8-17

MVA2007 IAPR Conference on Machine Vision Applications, May 16-18, 2007, Tokyo, JAPAN

Objects Recognition in Real Time via Edge Point Analysis

Nauman Iftikhar
Military College of Signals
NUST Rawalpindi Pakistan
Tamizaudin Road Rawalpindi Pakistan
nsatti@gmail.com

Saba Iqbal
Military College of Signals
NUST Rawalpindi Pakistan
Tamizaudin Road Rawalpindi Pakistan
sabanauman@gmail.com

Naveed Sarfraz Khattak
Military College of Signals
NUST Rawalpindi Pakistan
Tamizaudin Road Rawalpindi Pakistan
naveedkhattak@mcs.edu.pk

Abstract

Matching strategy plays a decisive role for the success of any particular algorithm and is o`1`ne of the most challenging tasks in digital image processing. The present research work exclusively automates the object recognition technique by applying digital image processing techniques and pattern analysis. Capturing scene / frame of interest from a video stream serves as a baseline for further recognition and is done by object identification, which yields a frame with moving objects. While recognition is achieved through edge point analysis against the pre-processed images already stored in data base.

1 Introduction

Object recognition techniques in digital image processing plays significant role in real time moving objects. There are many definitions for object recognition depending upon the context in which it is being applied. Matching can be defined as the establishment of the correspondence between various data sets, which is also referred as correspondence problem.

Image matching is also an ill-posed problem, as no guarantee can be given that a unique and stable solution exists with respect to small variation in the input data. Because for a given point in one image, a corresponding point may not exist due to occlusion and noise or there may be more than one possible match due to repetitive patterns and the solution may be unstable [1].

The application of current research work is vehicle tracking, traffic monitoring, measuring the number of vehicles, assisting road surveillance, traffic flow estimation and prediction. For automated traffic monitoring and analysis system, the information about vehicle type is very important. According to [6] [7] [8] recent Image Tracking Systems (ITS) require detailed traffic parameters. Due to these detailed parameters input becomes extensively heavy so it affects the efficiency of the system. This paper proposes a technique for vehicle-type recognition which focuses attention on the structure of vehicles and needs only global information which requires minimal input information, less computation power and is robust against small variations such as rotation, scaling and color of vehicle. Even it is equally efficient for low or high luminous condition.

In proposed system first make set of training images for each vehicle family by taking the point information, make and model in order to create a template for its family and then store this information in database. The algorithm selects frame of interest and identifies moving objects, than finally recognize objects through edge point analysis. It has been observed upon more than 400 vehicles of different families that vehicle distances between their edge points are different. For vehicles of different companies like Nissan, Toyota all edge point distances are different. Therefore, we categorize vehicles into different classes. Class A consists of large size vehicles like jeeps, wagons etc and class B consists of medium size vehicles like Honda Civic, Toyota Corolla etc and class C consists of small size vehicles like Suzuki Alto and Hyundai Santro etc.

Rest of the paper is organized as follows. Section 2 elaborates the steps for the template creation process. Section 3 address the method used to capture frame of interest (Query Frame) using Critical Area Analysis. Section 4 explains the technique used for location of object. Section 5 explains the method used for classification. Section 6 shows results of experiments made with typical images while section 7 presents the main conclusions and points to future work.

2 Training Image Creation

For edge point analysis there should be some predefined training images on the basis of which matching is performed. To create training images for each vehicle type, load the image and click on the edge points get x, y coordinates of these points. Now simply calculate the x and y distances between these points. Figure 1 show these points and table 1 illustrates the formula for calculation of distance at the time of saving and retrieval. For example the distance for ‘x’ coordinate of point ‘a’ and point ‘b’ is almost same so it is constant and names it as xdist. But the y coordinate is not constant so calculating the difference between y coordinates of point ‘a’ and point ‘b’ yields ydist. The value of x dist will give the distance between x coordinates of any two points. Similarly, the value of y dist will give the distance between y coordinates of any two points. So at time of template matching to locate point ‘b’ add ydist value to y coordinate value of point ‘a’. For point ‘a’ and point ‘c’ value of x coordinate is different so compute xdist by subtracting value for x of point ‘a’ from point ‘c’. For template matching at time of locating point ‘c’ add xdist to x coordinate value of point ‘a’. On the other to locate point ‘d’ from point ‘a’ add both xdist and ydist in x and y coordinate value of point ‘a’ respectively.

Figure 1: Image showing the points marked for template creation.

Equation 1: Showing the formulas for edge point calculation at the time of creating training image and matching query image with edge point.

Creating	**Matching**
Xdist= X_2 - X_1	F(x) =Xdist + X of given point.
Ydist= Y_2 - Y_1	F(x) =Ydist + Y of given point.

3 Object Identification

This section starts with the technique used to select frame of interest from a video stream. Potential object identification is also explained in this section. Frame of interest is that frame which contains the complete image of vehicle. As input is video stream and vehicle is moving with limited speed so the desire frame is that which contains the complete image of vehicle, first the subsequent change in the frame (query frame) is determined through background subtraction [2] and if the result contains a subsequent amount of non-zero pixels it means that any object is found. Then the next step is to find the vehicle occurrence in frame area and to eliminate noise that in result of background subtraction. Thresholding with alpha value 50 is applied on resultant frame [3], as shown in figure 2. If vehicle is found in first 5th row of frame, than the frame will be discarded as it is incomplete. But if it is found between the 10 to 15th row, then frame after immediate next frame is considered for further processing.

To locate vehicle, apply contouring on resultant image. The technique which is used finds 4-8 adjacent pixels for every candidate pixel, there can be two expected values of the adjacent pixels, it could be the same as previously marked portion or it contains different value so in later case it belongs to same object and in former it belongs to the different object [3] [4] [5]. Image after contouring is bounded by a rectangle by taking the height and width of the object as shown in the figure 5. After contouring there could be more than one object, not all of these objects are of our object of interest so three criteria are used to locate the object of interest from the scene.

- No. of pixels
- Area covered
- Dimension

The result after countering but prior to bounding by rectangle is shown in figure 5.

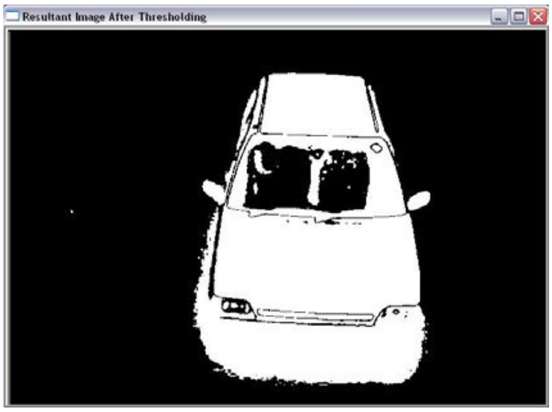

Figure 2: Resultant image after thresholding.

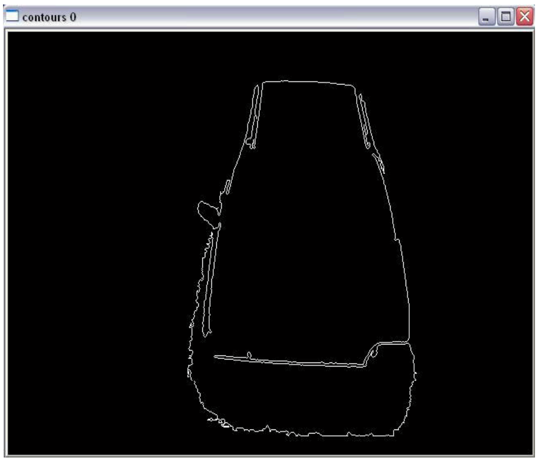

Figure 3: Extracted image after contouring before bounded by rectangle.

4 Object Recognition

Once the vehicle is identified then we have to recognize it further from which class it does belong. This section describes the algorithm used to categorize the vehicle into different classes which is achieved through edge point analysis.

4.1 Edge Points Analysis

After getting the vehicle now locate edge points of vehicle. So start with raster scanning to find first non-zero pixel, then apply certain checks to make it seed pixel. The pixels are checked in their forward and downward direction and are marked as seed pixel. Figure 4 indicates the expected locations of seed pixels which are drawn in red and blue color. There could be four possibilities.

- If the points are found in next row and column like down stair case effect towards right and then check either seed pixel lies near left top of the bounding rectangle then it is point ‘a’ and the vehicle is left rotated.
- In second case points are found below and before seed pixel like down stair case effect towards left and the seed point is located near right top then this is point ‘b’ and vehicle is right rotated.
- In the third case pixels found below towards right and left the seed pixel , check the distance of seed point from left top and right top, compare the distance and traverse towards that side whose distance is lesser than other if it is near to left top then it is point ‘a’ like down stair case effect towards right.
- In last case pixels are found next to the seed and the point found is ‘a’ and expected distance is added to compute point ‘b’.

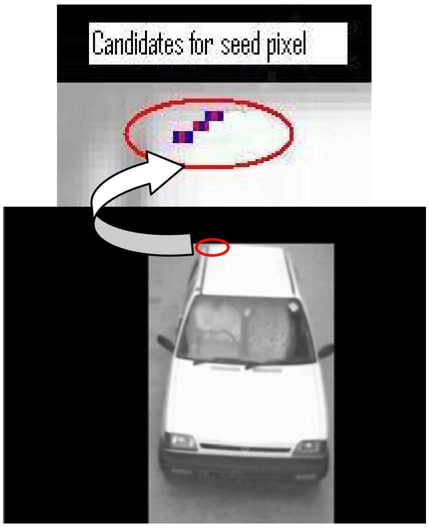

Figure 4: Indicating the expected location of seed pixel.

After locating point ‘b’ go for point ‘c’. If point ‘c’ is found then search point ‘d’. Distances of points are saved in database for every class of vehicle, we just have to get the expected distance value for each class one by one and find the next point at expected position. Point ‘a’, ‘b’, ‘c’, ‘d’ helps in classification of vehicles in to classes(A,B,C) while rest of the six points help is determination of family of the vehicle (Toyota, Honda etc) . After matching point ‘d’ class of vehicle is determined, to find its family search for point ‘e’ then for point ‘f’, ‘g’, ‘h’, ‘i’, ‘j’, ‘k’, ’l’. After class determination for every match hit counter is incremented and on the basis of number of hits class and family of vehicle is determined. If any point is not found at expected position, to locate it a circle with radius of 4 pixels is drawn, and the point could lie anywhere within that circle, these circles are marked with red color in figure 1 and these are marked as red circles by implemented system and is shown in figure 5.

If the vehicle is not recognized then it is considered to be an anonymous vehicle. Matching is always started from class A, then move on to next classes because class A consists of large size vehicles so all distances of class B and C lies within the distances of class A therefore all points of vehicles of class B and C match with class A and are considered to be the member of class A.

Scaling problem is automatically catered through critical area analysis. The technique discussed in section above for frame selection uses critical area analysis to grab the frame only when the vehicle is at its interested position to remove any problem of scaling. This procedure is explained in section 3. Figure 5 shows the image after edge point analysis. The red points on the image are the result yield during edge point analysis.

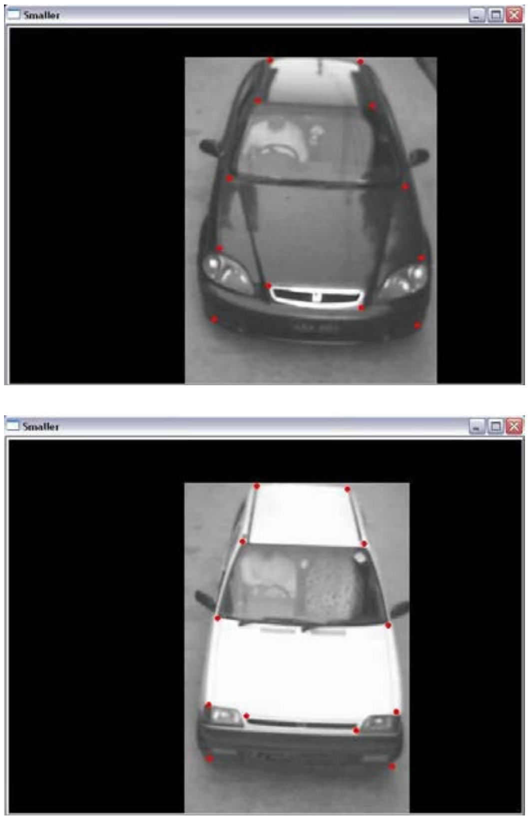

Figure 5: Image after successful match against Edge value analysis.

5 Conclusion and Future Work

Edge point analysis approach is presented in this paper. The proposed technique deals with video streams which identifies moving object from a video stream and recognize it using edge value analysis. The categorization process classifies potential objects on global features. Therefore this technique does not require extensive lowlevel information. Moreover in case of noise or changing lightening conditions, it still yields good results. The point information also assist the extraction of local features like from point K & L we can easily locate the mark up of the company which can be help full to validate the family of vehicle.

We have performed tests on 50 static vehicles belonging to different families and results of object identification are 85% and results of classification are 83% but in case of video frames we performed experiments on 16 vehicles and out of which 12 are located and classified correctly and results are 75%.

For future work based on this approach, it is recommended to take local features into consideration for accuracy of results, but it will affect the efficiency. For multiple objects, issues pertaining to occlusions can be addressed to make the technique more robust in different situations. Two cameras in conjunction to this approach can be used for vehicle speed detection in order to get the occurrence of exact position for vehicle in next camera and avoid limitation of FPS (frame per second). Use of two cameras will also increase the efficiency of algorithm. Number plate recognition, car damage inspections and decoration can be built in based on this approach for detection of suspected vehicle.

References:

[1] Christian Heipke*," Overview of Image Matching Techniques*", Chapter 1.

[2] Copyright © 2001 Intel Corporation, Open Source Computer Vision Library, Chapter 2, 2001, pp 1-3.

[3] Copyright © 2001 Intel Corporation, Open Source Computer Vision Library, Chapter 3, 2001, pp 1-15.

[4] Linda Shapiro and George Stockman, Computer Vision , Chapter 3, Mar 2000, pp 63-77.

[5] S. Suzuki, K. Abe. "Topological Structural Analysis of Digital Binary Images by Border Following". CVGIP, v.30, n.1. 1985, pp. 32-46.

[6] Frederic Jurie, Michel Dhome, Real Time Robust Template Matching , Electronic Proceedings of The 13th British Machine Vision Conference University of Cardiff 2-5 September 2002, British Machine Vision Association and Society for Pattern Recognition ,2002.

[7] Shirmila Mohottala, Masataka Kagesawa, Katsushi Ikeuchi, Vehicle Class Recognition Using 3d Cg Models, Institute of Industrial Science, University of Tokyo 4-6-1 Komaba, Meguro-ku, Tokyo, 106-8558, Japan.

[8] Tatsuya Yoshida, Masataka Kagesawa, Tetsuya Tomonaka, Katsushi Ikeuchi, Vehicle Recognition With Local-Feature Based Algorithm Using Parallel Vision Board, Institute of Industrial Science, University of Tokyo 4-6-1 Komaba, Meguro-ku, Tokyo, 153-8505, Japan.