Automatic License Plate Recognition

Jason Grant
Department of Computer Science and Engineering,
University of Notre Dame,
Notre Dame, IN 46556
jgrant3@nd.edu

1 Introduction

Automatic license/number plate recognition is a specific application of optical character recognition. Typically employed by law enforcement agencies, the uses for automatic license plate recognition have grown tremendously since its inception. Automatic license plate recognition may be used to cite individuals who violate traffic signals or drive in excess of the speed limit, as a method of electronic toll collection, to place a suspect at a scene, or identify uninsured motorist (when combined with a database search).

License plate recognition may be complicated by frames that obscure parts of the plate, debris, complex backgrounds, and a wide variety of fonts. Furthermore, license plates are not configured in a standard format; license plates typically vary across issuing states and countries. Recognition systems must also account for rotation in the plane if a license plate is improperly mounted (this is more likely to occur after an accident). Difficulties also occur in the acquisition phase. Many systems acquire license plate images in the near-infrared spectrum and providing proper illumination is challenging. Also, cameras must be equipped with extremely shutter speeds to capture objects moving at very fast speeds. Motion blur and noise make character recognition (and most other areas of computer vision) difficult. Acquisition devices may only have one chance to capture the target, so it is imperative that images are of good quality.

2 Background

2.1 Optical Character Recognition

Optical character recognition (OCR) is the electronic conversion of handwritten, typewritten, or printed text from still or motion images to machine-encoded text. Common uses include scanning of books for electronic retrieval or scanning to edit documents electronically. OCR dates back to the late 1920s when Gustav Tauschek first obtained a patent in Germany. Today, machine printed OCR is regarded as a “solved problem.” Researchers are challenged to conduct OCR on complex images which include complicated backgrounds, degradation, low resolution, rotation and such. Many of these issues are what make license plate recognition quite challenging. Hand-printed and cursive writing, and East Asian and Latin-script characters are also regarded as complex tasks in OCR.

2.2 Automatic License Plate Detection

Though the applications of automatic license plate detection have emerged in the last decade or so, the technology has been present for nearly 30 years. In the late 1970s, researchers for the United Kingdom police department manufactured the first working license plate recognition system and began deploying it by the beginning of the 1980s. The first US patent for an
Figure 1: The diagram is taken from 1989 patent from Gonzalez et al. on an apparatus for reading a license plate.

Automatic license plate reader was issued in 1989. In a rare break for computer science, some issuing states/countries around the world adapted their license plates to assist automatic license plate recognition systems. In 2003, the Netherlands modified their license plate by introducing a new a typeset font. During the same year, Texas passed a bill banning novelty frames (later overturned in 2007) because they impeded the view of the license plate to recognition systems.

3 Experimental Methods

Automatic License plate recognition generally incurs four steps. The first step is to create a binary image of the frame. This may be done using thresholding, edge detection, or a variant. The second procedure is finding the actual plate in the image. At this step, rectification and alignment may also be done. Afterward, letters are segmented from the license plate. Finally, recognition occurs on the segmented images. Below outlines my procedures in attempting OCR.

3.1 Create Binary Image

Many of the systems that capture license plates do so in the near-infrared spectrum. Therefore, the image was first converted from color to grayscale. This was also necessary to create a binary image of the input source. There were several approaches that previous researches have taken to create a binary image. Adaptive thresholding and Canny edge detection yielded the best results. Other methods included black hat morphology and difference of Gaussians. I opted for adaptive thresholding because it produced solid characters instead of the contours with edge detection.
Figure 2: This displays the output of an image after adaptive thresholding (using the inverse binary method). Lines generated from the probabilistic Hough transform are overlayed on the resulting image from adaptive thresholding. Observing the region around the license plate, one can notice that the ends of the line do not meet. Lines have also been dilated, and still did not make contact.

Figure 3: The image on the left illustrates an example of global thresholding that works well. This is due to the high contrast between the license plate and the color of the car. As one can see, this method fails against low contrast license plates and cars.

3.2 Segment License Plate Bounding Box

Using the output of either the Canny edge detector or from adaptive thresholding, the probabilistic Hough transform can be applied to the image to find line segments. Ideally, connected line segments would outline the license plate area. Using a corner detector, the four connected corners would form the outline of outline of a rectangle. Examples are shown in Figure 2.

The second method of finding the license plate in the image is a simpler approach. Using a global threshold and simple morphological operators, the license plate box can be isolated. This method works with a limited data set (either for a dark license plate on a light car or a light plate on a dark car). For most other configurations, this method does not seem to work so well. Figure 4 demonstrates this.
Figure 4: Characters segmented in a license plate. The image on the left has more detections because the morphology was able to fill holes caused by the adaptive filtering. This aided the blob segmenting. However, the characters $U$ and $D$ have been smeared together in the process. This would easily lead to a misdetection.

3.3 Extracting Characters from the Image

Characters were extracted from the image using a blob segmenter. Since images were normalized to the same size, the approximate height and width of the characters was known. Therefore, using the resulting image from the adaptive threshold image filter, segments of were extracted from the image. This worked well for many images. Problems occurred when letters became disjoint after thresholding. To rectify this, morphology was used to dilate the image and restore holes in the letters. Nevertheless, this also came with a price. In some images, letters became joined by their ligatures. Multi-letter samples are treated as outliers and were discarded by the system.

3.4 Using Optical Character Recognition

Since the 3rd homework assignment dealt with optical character recognition, I implemented the same technique of shape and contour matching. This method yielded a low performance in a controlled experiment, and again performed poorly in this uncontrolled environment. The application of contour matching exploits the knowledge that license plates in the United States are typeset with similar fonts (Zurich condensed font).

4 Discussion and Future Work

Though many researchers consider optical character recognition of machine printed font a solved problem, automatic license plate detection in unconstrained environments poses difficulty to researchers. Finding the border and bounding rectangle of the license plate was one of the biggest challenges in this problem. Researchers and companies often omit the details of this “secret sauce” step. Methods used in this paper have produced a some what accurate segmentation of the individual characters in the license plate. To further verify the accuracy of these selections, I would explore a voting method for the row on which characters were found. Rows with the largest number of votes would assumed to be the row on which the license plate was found. Another option would be to consider the method used by Busch et al. and to count the number of switches between black and white pixels on each line of a binary image. Regions with a higher number of switches would indicate a potential for a license plate segment.
References


