OBJECTS EXTRACTION FOR LICENSE PLATE DETECTION

Charbel Fares
Holy Spirit University of Kaslik, Lebanon
charbelfares@usek.edu.lb

Abstract

In this paper we are presenting an intelligent system to extract objects in an image. We have to specify the shapes that need to be detected as well as its dimensions. Examples are given in the radar application in order to detect license plate in a radar image. A comparative study is done in order to show the most effective contour detection method for this application. The masks used for this comparison are: Sobel, Canny, Prewitt, Roberts, Log and Zerocorss. The study shows that Sobel is the most effective for such type of applications. Then three methods were used for the object extraction. Two known methods were used: Hough Transform and Watershed. The third method is a new Hybrid Segmentation algorithm. Results show that the proposed algorithm is the fastest with a time of execution of 0.8 seconds for images of size 658x486 pixels.

Keywords: *Object Extraction, Watershed, Hough Transform.*

1. INTRODUCTION AND OBJECTIVES

The goal of our work is to provide a system that detects primitive objects in an image such as circles, squares, rectangles and triangles. In particular, we did create a system for the detection of the license plate of vehicles. The auto detection of the vehicles license plates is a technique that is becoming increasingly popular and used in different situations. In addition, there are several systems that are already created such as Optasia [1] and Zamir [2]. With such systems we can monitor the speed of vehicles and each vehicle exceeding the speed limit will be automatically identified. On the other hand, if vehicle is stolen or requested by the police, the system can help in identifying the car location. System can also help in identifying vehicles to pay for the usage of highways. It can also help in ports; this system is used to identify trucks and assign them suitable containers for transportation.

One of the limitations of the system is the position of the license plate. Another problem that arises is the proper marking of characters and edges according to the law of each country. Some cases are resolved by the system while other solutions are impossible since it leads to an increase in time and complexity. Processing by the system imposes the following requirements: Adaptation to climate and light; Detection of the license plate even when acquired images are not clear; Accept some morphological changes of the plate.

Unfortunately, manual processing is a must if the system cannot handle plates in bad conditions. What is also essential to mention is that the system is "intelligent" in a way that it validate the selection of the license plate only when we have 100% match.

The rest of the paper is structured as follows: in section 2 we will discuss the Hough Transform in its standard and modified version while the Watershed algorithm is discussed in section 3. In section 4 a new hybrid algorithm is shown. Finally Conclusions and future works are shown in section 5.

2. HOUGH TRANSFORM

The Hough transform is a widely used technique in image processing. The purpose of this technique is to find simple shapes (circle, square, rectangle, triangle, etc.) in an image. In 1962, the classical Hough transform was implemented by Paul Hough. It was used to identify lines in an image [3]. In 1972, Richard Duda and Peter Hart have implemented "The generalized Hough transform" that identifies specific shapes in an image [4].

In this paper we present two methods of the Hough transform: the classical or standard and the modified or generalized Hough transform. Both are applied to image superposed with the contour of objects. Contour was detected after grayscale transformation and using different masks: Sobel, Canny, Prewitt, Roberts, Log and Zerocorss [5]. Figure 1 shows the different results. A comparative study showing the execution time is shown in table 1. Algorithms were tested on a 2.2 GHz dual core processor.

According to the results, we find that the methods "Canny", "Log" and "Zerocross" detect many unnecessary details, which lead to an increase in the execution time of the Hough transform. In addition, the outline of objects detected by the method of "Roberts" is discontinuous and this is not usable for this proposed system. Since "Sobel" and "Prewitt" methods detect the outline of objects showing a clear contrast, they are the most effective techniques.

Almost all methods take the same execution time except the method of "Canny" which takes more time. Both methods "Sobel" and "Prewitt" use two filters of size 3x3 pixels, one of which is designed to detect the horizontal difference and the other vertical difference. But the only difference is that "Sobel" has a double weight in the center of the order of 2, which is more effective in practice. For this in the following, we use the "Sobel" method.

Table 1: Execution Time of the Algorithms of Contour Detection

Algorithm	Execution Time (s)
Sobel	0.1280
Canny	0.7370
Prewitt	0.1240
Roberts	0.1260
Log	0.1800
Zerocross	0.1800

2.1 Classical Hough Transform

First, we must create the filter that contains the shape of the object we are searching for. We specify its type (circle, square, rectangle, triangle, etc...) and its dimensions. For example, to create a circle we specify the radius in pixels, to create a rectangle we specify its length and width in pixels, or to create a square we specify its side in pixels. Now the image is ready for the Hough transform. The convolution is used. We travel throughout the image using the filter, and at each position we determine the number of pixels in common with the filter and we put in a new matrix in the same position. Finally, to decide the position of the object or objects, we have several choices. Two methods are commonly used. Either we choose the largest number and its position will be sought, or we choose all positions whose number exceeds a threshold value.

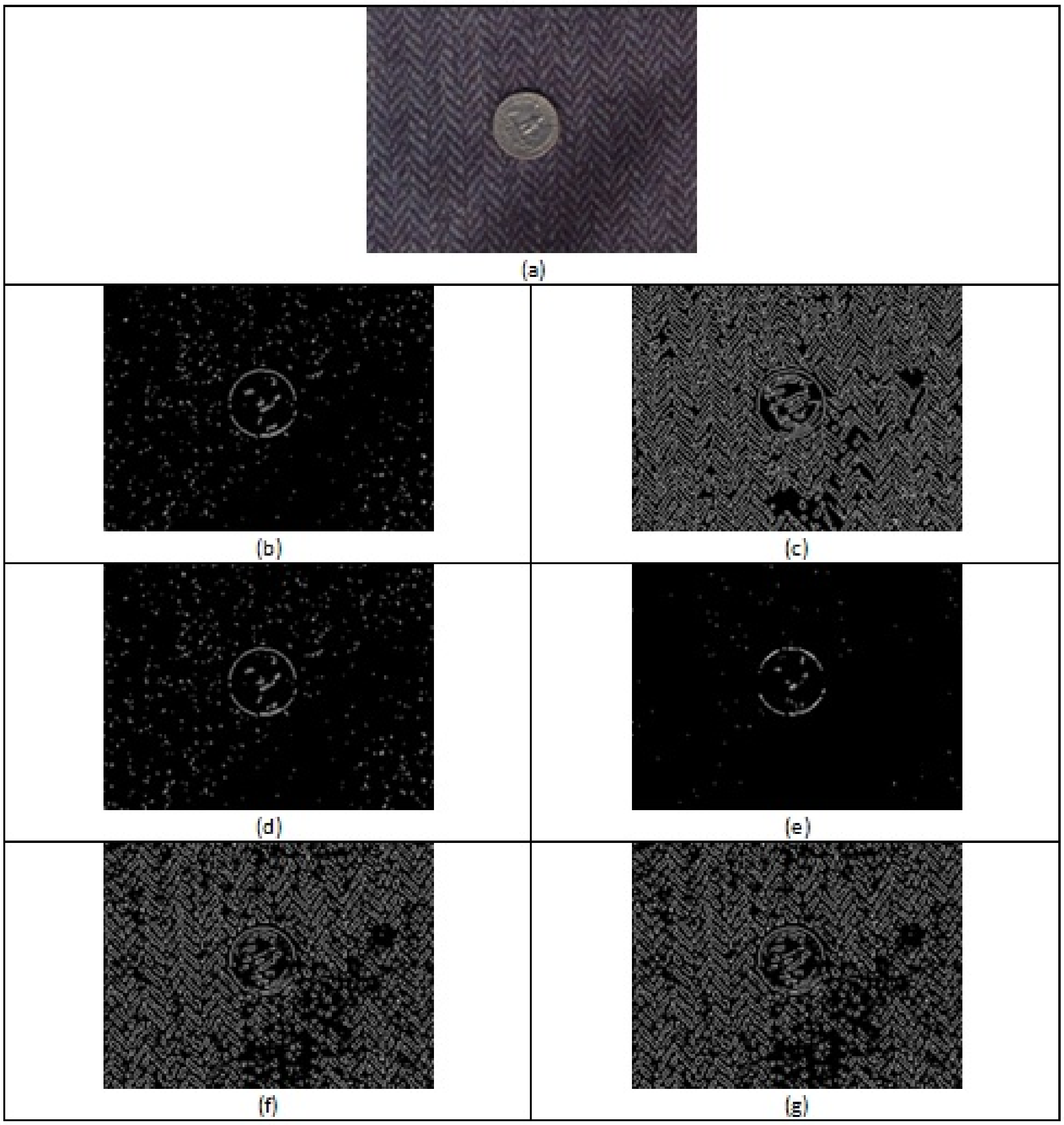

Figure 1: Detection of Contour: (a) Original image, (b) Sobel,(c) Canny, (d) Prewitt, (e) Roberts, (f) Log, (g) Zerocross

2.2 Modified Hough Transform

After creating the filter and detecting the image contour, we pass thourgh the image. At the position where there is an edge, we add the filter (the shape to be found) to a new image of the same size as the original image and at the same position. Finally, to detect the position of the object, we cross for example a contour point p of the filter, the position having the greatest value will be the point symmetrical relative to the center p.

2.3 Hough Transform Differences

Both methods give the same results for the same parameters. But instead of working on the whole image by the filter (the classical Hough), we cover only the outline (modified Hough). The advantage of the modified Hough transform is that the execution time of the program is much less than in the first method (dependings on the contour). The system was tested on a 2.2GHz dual core processor and we measured the execution time for the two methods, knowing that the original image is sized 425x566 pixels for. The first method took 10.4830s, and the second method took 1.2330s. We repeated the test on an image of 213x283 pixels dimension. The first method took 1.1860s, and the second method took 0.3430s.
The original colored image is shown in figure 2-(a). We transform it into grayscale. Then, we apply a Gaussian filter to the image to blur the face as shown in figure 2-(b). This step serves to minimize contrast from the light and shadows. After that, we minimize the dimensions of the image from 2592x1944 pixels to 324x243 pixels. Then we detect the contour using "'Sobel" method. Results are shown in figure 2-(c). We will use a margin size for the filter with width varying between 22 pixels and 24 pixels and length varying between 48 pixels and 50 pixels. To this we add zeros so that all filters traverse all edges in the image. Results are shown in figure 2-(d). Now the image is ready for the Hough transform. After completing nine filters, we apply them while observing the results obtained. Then, we seek the position with the largest number knowing that more than 50% of the number of pixels of Figure 2-(e) filter. Finally, we extract license plate as shown in Figure 2-(f).

2.4 Hough: Advantages and Disavantages

The Hough transform is a simple and very effective way to find very specific shapes in an image, especially the rectangular license plates. But sometimes we do not know the exact size of the object, we will be forced to use a size range, and for each size we need a different filter which increases the execution time. In the detection of the license plate, the execution time is critical. Using a 2.2GHz processor with a safety margin of 1 pixel (for nine different filters), the operation took 3.6560s. If the license plate is not rectangular or the taken picture is not in the right position, this technique will be less effective. Creating the filter is more complex. In addition, if we do not know the position of the camera, we can not determine the dimensions of the filter. We will have to use a great number of filters and the execution time becomes very large and inefficient.

3. WATERSHED SEGMENTATION

The Watershed segmentation algorithm applies to a grayscale image which is seen as topographic relief. This image is usually a gradient used to identify areas of homogeneous and heterogeneous input signal [6]. There are several ways to implement the Watershed technique. The algorithm we have chosen is an immersion Vincent-Soille.
Imagine that the water squirts out of each minimum and the surface is flooded from these sources. Gradually, the water level rises at a constant speed. To prevent mixing of waters from different minima, we create a dam in each elementary point of contact. Water continues to rise at a constant speed. At the end of the immersion, all dams is the dividing line of the waters [7]. The original image is transformed into an grayscal image. We first calculate the gradient of the image. Then applied the implemented Watershed as connectivity parameter 8. The result thus obtained is over-segmented. This is due to the large number of local minima present in the image and because of the noise. So to overcome this problem, we use a marker. The solution of the Watershed is constrained by markers to avoid creating pools at local minima, it is necessary to make a change on the homotopy of the input structure that is to put the marked areas at lowest structure. This removes information that are of no interest for the system. In our project, the markers of the image are obtained from the global thresholding. A direct application of the transformation of the image gradient Watershed leads to over-segmentation due to noise and detection regions too. So one approach is used to avoid over-segmentation, this approach is based on the concept of markers. A marker is a connected region. There are two types of markers: inside marker (within the object) and outside marker (background). These markers are used to modify the gradient image by imposing the places where the markers as indoor and outdoor areas minima.

3.1 How to Get Inside and Outside Markers

To eliminate minima that are not interesting, we use extended minima regions using global thresholding. Thresholding operation is used to separate an image into two sets of objects. This operation is to set to zero all pixels having a gray level below a certain value (threshold) and the maximum pixels with a higher value. Thus the result of thresholding is a binary image containing black and white pixels. Example of an algorithm for automatic global thresholding:
- Choose an initial T (mean,...);
- We obtain 2 groups G1 if $f(x, y) > T$ & G2 if $f(x, y) < T$;
- Calculate the average gray levels for G1 & G2 namely Y1 and Y2;
- The new T = 1/2 (Y1 + Y2);
- Repeat until T is almost constant.

This operation is used for inside markers. Outside markers are pixels between inside markers.
Two examples are shown. The first example is shown in figure 3. The original image is shown in figure 3-(a). We convert it to grayscale in order to apply Watershed. We identify inside markers using the global thresholding technique after smoothing. The result is shown in figure 3-(b). We deduce the outer markers after dilation with two structuring element of 5x5 dimension (one horizontal and another vertical). This is shown in figure 3-(c). We note that the number plate of the vehicle is divided into several adjacent parts. Finally we extract the original image as shown in figure 3-(d).
The second example is shown in figure 4. The original image is shown in figure 4-(a).We convert it to grayscale. We identify inside markers. The result is shown in figure 4-(b). We deduce the outer markers after dilation with two structuring element as in the previous example. This is shown in figure 4-(c). Finally we extract the original image as shown in figure 4-(d).

The line of watershed provides first by definition closed contours. In fact, this technique allows the detection of homogeneous regions in the image or the marking of these areas are spreading. This method produces an over segmentation because of the high sensitivity to noise, and from memory, it needs a large memory for example 3.5 * N to 11 * N bytes for an image with N pixels, in addition, it is not feasible in real time. The execution time of an image of size 512x512 pixels on a 2.2 GHz processor is shown in table 2.

4. HYBRID SEGMENTATION

This section introduces a segmentation method used to detect each object having a contrast with the background of the image. This

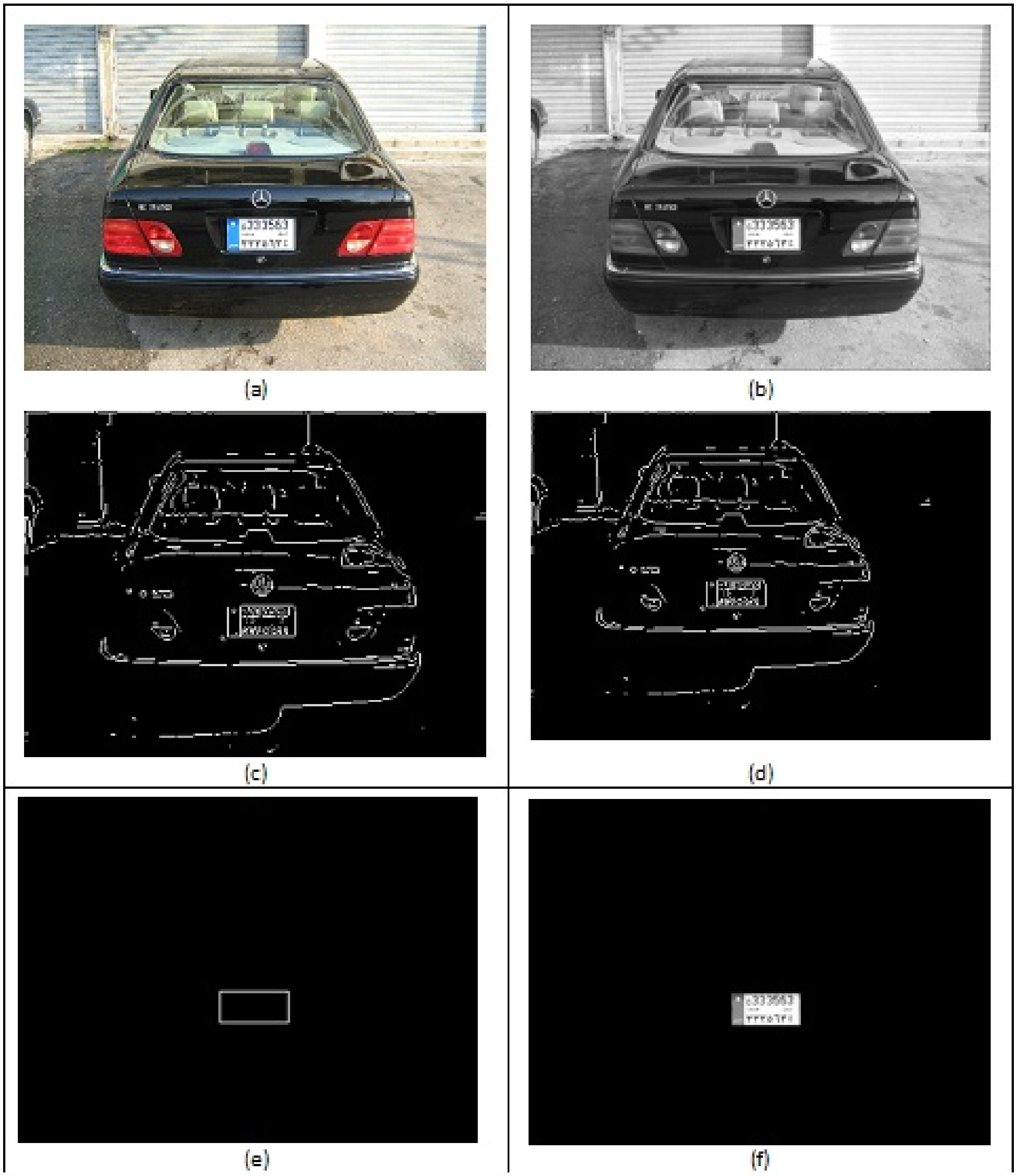

Figure 2: Hough Transform: (a) Original image, (b) Gaussian Filter, (c) Sobel Contour Detection, (d) Adding Zeros, (e) Used Filter, (f) Detection of the License Plate.

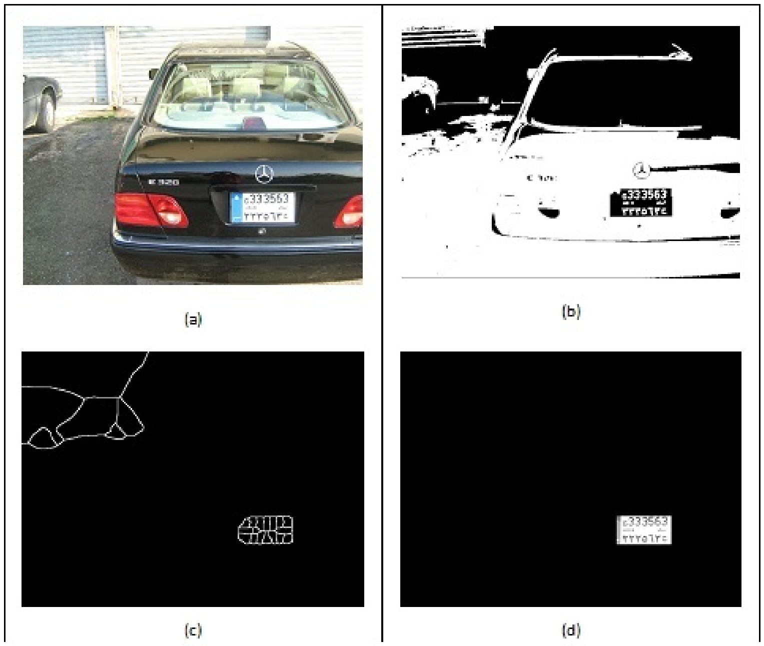

Figure 3: Watershed Algorithm: a) Original Image, b) Markers interior after smoothing, c) external markers after dilation with a structuring element horizontal and vertical size 5x5, d) Detection of the car plate

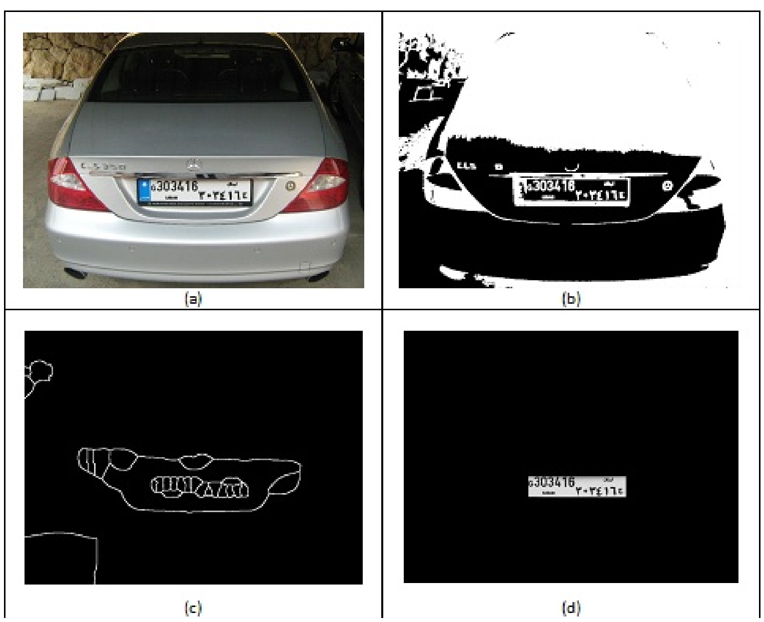

Figure 4: Watershed Algorithm: a) Original Image, b) Markers interior after smoothing, c) external markers after dilation with a structuring element horizontal and vertical size 5x5, d) Detection of the car plate

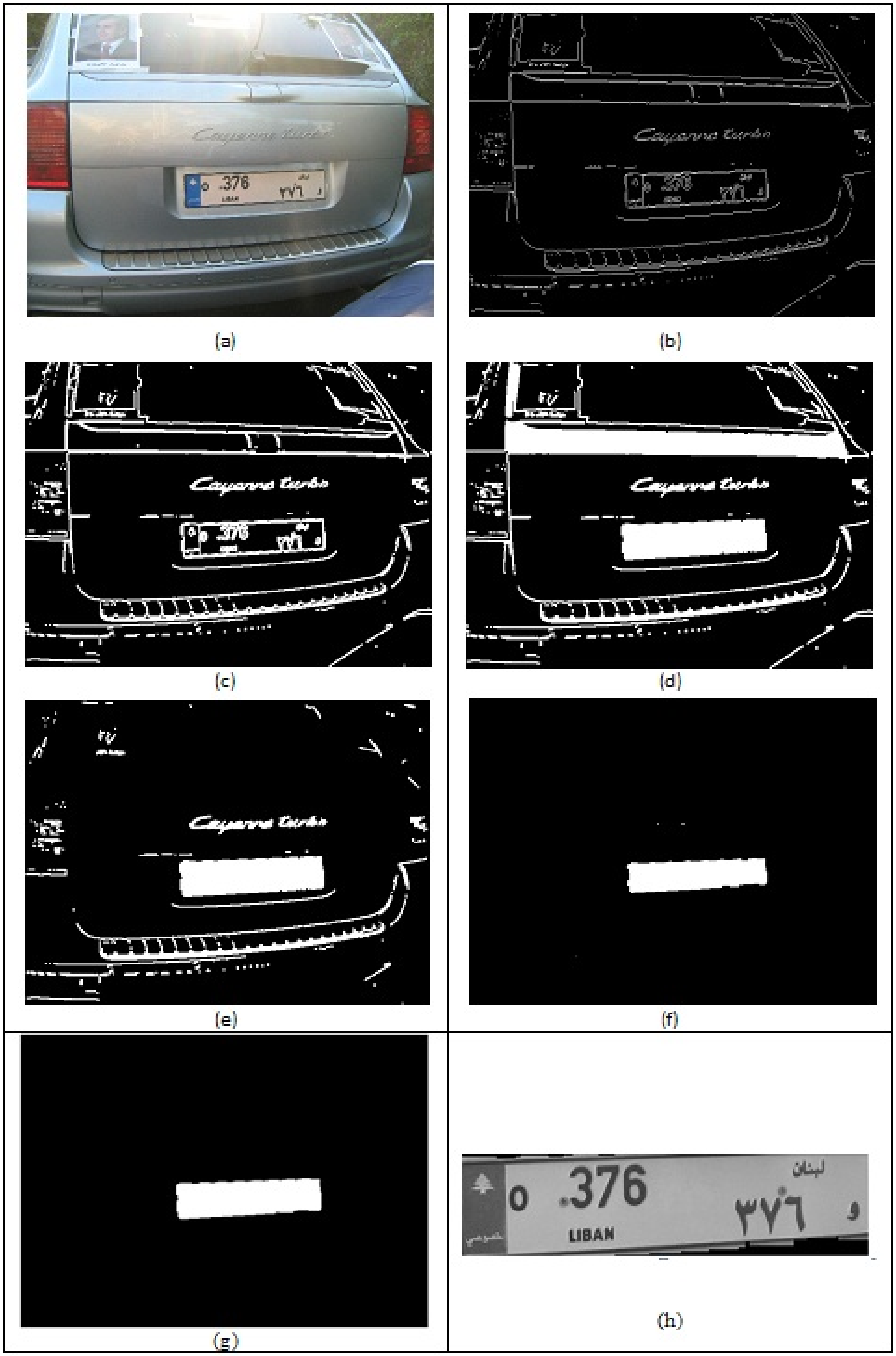

Figure 5: Hybrid Algorithm: a) Original Image, b) Sobel Contour Detection, c) Contour Expansion, d) Filling of Closed Contour, e) Removing objects related to the edge of the image, f) Erosion by a square of side 10 pixels, g) Objects Eliminition, h) License Plate Detection.

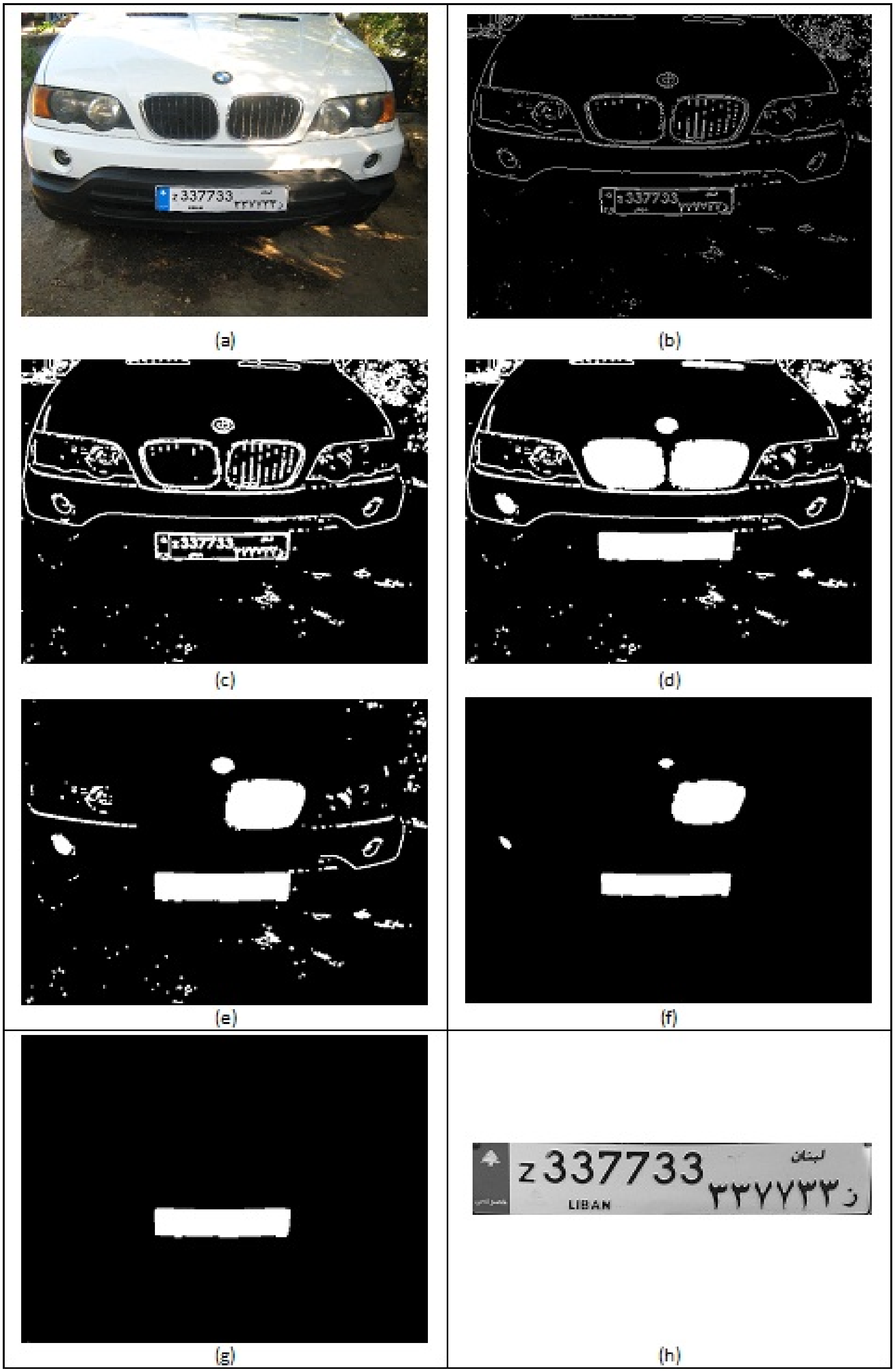

Figure 6: Hybrid Algorithm: a) Original Image, b) Sobel Contour Detection, c) Contour Expansion, d) Filling of Closed Contour, e) Removing objects related to the edge of the image, f) Erosion by a square of side 10 pixels, g) Objects Eliminition, h) License Plate Detection.

Table 2: Execution Time of Watershed Algorithm

Connectivity	Time of execution of Watershed (s)
8-Connectivity	2.13
4-Connectivity	1.68
Filtered Image with 8-Connectivity	1.84

technique is used in automated microscopes to see the existence of specific types of cells. We have modified this technique so that it is effective for detecting license plate of vehicles. Some of the technics used in this method are: dilatation, erosion, opening [8].
First, we convert the original image into grayscale. The we minimize its dimensions so that the program can run faster. Then we use the Sobel method to extract the contour. We continue by applying two expansions by two structural elements of size 1x3 and 3x1 respectively with 1 as value for all the elemets of both vectors. The role of these expansions is to connect the contours in case of discontinuity. After this step, we fill inside closed contours by the value 1 and we eliminate the related objects at the edge of the image. After that, we apply an opening (erosion and dilation) by the same structuring element which is a rectangle of length m and width b. This opening is to eliminate objects that are smaller than plate vehicles. At this stage we might get an image that contains more than one object. If we got one or more objects we consider each object separately, and if its shape and dimensions verify the form of a license plate we extract this shape from the original image. If we do not get an object, we apply a rotation to the image in the case where it is tilted according to the camera.
Two examples are shown. In the first example we are working on the license plate of the back of the car. The original image is shown in figure 5-(a). The image is then converted to grayscale and its dimensions are reduced from 2492x1944 pixels to 648x486 pixels. The contour is then detected using the Sobel Algorithm as shown in figure 5-(b). At this stage, we dilate the contour by two dilations as shown in figure 5-(c). We fill the interior of closed contours with 1 (white color). Results are shown in figure 5-(d). We eliminate objects that are connected to the edge as shown in figure 5-(e). After that, we apply an erosion by a structuring element, which is a square of side 10 pixels, and we get an image that contains four objects as shown in figure 5-(f). At this stage, we consider each object separately, and if its shape and dimensions do not map the form of a license plate we eliminate. Then, we apply a dilation by the same structuring element which is a square of side 10 pixels.Results are displayed in figure 5-(g). Finally, any remaining object will be extracted from the original image in grayscale. In this example we obtained a single object that is the vehicle License Plate as shown in figure 5-(h).
The second example is the detection od the license plate of the front of the vehicle. The original image is shown in figure 6-(a). The image is then converted to grayscale and its dimensions are reduced from 2492x1944 pixels to 648x486 pixels. The contour is then detected using the Sobel Algorithm as shown in figure 6-(b). At this stage, we dilate the contour by two dilations as shown in figure 6-(c). We fill the interior of closed contours with 1 (white color). Results are shown in figure 6-(d). We eliminate objects that are connected to the edge as shown in figure 6-(e). After that, we apply an erosion by a structuring element, which is a square of side 10 pixels, and we get an image that contains four objects as shown in figure 6-(f). At this stage, we consider each object separately, and if its shape and dimensions do not map the form of a license plate we eliminate. Then, we apply a dilation by the same structuring element which is a square of side 10 pixels.Results are displayed in figure 6-(g). Finally, any remaining object will be extracted from the original image in grayscale. In this example we obtained a single object that is the vehicle License Plate as shown in figure 6-(h).

4.1 Benefits and Drawbacks

This method is very fast, the execution time of the program is much less than the two methods already seen. (The Hough transform and the watershed segmentation). We tried the examples on a 2.2GHz dual core processor, for an image size of 648x486 pixels and the execution time was 0.8012s. In addition, we tested examples on the same processor but for an image size of 324x243 pixels and the execution time was 0.3374s. In this method we could detect the inclined license plates and as well as the one photographed at different positions. The only disadvantage of this method is that sometimes we get closed contours with the license plate inside. In this case we have to take this object and apply the same method again to extract the plate: this leads to a double execution time. This method is the most effective of all the methods seen. We obtained a yield of 95% for size 648x486 pixels images and 50% yield for 324x243 pixel sized images.

5. CONCLUSIONS AND PERSPECTIVES

Our project is to extract primitive objects in an image. Three methods were shown for this extraction. The first is that of the Hough transform. This method is slow in detecting license plate of vehicles because we have to use at least nine filters. The execution time for this method is 3.6560s. In addition, if we do not know the dimensions of the license plate, the execution time becomes worse. Add to this, if the vehicle is not photographed at 90 degree angle the filter becomes more complex. The second is the watershed. This method is slow by comparison with the first. It will lead to over segmentation because it is sensitive to noise, and in addition, it is not executable in real time and it needs a large memory. The third is that the hybrid segmentation. It is the fastest, we can detect the plate in less than 0.8s with an efficiency of 95%. In addition, it allows the detection of an object tilted and adjusted.
We can imagine a continuation of our project by proposing a system that verifies that the detected object is the plate or another object identified among several objects detected.

Acknowledgments: *part of this work was done by Rodny Zgheib and Maya Sahyoun as a final year engineering project.*

6. REFERENCES

[1] License Plate Recognition System, Optasia. Official Website.

[2] Zamir Recognition Systems Ltd.,2006. Official Website.

[3] Hough P.V.C., "Method and Means for Recognizing Complex Patterns", U.S. Patent 3 069 654, December 18, 1962.

[4] Duda R.O., and Hart P.E., "Use of the Hough Transform to Detect Lines and Curves in Pictures", CACM, Vol. 15, No. 1, January 1972.

[5] Gonzalez R.C. and Woods R. E., "Digital Image Processing", third Edition, 2008 by Pearson.

[6] Beucher S. and Meyer F., "The morphological approach to segmentation: The watershed transformation", in Mathematical Morphology in Image Processing, E. Dougherty, Ed. New York: Marcel Dekker, 1992.

[7] Vincent L. and Soille P., "Watersheds in Digital Spaces : An Efficient Algorithm Based on Immersion Simulations", IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 13, No. 6, 1991.

[8] Beucher S. and Meyer F., "Mathematical morphology in image processing. The morphological approach to segmentation: the watershed transformation" (E. Dougherty, ed.), Marcel Dekker, 1993.