An Automated Vehicle License Plate Recognition System

Hitesh Rajput and Tanmoy Som, Indian Institute of Technology (Banaras Hindu University)

Soumitra Kar, Bhabha Atomic Research Centre

Using a three-phase process—image capture, plate localization, and number recognition—an automated license plate recognition system achieves 97 percent accuracy in identifying the plates on 40 different car models.

In recent years, license plate recognition (LPR) has become a core technology of security and traffic applications that range from traffic surveillance to parking lot access control to information management for monitoring purposes.[1] Simply stated, LPR helps identify vehicles and provides a reference for further vehicle tracking and activity analysis.

A key LPR challenge is the large variety of license plates, which differ with respect to color, shape, size, and pattern. Other obstacles include severe weather conditions, poor lighting, and low camera resolution as they affect image quality when the plate is captured by a camera in real time. A moving vehicle also can affect the camera's aperture speed, causing a blurring effect. Several approaches have attempted to deal with these challenges, including optical character recognition, the indirect Fourier transform (IFT)-based fast method, and the morphologic method (see the sidebar for additional approaches). But these were only valid for cases with specific constraints.[2,3] We describe here a method for LPR that uses Daubechies wavelet transform to overcome multiple limitations.

THE PROPOSED METHOD

Our proposed LPR framework aims to overcome the limitations of existing approaches. It consists of three phases: image capture, license plate localization, and number recognition.

Image capture

In the first phase, we capture the image of the vehicle and normalize to a standard dimension of 400 × 300 pixels. We then convert the RGB image into a grayscale one:

$$A_{GL} = \frac{3A_R + 6A_G + A_B}{10}, \tag{1}$$

where A_{GL} is converted gray-level image, and A_R, A_G, and A_B are the RGB spectrum of the color image, respectively. Figure 1a shows original image, A_R, A_G, and A_B.

 0018-9162/15/$31.00

Plate localization

We took A_B as our candidate for plate localization and performed a wavelet decomposition to compute the approximation and details coefficient matrices. Figure 1b shows the four frequency bands representing low, horizontal, vertical, and diagonal frequency energy, respectively. The horizontal and vertical frequency energies can locate the license plate because the plate has a high frequency, so to find the plate's vertical location, we plot the pixel intensity's energy curve for each row of the vertical frequency band. Similarly, to find the plate's horizontal location, we plot the pixel intensity's energy curve for each column of the horizontal frequency band. Figure 2a shows the plots.

A Gaussian filter modifies the input signal with a Gaussian function. Equation 2 gives the one-dimension impulse response:

$$g(x) = \sqrt{\frac{a}{\pi}} e^{-ax^2} . \tag{2}$$

Equation 3 gives the frequency response:

$$\hat{g}(f) = e^{-\frac{\pi^2 f^2}{a}} , \tag{3}$$

where f is the ordinary frequency. These equations can also be expressed with the standard deviation as a parameter:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-x^2}{2\sigma^2}} . \tag{4}$$

Equation 5 gives the frequency response:

$$\hat{g}(f) = e^{-\frac{\pi^2 f^2}{2\sigma_f^2}} . \tag{5}$$

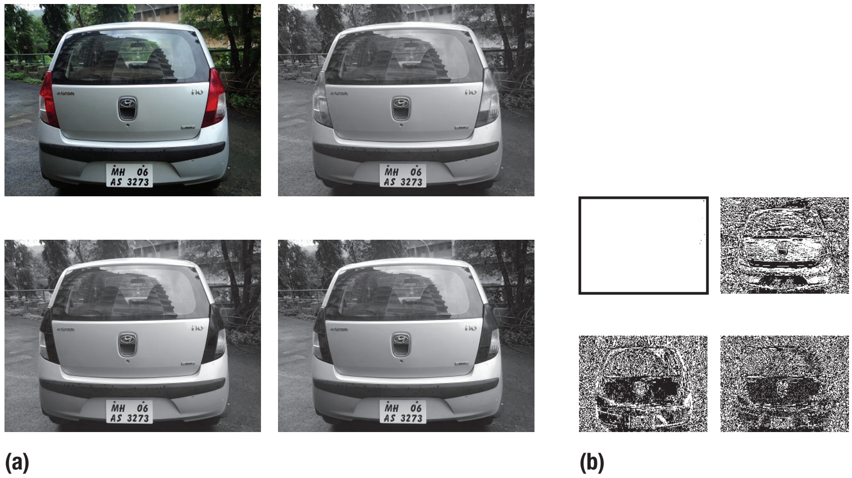

FIGURE 1. Image capture and plate localization: (a) clockwise, from top left, original image, A_R, A_G, and A_B and (b) single-level discrete 2D wavelet transform of A_B. The four frequency bands represent low, horizontal, vertical, and diagonal frequency energy, respectively.

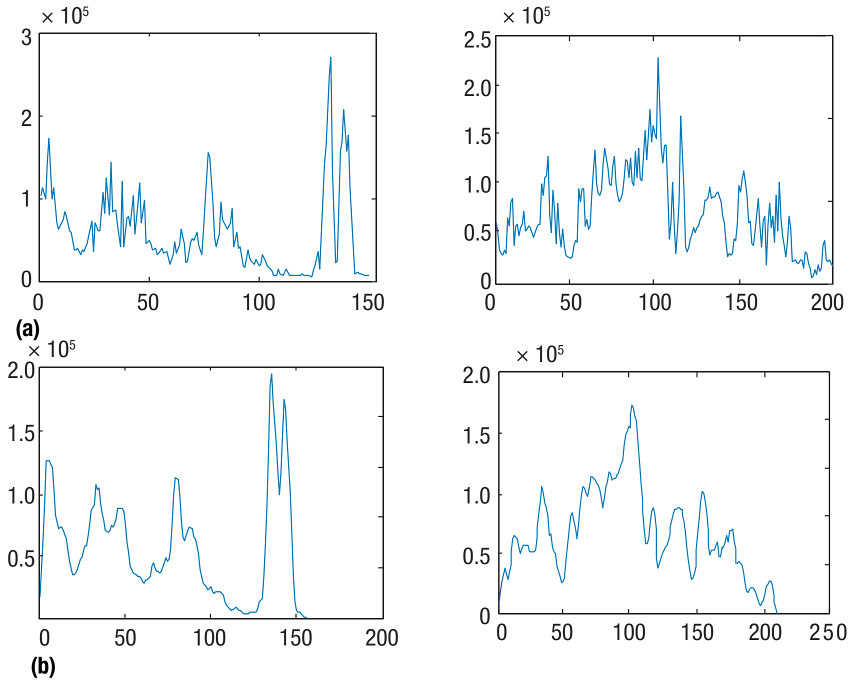

FIGURE 2. Vertical (summing up pixel intensities of each row for vertical location) and horizontal (summing up pixel intensities of each column for horizontal location) frequency energy curves: (a) original and (b) smoothed. The vertical frequency curve's high energies represent the number plate. The x-axes show number of pixels, the y-axes show pixel intensities.

Gaussian filtering has many advantages: it is rotationally symmetric; filter weights decrease monotonically from the central peak, giving the most weight to central pixels; there is a simple and intuitive relationship between the size of σ and the smoothing; and Gaussian filtering is separable. This latter point offers additional advantages: being separable means that Gaussian filtering first convolves the image with a 1D horizontal filter and then convolves

RELATED WORK IN LICENSE PLATE RECOGNITION

In most cases, license plate localization is a necessary precursor to license plate recognition (LPR). We can group the methods used to locate the license plate's location or region in images or videos in the literature into three processing categories: binary image, gray-level, and color. Character segmentation is an important precursor to character recognition, which we can similarly break down into the same three categories. To recognize segmented characters, several algorithms use pattern/template matching or learning-based classification.

BINARY IMAGE PROCESSING

To extract license plate regions from background images, techniques based on combinations of edge statistics and morphology can achieve good results. Some researchers have applied edge operators on a gray image after smoothing and normalizing to extract horizontal and vertical edge maps. They then perform statistical analysis on the edges to detect the license plate's shape. The final decision is based on connected component analysis (CCA).

This algorithm can achieve a 99.6 percent detection rate from 9,825 images—assuming that the license plate frame's edges are clear and horizontal. To accommodate frames that were deformed or skewed, Hao Wooi Lim and Yong Haur Tay used maximally stable extremal regions to obtain a set of character regions.[1] However, this method of extracting characters from the binary image to define the license plate region is time-consuming because it processes all binary objects. Furthermore, it gives an incorrect identification if there is other text in the image.

GRAY-LEVEL PROCESSING

Some researchers have exploited the contrast between the characters and the background—for example, Hsi-Jian Lee and colleagues considered blocks with a high edge magnitude and variance as the license plate region.[2] Researchers have also applied image transformation methods based on the Hough transform (to detect straight lines) and Gabor filters (to analyze textures).[2] However, this method is valid only when the image background is simple. Another disadvantage of this method is that the Hough transform's computational complexity is very high.

COLOR PROCESSING

In many countries, license plate text and background colors are strictly fixed, based on algorithm design, but such algorithms still fail because of varying lighting conditions.[3]

Accurate plate location is fundamental to the whole recognition process's success. Some researchers propose the use of color features to

FIGURE 3. Plate localization. This is the result of taking half the maximum vertical and horizontal frequency energy curves and traversing from both left and right sides to localize the plate vertically.

the result with a 1D vertical filter. For a $k \times k$ Gaussian filter, 2D convolution requires k^2 operations per pixel, but by using separable filters, we reduce this to $2k$ operations per pixel. Figure 2b shows the results of applying the Gaussian filter on both energy curves. The vertical frequency curve's high energies represent the number plate because, as we mentioned earlier, we find a higher frequency near the plate.

To compute the high-energy frequency band, we compute a threshold value to operate on the vertical and horizontal frequency energy curves. We take half the maximum vertical energy and traverse from left to right and right to left in the vertical frequency curve to compute the y coordinates. The y coordinates that are greater than the threshold value represent the plate's vertical location. Figure 3 shows an example.

To accurately measure the horizontal location, rather than to just operate on the horizontal frequency energy curve in Figure 2, we perform the same operation on the image in Figure 3. If we traverse from left to right on this image, we clearly observe that the frequency is highest near the number plate, so to find the horizontal location, we repeat the process we did to find it vertically. Figure 4 shows the result. Note that we do not need half the maximum y coordinates to compute the threshold value here because the frequency is very high near the number plate compared to other locations. Instead, we take 0.9 times the maximum value to compute the threshold value and traverse from left to right and right to left to compute the x coordinates. Figure 5a shows the result.

localize the plate,[4] but this approach fails when plates have multiple colors. Yu-Chiun Chiou and colleagues proposed using vertical edge matching for plate recognition, but varying plate sizes or poor contrast between the plate and the car body make this method unreliable.[5] Mu-Liang Wang and colleagues used horizontal scans of repeating contrast changes for plate recognition, but it suffers from a ringing effect that occurs along the edges of the filtered spatial domain image.[6]

Fully connected feed forward artificial neural networks with sigmoidal activation functions have also been used for character recognition, but the successful number plate identification rate is only 80 percent and processing time is 15 seconds.[6] One study proposed a license plate detection method based on sliding concentric windows and a histogram, but it was both time-consuming and only suitable for Taiwanese plates.[7] Zhen-Xue Chen and colleagues combined the rectangle shape, texture, and color features to extract the license plate,[8] and had a success rate of 97.3 percent, but the process was too computationally complex and time-consuming. Another study used dynamic programming to extract license plate numbers, but the success rate of segmentation was only 84.5 percent.[9]

References

1. H.W. Lim and Y.H. Tay, "Detection of License Plate Characters in Natural Scene with MSER and SIFT Unigram Classifier," *Proc. IEEE Conf. Sustainable Utilization Development Eng. Tech.* (CSUDET 10), 2010, pp. 95–98.
2. L. Hsi-Jian, C. Si-Yuan, and W. Shen-Zheng, "Extraction and Recognition of License Plates of Motorcycles and Vehicles on Highways," *Proc. 17th Int'l Conf. Pattern Recognition* (ICPR 04), 2004, pp. 356–359.
3. L. Miao, F. Wang, and H. Wang, "Automatic License Plate Detection Based on Edge Density and Color Model," *Proc. IEEE Control and Decision Conf.* (CDC 09), 2009, pp. 3718–3721.
4. W. Wei, "An Automatic Method of Location for Number-Plate Using Color Feature," *Proc. IEEE Int'l Conf. Image Processing* (ICIP 01), 2001, pp. 782–785.
5. Y.-C. Chiou et al., "Optimal Locations of License Plate Recognition to Enhance the Origin-Destination Matrix Estimation," *Proc. Eastern Asia Soc. Transportation Studies*, vol. 8, 2011, pp. 1–14.
6. M.-L. Wang et al., "A Vehicle License Plate Recognition System Based on Spatial/Frequency Domain Filtering and Neural Networks," *Proc. 2nd Int'l Conf. Computational Collective Intelligence: Technologies and Applications* (ICCI 10), LNCS 6423, Springer, 2010, pp. 63–70.
7. K. Deb, H.-U. Chae, and K.-H. Jo, "Vehicle License Plate Detection Method Based on Sliding Concentric Windows and Histogram," *J. Computing*, vol. 4, no. 8, 2009, pp. 771–777.
8. Z. Chen et al., "Automatic License Plate Location and Recognition Based on Feature Salience," *IEEE Trans. Vehicle Tech.*, vol. 58, no. 7, 2009, pp. 3781–3785.
9. D.-J. Kang, "Dynamic Programming-Based Method for Extraction of License Plate Numbers of Speeding Vehicle on the Highway," *Int'l J. Automotive Tech.*, vol. 10, no. 2, 2009, pp. 205–210.

Some adjustments might be required to compute the threshold values for horizontal and vertical locations.

Number recognition

To recognize and identify plate numbers, we convert the extracted gray-colored plate in Figure 5a into the black-and-white image in Figure 5b. We start by setting the pixel intensities to 255 and 0, where pixel intensity is greater than or equal to and less than 128, respectively. The next step is character segmentation. We know that the numbers are written line-wise—plates in India normally have two lines, so to separate them, we plot the vertical frequency's energy curve as in Figure 6a. The number of subcurves that start and end with 0 represent the number of lines in the license plate. Figures 6b and 6c show the result.

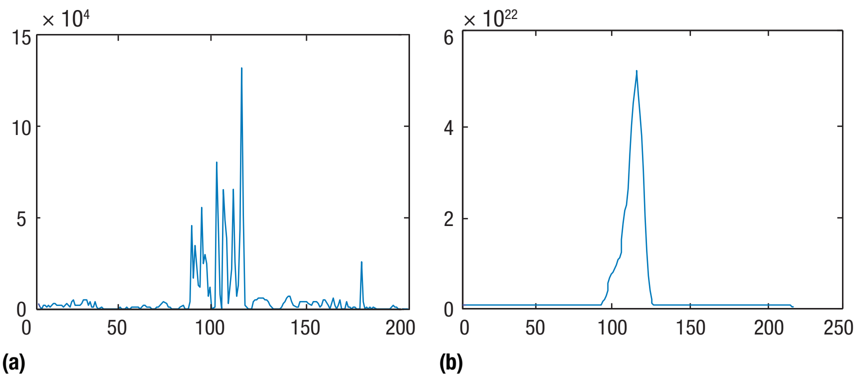

FIGURE 4. Energy and smoothing curves for Figure 2's horizontal frequency energy. The spike indicates the location's proximity to the number plate. The *x*-axes show number of pixels, the *y*-axes show pixel intensities.

To perform character segmentation, we plot the energy curve,

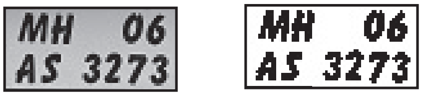

FIGURE 5. Plate localization: (a) gray and (b) and black-and-white images. Applying thresholding converts the gray image into black and white for accurate character/number recognition.

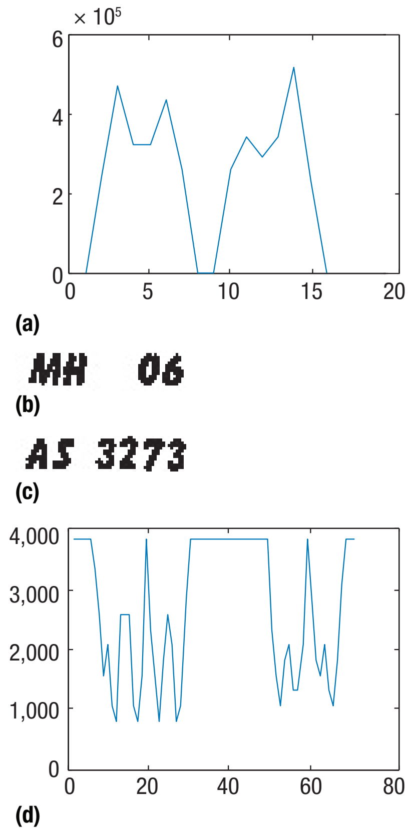

FIGURE 6. Number recognition: (a) vertical frequency's energy curve; (b) and (c) line split; (d) Figure 6b's energy curve. The *x*-axes show number of pixels, the *y*-axes show pixel intensities.

FIGURE 7. Character segmentation of license plate number. The area between the points at which the curve starts falling toward the *x*-axis represents a single character.

TABLE 1. Results of plate localization, character segmentation, and character recognition tests.

	Quantity of images	Percentage
Plate localization technique		
Total number of available images	250	100
Correctly found plate images	244	97.6
Images whose plates were found in only candidate region	218	87.2
Images whose plates were not found	6	2.4
Character segmentation technique		
Total number of characters	2,500	100
Successfully segmented characters	2,445	97.8
Unsuccessfully segmented characters	105	2.2
Character recognition technique		
Total number of characters	2,500	100
Successfully recognized characters	2,400	96.0
Unsuccessfully recognized characters	100	4.0

column-wise, as in Figure 6d. Smoothing of this curve is not required because we converted the gray image into black and white. The area between the points at which the curve starts falling toward the *x*-axis represents a single character. Figure 7 shows the segmented characters. We have taken a margin of 1 pixel on both sides of a character.

After the segmentation process, we need to normalize the characters to refine them into a block containing no extra white spaces (pixels) in any border. To match the characters with the database, the input images of the segmented characters must equalize to a 38 × 20 block of database characters. The next step is template matching, which is an effective algorithm for character recognition. Template matching compares the segmented characters with the ones in the database to get the best match. We use a statistical correlation-based method to measure the correlation coefficient between the unknown and known images, with the highest correlation indicating the best match. We use a database of 36 alphanumeric characters (26 letters and 10 numbers) in a 38 × 20 block. Some errors can occur during the recognition phase because of similarities such as O and 0. To reduce this error, determining the difference between O and 0 involves some special properties of each character, such as the aspect ratio of the character's horizontal to vertical length.

EXPERIMENTAL RESULTS

To measure our approach's accuracy, we performed experiments on 40 different models of cars with different shapes, sizes, and colors of plates under varying lighting conditions and distances. The input colored images were 400 × 300 pixels, and we took 250 license plate images. Table 1 summarizes the results.

Our technique achieved commendable results: plate localization reached 97.33 percent accuracy, character segmentation reached 95.93 percent, and character recognition reached 95.6 percent. Table 2 compares the accuracy of plate localization, character segmentation, character recognition using existing methods, and our method, which gives higher accuracy. These results indicate our proposed technique's effectiveness in real-world applications.

Our approach is based on single-level wavelet transform, and we have proven it to be efficient in varying conditions and therefore useful for real applications. In our future work, we shall try to devise a technique for skewed and distorted license plates. ■

ACKNOWLEDGMENTS

Special thanks to the Head of Electronics & Instrumentation Services Division, Bhabha Atomic Research Centre, who provided the requisite facilities for this research.

REFERENCES

1. C.N.E. Anagnostopoulos et al., "A License Plate-Recognition Algorithm for Intelligent Transportation System Applications," *IEEE Trans. Intelligent Transportation Systems*, vol. 7, no. 3, 2006, pp. 377–392.
2. V. Kamat and S. Ganesan, "An Efficient Implementation of the Hough Transform for Detecting Vehicle License Plates Using DSP's, Cellular Neural Networks and Their Applications," *Proc. IEEE Int'l Carnahan Conf.*, 1997, pp. 209–218.
3. C.A. Rahman, W. Badawy, and A. Radmanesh, "A Real-Time Vehicle's License Plate Recognition System," *Proc. IEEE Conf. Advanced Video and Signal Based Surveillance* (IVSS 03), 2003, pp. 163–166.

TABLE 2. Performance comparison of some existing license plate recognition (LPR) systems with our LPR system.

Methods	Plate localization rate (%)	Character segmentation rate (%)	Character recognition rate (%)	Total rate (%)
Y.-C. Chiou and colleagues[*]	96.2	—	—	95
M.-L. Wang and colleagues.[†]	99	—	98	80
K. Deb and colleagues[‡]	92.4	—	—	—
D.-J. Kang[§]	—	84.5	—	—
Our method	97.6	97.8	96	96.4

[*] Y.-C. Chiou et al., "Optimal Locations of License Plate Recognition to Enhance the Origin-Destination Matrix Estimation," *Proc. Eastern Asia Soc. Transportation Studies*, vol. 8, 2011, pp. 1–14.

[†] M.-L. Wang et al., "A Vehicle License Plate Recognition System Based on Spatial/Frequency Domain Filtering and Neural Networks," *Proc. 2nd Int'l Conf. Computational Collective Intelligence: Technologies and Applications* (ICCI 10), LNCS 6423, Springer, 2010, pp. 63–70.

[‡] K. Deb, H.-U. Chae, and K.-H. Jo, "Vehicle License Plate Detection Method Based on Sliding Concentric Windows and Histogram," *J. Computing*, vol. 4, no. 8, 2009, pp. 771–777.

[§] D.-J. Kang, "Dynamic Programming-Based Method for Extraction of License Plate Numbers of Speeding Vehicle on the Highway," *Int'l J. Automotive Tech.*, vol. 10, no. 2, 2009, pp. 205–210.

ABOUT THE AUTHORS

HITESH RAJPUT received a PhD in image processing from the Indian Institute of Technology (Banaras Hindu University), Varanasi. Her research interests include software design, image processing, and control systems. Contact her at hrajput.rs.apm@iitbhu.ac.in.

TANMOY SOM is a professor in the Department of Mathematical Sciences at the Indian Institute of Technology, Varanasi. His research interests include functional analysis, optimization theory, mathematical modelling, fuzzy set theory, soft set theory, and pattern recognition. Som received a PhD in mathematics from Banaras Hindu University. Contact him at tsom.apm@iitbhu.ac.in.

SOUMITRA KAR is scientist and head of the Image Processing and Machine Vision Section at the Bhabha Atomic Research Centre, Mumbai, India. His research interests include image processing, machine vision, and biometrics. Kar received a PhD in image processing from the Indian Institute of Technology (Bombay). Contact him at skar@barc.gov.in.