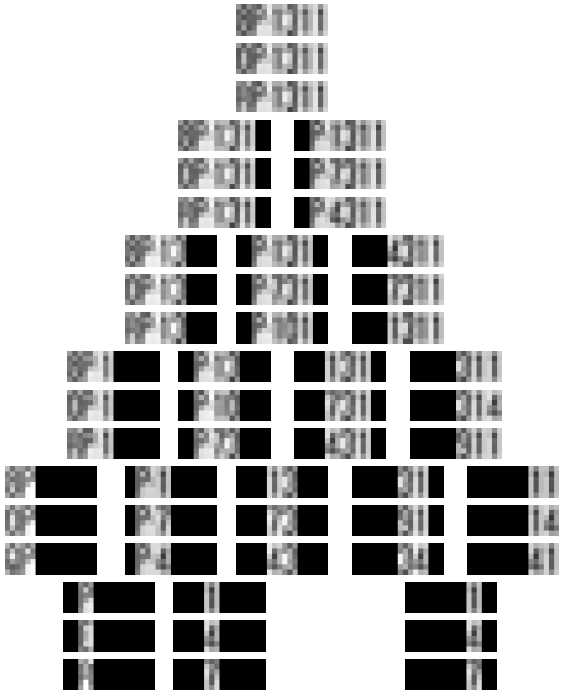
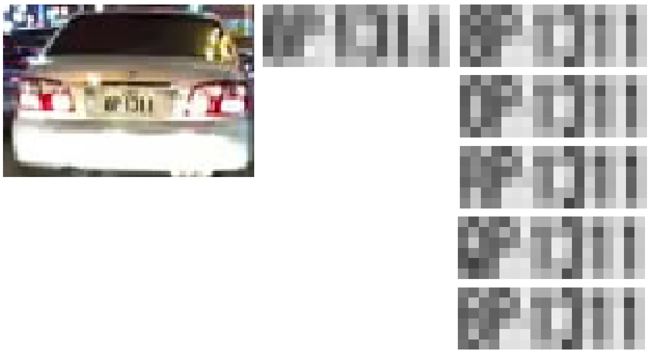


Fig. 14 A hierarchical structure of multiple-character template correction.

In the process of crime investigation, an accurate analysis of license plate images can help reduce the search space and avoid spending time on improbable vehicles. Since the objective of this investigation is to recognize the characters of a blurred license plate image, our strategy is to reduce the size of the search space to a small range. Fig. 15 and 16 show two sets of results obtained by applying our license plate recognition algorithm. In the results shown in Fig. 15, the ground truth license plate number: “8P-1311,” was ranked number one in the estimation list. Fig. 16 shows another set of results. In this case, the ground truth license plate: “SV-4232,” was ranked number two out of the top five estimated results.



(a) (b) (c)

Fig. 15 Recognition result of the license plate “8P-1311.” (a) the low-resolution vehicle image, (b) the cropped license plate region, (c) the recognition result with top 5 ranks: “8P-1311,” “0P-1311,” “AP-1311,” “QP-1311,” and “QP-1311.”

(a) (b) (c)

Fig. 16 Recognition result of the license plate “SV-4232.” (a) the low-resolution vehicle image, (b) the cropped license plate region, (c) the recognition result with top 5 ranks: “VW-4232,” “SV-4232,” “SW-4232,” “VW-4332,” and “SV-4732.”

4. Conclusions

We have proposed a systematic way to perform recognition of blurred license plate images. Only one license plate image is needed to estimate potential license plate numbers. Besides, there is no need to perform character segmentation. The framework is composed of three main steps. First, we identify the character positions and then derive their corresponding character lists using single-character templates. In the second step, we try to identify the location of the special symbol of a license plate. Finally, we expand the templates from single-character templates to multiple-character templates for refining the recognition results. The experiment results show that our method is quite successful in recognizing characters of blurred license plate images.

References

[1] C.-N. E. Anagnostopoulos, I. E. Anagnostopoulos, I. D. Psorulas, V. Loumos, and E. Kayafas, “License plate recognition from still images and video sequences: a survey,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 9, No. 3, pp. 377-391, Sep. 2008.

[2] S.-L. Chang, L.-S. Chen, Y.-C. Chung, and S.-W. Chen, “Automatic License Plate Recognition,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 5, No. 1, pp. 42-54, Mar. 2004.

[3] C. N. Anagnostopoulos, I. Anagnostopoulos, V. Loumos, and E. Kayafas, “A license plate recognition system for intelligent transportation system applications,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 7, No. 3, pp. 377-392, Sep. 2006.

[4] X. Shi, W. Zhao, and Y. Shen, “Automatic License Plate Recognition System Based on Color Image Processing”, in Lecture Notes on Computer Science, vol. 3483, pp. 1159–1168, 2005.

[5] H. L. Bai and C. P. Liu , “A hybrid license plate extraction method based on edge statistics and morphology,” in *Proceedings ICPR*, 2004, pp. 831-834.

[6] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, and T. Shiose, “A novel adaptive morphological approach for degraded character image segmentation,” *Pattern Recognition*, Vol. 38, No. 11, pp. 1961–1975, Nov. 2005.

[7] X. Pan, X. Ye, and S. Zhang, “A hybrid method for robust car plate character recognition,” *Engineering Application Artificial Intelligence*, Vol. 18, No. 8, pp. 963– 972, Dec. 2005.

[8] Y. Amit, D. Geman, and X. Fan, “A coarse-to-fine strategy for multiclass shape detection,” *IEEE Transactions on Pattern Analysis Machine Intelligence*, Vol. 26, No. 12, pp. 1606–1621, Dec. 2004.

[9] P. Comelli, P. Ferragina, M. N. Granieri, and F. Stabile, “Optical Recognition of Motor Vehicle License Plates,” *IEEE Transactions on Vehicular Technology*, Vol. 44, No. 4, pp. 790-799, Nov.

1995.

[10] S. Park, M. Park, M. Kang, "Super-Resolution Image Reconstruction: A Technical Overview," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21-36, May 2003.

[11] L. G. Brown, "A survey of image registration techniques," ACM Computing Surveys, Vol. 24, No. 4, pp. 325-376, Dec. 1992.

[12] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graph. Models Image Processing*, Vol. 53, No. 3, pp. 231–239, May 1991.

[13] M. Elad and A. Feuer, "Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images," *IEEE Transactions on Image Processing*, Vol. 6 , No. 12, pp. 1646-1658, Dec. 1997.

[14] K. V. Suresh, G. Mahesh Kumar, and A. N. Rajagopalan, "Superresolution of License Plates in Real Traffic Videos," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 8, No. 2, pp. 321–331, Jun. 2007.