VEHICLE DETECTION AND CLASSIFICATION BASED ON IMAGE PROCESSING TECHNIQUES

SOTHEANY NOU

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING IN COMPUTING IN ENGINEERING SYSTEMS

INTERNATIONAL COLLEGE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2017

KMITL-2017-IC-M-11-01

VEHICLE DETECTION AND CLASSIFICATION BASED ON IMAGE PROCESSING TECHNIQUES

SOTHEANY NOU

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING IN COMPUTING IN ENGINEERING SYSTEMS

INTERNATIONAL COLLEGE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2017

KMITL-2017-IC-M-11-01

Thesis Title	Vehicle Detection and Classification Based on Image Processing Techniques
Student	Sotheany Nou
Student ID	58610015
Degree	Master of Engineering
Program	Computing In Engineering Systems
Thesis Advisor	Asst.Prof.Dr. Chaiwat Nuthong

ABSTRACT

Traffic congestion is one of the major problems in the big cities. Many systems are proposed to solve this problem, for example, Intelligent Transportation System (ITS). In general, ITS consists of many subsystems including traffic monitoring. The monitoring system can provide the information of vehicles that pass through the monitoring area. For this reason, developing the traffic monitoring system with high accuracy plays important role to support ITS. In order to develop traffic monitoring system, there are other subsystems needed such as vehicle detection, vehicle classification, and traffic density measurement.

Vehicle detection subsystem needs several techniques ,i.e., background subtraction, morphological operation, and BLOB detection. Although the vehicles can be detected, they come in occluded scenarios. These scenarios result the poor vehicle counting performance and also affect to the vehicle classification. In order to solve this problem, skeleton feature extraction has been applied.

Vehicle classification can be implemented after vehicle can be well detected and occlusion scenarios is solved. In order to classify vehicle, this study presents back-propagation neural network and badial basis function network. The performance of those two neural network algorithm are compared. The experimental results illustrate that back-propagation neural network can classify vehicles with comparable performance to Radial basis function network.

The proposed system can detect vehicles in occluded scenarios and classify vehicle for various videos input with a satisfied accuracy. Moreover, traffic density measurement also give the information of the density of the monitoring zone in the road. This information can help traffic monitoring system to analyse the status of the road or monitory zone.

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my gratitude and thank to all following people who have directly and indirectly being part of this research study whose name will be listed as following.

First, I would like to express my grateful acknowledgement toward my advisor Dr. Chaiwat Nuthong of International College at King Mongkut's Institute of Technology Ladkrabang who has supported me throughout my thesis with his patience and knowledge while allowing to work in my own way. The door of his lab is always open and welcome me at any time whenever I need his help. My degree would not be successful without his guidance and support.

In addition, I profoundly thanks to all lecturers and faculty staffs in my faculty who have always been part to extend my knowledge and provide very useful tools for my research work.

Moreover, my thanks would extended to AUN Seed Net scholarship that make this opportunity possible for me and has constantly provide with all support in financially without fail to extend this research work.

I would never forget the most important people to be grateful for is my parents who being part of my encouragement without fail with both emotional and financial support during my journey from the start. The completion of this study could not have been successful without the unlimited supports and assistance from many people including my friend whose name will not mention all together in here.

Bangkok, December 2017

Sotheany Nou

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

Figure Page

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction to the research

Rising of traffic congestion is a remarkable condition in the large metropolitan areas. This problem occurs when the number of vehicles are increasing everyday with the poor road management. As shown in Figure 1.1, the road is blocked and cars are in trouble traffic because of the large amount of car in the road space. Based on this issue, surveillance cameras have been installed along the road to capture the traffic. One of the objectives of installing surveillance cameras along the road is to assist in solving traffic congestion problem.

Figure 1.1: Traffic congestion in the road

In the past, only small number of evidences show that the videos recorded by the surveillance cameras have been used to solve traffic congestion problem systematically. However, with the advancement in computer and camera technology at the present, the researchers have a vision that the information obtained from the surveillance cameras can be extracted and possibly used to assist in solving the traffic congestion problems. Hence, this research has been proposed.

1.2 Objective of the research

Traffic monitoring system is a system that can monitor the traffic to avoid traffic congestion problem. This study aims to develop a software prototype for traffic monitoring based on image processing techniques. The developed software attempts to extract information such as the classes of vehicles, number of vehicles in each class, and traffic density estimation from recorded traffic videos. This is one of the techniques to use the available data from surveillance cameras for solving traffic congestion problem.

1.3 System overview

The process to extract the traffic information for traffic monitoring system consists of three main steps as shown in Figure 1.2.

Step 1: Vehicle detection

Vehicle detection is a technique used to detect vehicles in each frame of surveillance video. Vehicle detection starts with the input video captured from surveillance camera. From the input video, system extract current image and reference image as two main components for the input of background subtraction. Background subtraction is a technique used to remove the image's background to get foreground that are the objects in the monitoring zone of the road. It is applied after current image and reference image are generated. After background subtraction process is done, the foreground received is converted to binary image by using global thresholding. Then, morphological operation is implemented to binary image of the foreground in order to remove noise or texture. The output of morphological operation is used as the input of BLOB detection. BLOB detection is the technique used to extract vehicle's object from the foreground image. So the outputs are the objects of vehicles with its information. Vehicles can be detected after BLOB detection is implemented. However, vehicle occlusion problem occur in vehicle detection. In oder to solve this occlusion problem, skeleton feature extraction is implemented. So vehicle detection step can be used by giving the better detection of vehicle's objects.

Step 2: Vehicle classification

Vehicle classification is a technique applied to category vehicles into classes. It is implemented in order to recognise the type of vehicle and then the number of vehicle in each class can be counted. Artificial neural network is implemented to classify vehicles into classes. After Vehicle classification is applied the information of the vehicle's object is obtained and can be used for traffic monitoring system.

Step 3: Traffic density measurement

Traffic density is a method for measuring the density of the traffic flow in the specific zone of the road space. The information of traffic density can be used for traffic monitoring system by analysing the traffic status in order to solve traffic congestion problem.

1.4 Thesis organisation

This thesis consists of 5 chapters such as introduction, literature review, research methodology, experimental results, and conclusion. In this chapter is the chapter of introduction and the rest are outline as follow.

Chapter 2 gives the information of literature review showed about the previous work related to vehicle detection, vehicle classification and traffic density measurement.

Chapter 3 describes about the methods used in this research study. They are the methods and technique implemented to detect vehicle and the ways of solving occluded vehicle's object in the traffic videos. Moreover, artificial neural network is implemented to classify vehicle into classes. This chapter also presents the technique of fixing the error dataset that causes the misclassification in vehicle classification step. Furthermore, traffic density measurement also declares in this chapter after vehicle classification.

Chapter 4 illustrates the experiments and results with two different input traffic videos. It shows the accuracy of the methods and techniques implemented in vehicle detection and vehicle classification.

The last chapter presents the conclusion of the thesis and future work.

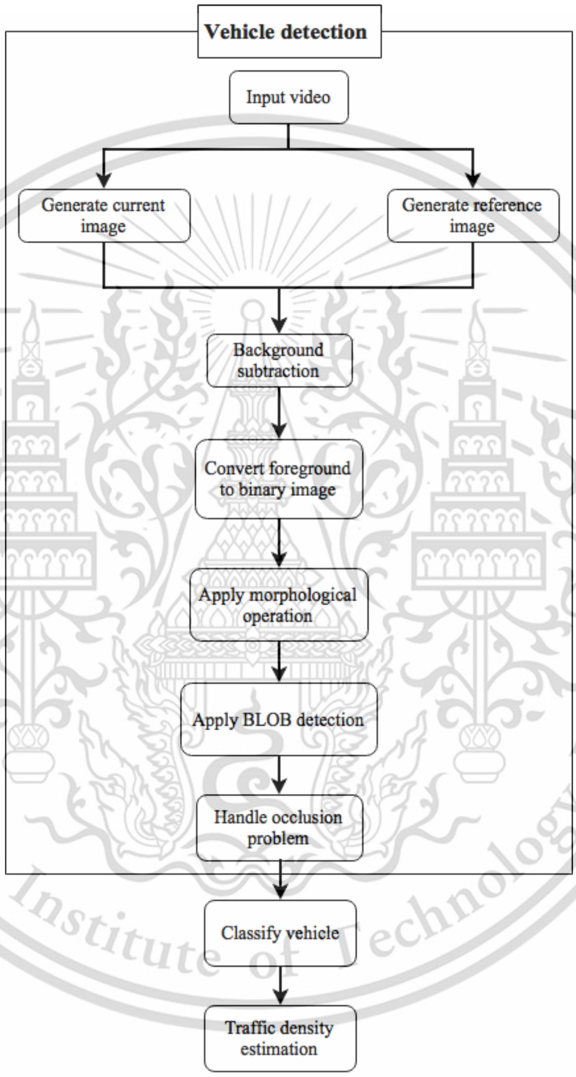


Figure 1.2: System overview

CHAPTER 2

LITERATURE REVIEW

This chapter provides an overview of previous research works. The review of existing literature is performed to support this study. The review of vehicle detection is first presented in this chapter. Next, vehicle classification is discussed followed by the last part is traffic density measurement.

2.1 Vehicle detection

Vehicle detection is one of the techniques used to extract the traffic information from the vehicle's objects in the surveillance video. There are many corresponding researches conducted with this technique to extract traffic information from in the traffic's videos. For the first research review, Rabiu [1] used background subtraction and Kalman filter algorithm to detect vehicles in the detection zone. To subtract background from the video's frame, masked background (see Figure 2.1) and masked current frame (see Figure 2.2) are needed in their system. Both masked background and masked current frame are generated by multiply the binary mask (see Figure 2.3) to the background image, which consists of only road without moving object, and current frame. Then they applied background subtraction and detect locations of pixel in the current image that have values greater than the set threshold. Those pixels are considered to belong to the object as shown in Figure 2.4 and the group the connected pixels in the vehicle object. From those techniques thier system can detect vehicle object in the video's frame. The performance of the proposed method was tested on a video sequence of 3,400 frames containing 76 vehicles. The system was successfully able to detected most vehicles correctly with approximatly 97.37%. The errors in the detection stage were mostly associated with occlusions. Because of vehicle occlusion, the result was still unsatisfied. The author put vehicle occlusion issue as the future work.

Chintalacheruvu and Muthukumar [2] implemented Harris-Stephen Corner detector Method (HSCM) in vehicle detection system. HSCM was applied to determine the corner points of the vehicle in the detection zone. For those corner points, it was used to detect the vehicles and determine vehicle counts and speeds at arterial road-ways and freeways. HSCM was implemented and compared to commercial vehicle detection. Because these methods used to detect vehicles in the lanes, HSCM produced better result compared to commercial vehicle detection only in direct angle camera field of view. The developed system was used as a vehicle detector for real-time

Figure 2.1: Masked background

Figure 2.2: Masked current frame

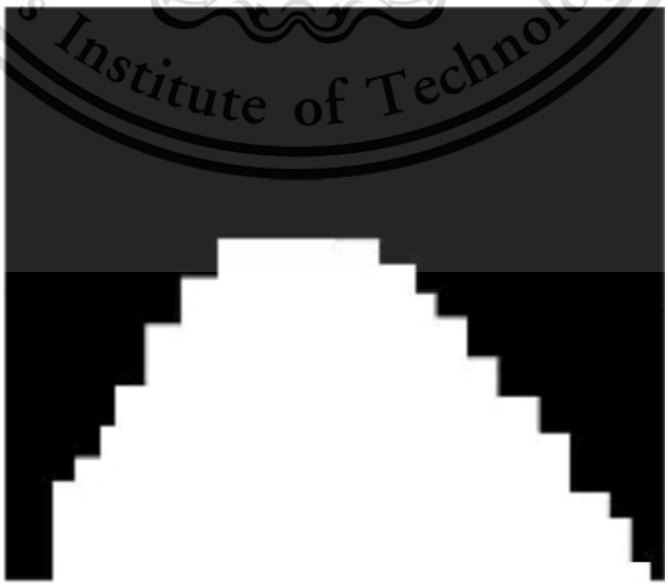


Figure 2.3: Binary mask

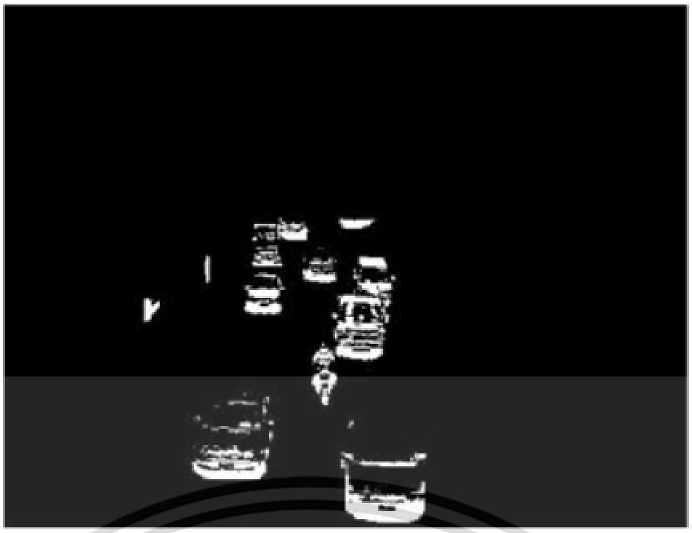


Figure 2.4: Group the connected pixels

advance warning ITS system.

Al-Garni and Abdennour [3] proposed background subtraction and edge detection techniques for vehicle detection. This technique uses the advantages of both approaches. The practical applications approved the effectiveness of this method. This method consists of two procedures: First, automatic background extraction procedure, in which the background is extracted automatically from the successive frames; Second vehicles detection procedure, which depend on edge detection and background subtraction. In order to apply automatic background extraction, there are 12 processing steps that they implemented to the first three successive frames. After they got background of the image, the extracted background must be subtracted from the current image with 6 processing steps to detect the vehicles:

1. Step 1. Subtract the extracted background from the current image.

2. Step 2. Find the edge of the current image and the background image.

3. Step 3. Subtract the edge of the background image from the edge of the current image.

4. Step 4. Fill the resulted images in steps 2 and 3. Implement logical And operation for the results in steps 2 and 3.

5. Step 5. Filter the resulted image.

6. Step 6. Count the resulted moving vehicles.

The result showed that the proposed methods can detect vehicles higher than 91% accuracy but it still had miss detection problem because of vehicles occlusion.

Han, Han and Hahn[4] presented Haar-like feature and the symmetric feature to detect vehicle. It can also remove the shadow of a vehicle on the road effectively. In addiction, they can detect the boundary of vehicles accurately. The vehicle detection algorithm can be divided into two main steps. One is hypothesis generation, and the other is hypothesis verification. In the first step, hypothesis generation determines vehicle candidates using features such as a shadow, intensity, and vertical edge. And in the second step, hypothesis verification determines whether the candidate is a vehicle or not by using the symmetry of vehicle edge features. The proposed algorithm was aimed to detect the vehicle on real time system. The execution time of the proposed algorithm was very short. So, it is proper to apply it to the real-time system. But it has a limit to detect vehicles using single camera. According to the experiment, the result still had error because of illumination changes of road environment. So they have to make the proposed algorithm more robust to work on embedded system.

Sharma, Kumar Katiyar, Kumar Gupta, and Singh [5] used differential morphology closing profile (DMP) to extract the vehicle automatically from the traffic image. They used a traffic image to test the proposed detection technique. Multi-scale DMP closing profile is applied on the traffic image for the automatic vehicle detection in the proposed algorithm. The following steps will be used to detect the vehicle in the proposed algorithm:

1. Input an image
2. Pre-processing
3. Differential Morphology profile
4. Thresholding
5. Filtering

After implemented this system, the experimental result showed that the DMP technique provided better result comparing to traditional image processing.

Khalid, Mazoul, and El Ansari [6] proposed a new vehicle detection method using so-called association combined with corner detector. The experiment results have validated the efficacy of our method, and they show that this method is capable to work in real time. In the future, they plan to improve their vehicle detection method, which will be tested to detect much more complex obstacles (pedestrian, traffic light, etc.) under different weather conditions.

Base on previously mentioned researches, vehicles occlusion affects the results of the vehicle detection. Some researches focus on vehicles occlusion detection and segmentation to solve this problem.

Chen, and Liu [7] presented a method to detect and segment occluded vehicles based on the effective skeleton nodes. The process of the detection and segmentation of occluded vehicles, first skeleton structure and some effective skeleton nodes are extracted from the blobs in foreground mask. Then blobs that are suspected of having more than one vehicle in them can be detected, and then find out the segmentation points in the blobs. Finally, suspected blobs are segmented into single vehicle based on the boundary of the blob associated with segmentation points. Figure 2.5 shows the result of vehilce occlusion segmentation. The experimental result shows that effective skeleton node method was effectively segment occluded vehicles but it still needed to be improved because skeleton features could be very sensitive to noise.

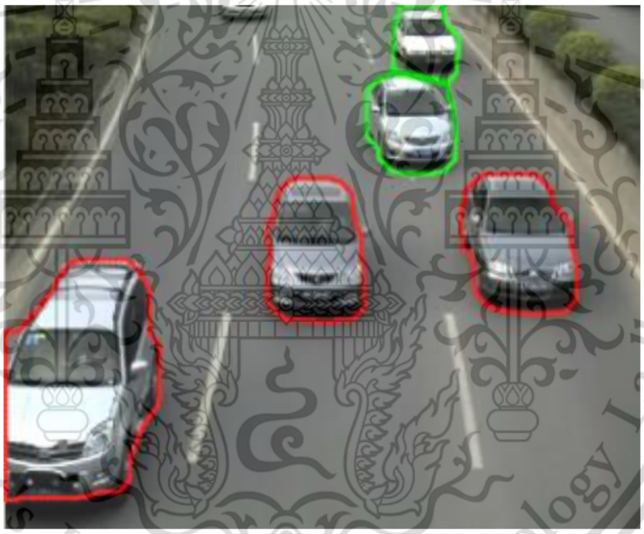


Figure 2.5: Vehicles occlusion segmentation result

Zhang, Zheng, and Liu [8] proposed morphology and the concavity property of the image to detect vehicles occlusion. From the experimental result, the algorithm showed the high performance in detecting occlusion in many kinds of scenarios.

2.2 Vehicle classification

Vehicle classification can give the information of number of each type of vehicles that passed the monitoring area. QiSen, and Mingjun [9] presents a novel method of vehicle classification using parameterized model and neural networks. First, they proposed the parameterized model, which can describe features of vehicle. In this model, vertices and their topological structure are regarded as the key features. Then they adopted classifier based on multi-layer perceptron networks (MLPN) to recognize vehicles. In this neural network classifier, learning algorithms based on the gradient descent method for the least exponential function error (LEFE) are adopted. Experimental results showed that parameterized model can satisfactorily and effectively describe vehicles. This system can classify vehicle by using neural networks classifier more than 91% accuracy. However, vehicle occultation and shadow effect to the result.

Avely, Wang, and Scott Rutherford [10] presented an image processing algorithm for length-based vehicle classification using an image stream captured. This method compare the length of vehicles to classify. In the experimental results, this system can classify almost 92% accuracy. Chen, TEllis and Smieee [11] used Support Vector Machine (SVM) as a classifier to classify vehicles into four categories i.e. car, van, bus, and motorcycle. This system can classify vehicle with 94.69% accuracy under varying illumination and weather conditions. The problem is the multiple vehicles are too close and it causes occlusion problem. Moreover, the similarity of vehicle's colour and background's colour also causes the classification problem. Messelodi, Modena and Zanin [12] presented vehicle classification based on a model-based matching technique to classify vehicle into seven categories. According to the experimental result, the system can classify vehicle with 87.3% accuracy. This problem happened because of shadows, reflections, occlusions, and pedestrian. They planned to solve the problem by separating a vehicle from its shadow, identifying and excluding pedestrian and uninteresting moving objects from the traffic statistics, and solve the occlusion problem. Ji, Jin and Li [13] proposed vehicle classification method using partial Gabor filter bank to classify five vehicles categorization: sedan, van, hatchback sedan, bus and van truck. The experimental results showed that the classification rate can reach to 95.17%. The problem happened because of the sampling method is still limited. Ozkurt and Camci [14] presented vehicle classification and traffic density calculation methods using neural networks. They classified vehicles into three categories i.e. car, van, and bus with 98% accuracy. The problem that affected the result is vehicle misclassification. After vehicles was identified, traffic density was calculated. In this study, they calculated traffic density of each type of vehicle by taking number of vehicle in each type divided by a period of time. Based on the experimental result, traffic density was calculated without error. However, the classification of vehicle types is performed with some error. Some of the vehicles have been misclassified.

2.3 Traffic density estimation

Traffic density estimation can help to predict the situation of the traffic that is going to happen. So it can prevent the traffic congestion problem. Al-Walid, Anjum, Tubba, Akter, Asfad

[15] counted the number of vehicles in the interested zone and set a threshold value to compare whether traffic density is high and low. Then the system can predict whether the road is block by the traffic or not. This system produced the results with unsatisfied accuracy. It needed to improve in the future work. Osuto, Absaloms and Nganga Ndungu [16] presented a simple and elegant approach for estimating the road traffic density during daytime using image processing and computer vision algorithms. Vehicles are detected and extracted from the images and counted. Then the traffic density is obtained as the number of vehicles per unit area of the road section. From this method the system produced the result with 96.0% and 82.1% accuracy of vehicle detection with fast moving and slow moving traffic scenes respectively. However, this algorithm also had problems with occluded vehicles. To minimize these problems, it is suggested to raise the position of the camera to be high enough with respect to the region of interest.

CHAPTER 3

RESEARCH METHODOLOGY

This chapter provides the detail information on the methods of this thesis. The main idea of the thesis is to develop a traffic monitoring system based on image processing technique. In order to develop this system, the information of traffic need to be identified. Vehicle classification and traffic density measurement are important informations used in for traffic monitoring system. In order to classify vehicle and calculate traffic density, vehicle detection need to be identified. So in this chapter presents firstly the vehicle detection in occluded scenarios. Then vehicle classification and traffic density measurement are secondly presented.

3.1 Vehicle detection in occluded scenarios

Vehicle detection is a method that is used to detect vehicle in the surveillance video. This method can be processed with the several step i.e. background subtraction by setting Region of Interest, morphological operation, and BLOB detection.

3.1.1 Background subtraction

Background subtraction is a technique in the fields of image processing and computer vision wherein an image's foreground is extracted for further processing. So background subtraction technique is introduced at this stage with the goal of foreground detection. To proceed with this technique, a reference image and current image are required. Both reference image and current image are shown in Figure 3.1 and Figure 3.2 are the important features for this technique. Reference image refers to the image consists only road or background without vehicles or moving objects. Current image refers to image that is extracted one by one iteratively from video. Because of the image consists not only desirable objects but also undesirable objects, region of interest (ROI) is set to focus on the specific zone. ROI refers to the a specific area of the road for monitoring. So ROI is set to both reference image and current image to do background subtraction. Figure 3.3 shows the reference image with ROI. After setting ROI, both reference image and current image are converted to grayscale images. This is because of background subtraction technique need grayscale image as the input.

Background subtraction technique takes current frame as shown in Figure 3.2 to subtract with reference image. The process of subtraction perform pixel by pixel at the same index as the following formula :

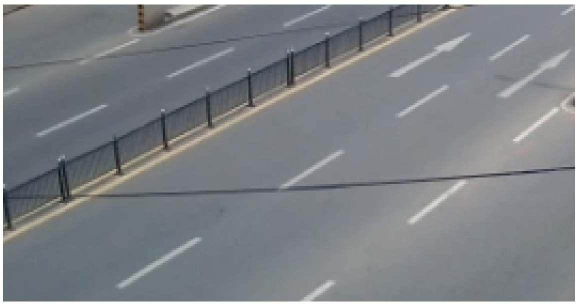


Figure 3.1: Reference image

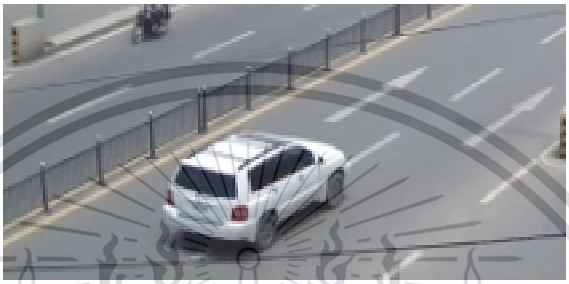


Figure 3.2: Current image

$$\text{diffImg}(i, j) = \text{currImg}(i, j) - \text{refImg}(i, j) \tag{3.1}$$

Where diffImg(i, j) is the intensity value at index (i, j) of the new image that obtains from the difference between current image and reference image at index (i, j). currImg(i, j) is the intensity value of current image at index (i, j) while refImg(i, j) is the intensity value of reference image at index (i, j).

The result of background subtraction is foreground or vehicle's object. The problem occurs with the binary image of the foreground because the object consist of noise and it is not completely connected. So morphological operation need to be implemented to solve this problem.

3.1.2 Morphological operation

Morphology is an image processing operation that processes images based on shapes. Morphological operation pursues the goals of removing noise and undesired texture in binary image. The most basic morphological operations are dilation and erosion. Dilation indicated by Equation 3.2, adds pixels to the boundaries of objects in an image while erosion indicated by Equation 3.3, removes pixels on object boundaries.

$$g = f \oplus s \tag{3.2}$$

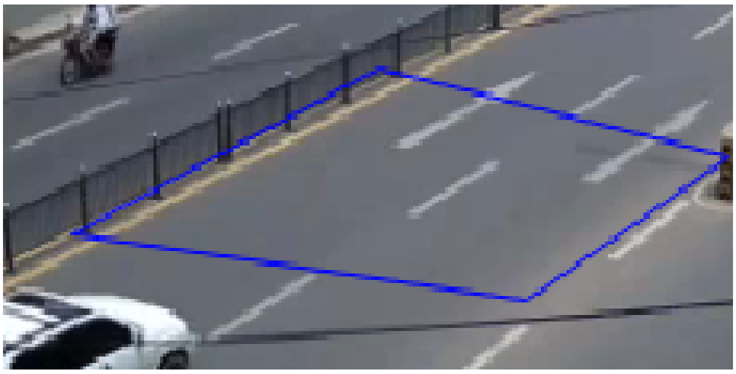


Figure 3.3: Region of interest of the image

$$g = f \ominus s \tag{3.3}$$

where g is the output image, f is the input image ,and s is the structure element. $\oplus$ and $\ominus$ represent dilation and erosion operation, respectively.

Morphological techniques processes an image with a small shape or template called a structuring element. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighbourhood of pixels. Erosion tests whether the structure element "fit" within the neighbourhood, while dilation tests whether it "hit" or intersects the neighbourhood. The technique to test "fit" or "hit" is shown in Figure 3.4.

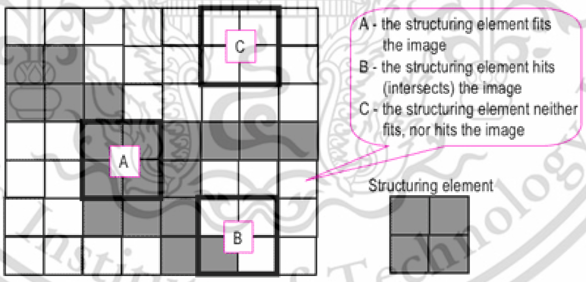


Figure 3.4: Fits or hits checking

The results of dilation and erosion operation look like the objects in Figure 3.5 and Figure 3.6, respectively. Where the objects at the left of arrow are inputs and at the right are output.

After background subtraction the noises are consist and object is not enclosed. So morphological operation is implemented to solve this issue. It is used to remove noise and add pixels to enclosed holes of the objects. Erosion operation is applied to remove the noise of the object and dilation operation is used to add missing pixels on the object to enclosed the vehicle's object. After these operations are applied the object becomes completely enclosed and noises are removed. Morphological operation makes unconnected pixel of vehicle object to be completely enclosed

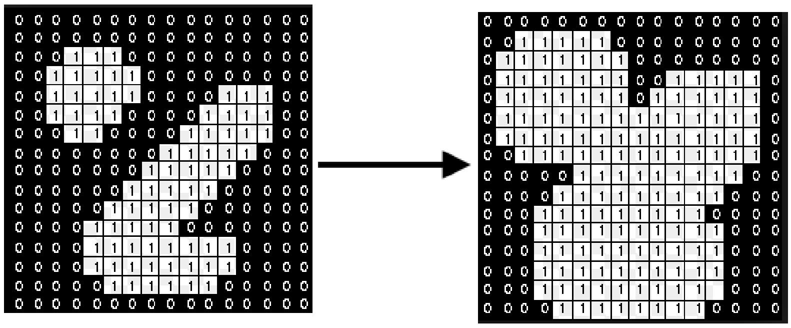


Figure 3.5: Result of dilation

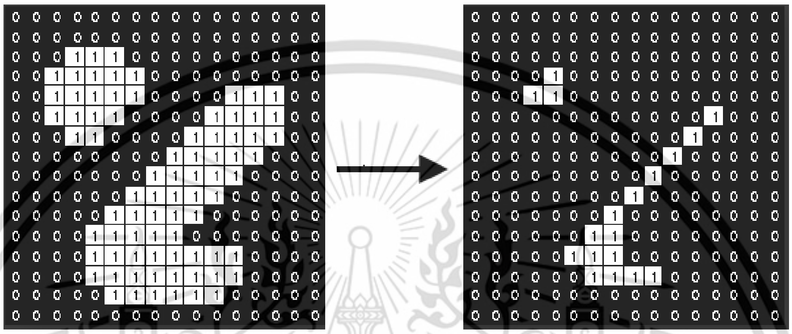


Figure 3.6: Result of erosion

and removes noises in order to make object well detected. Then vehicle's object can be detected by implementing BLOB detection.

3.1.3 BLOB detection

After the morphological operation, the white areas which are assumed to be the vehicles will be completely connected and can be seen as an object. The BLOB detection is then applied to detect the vehicle in every frame of the input video. BLOB stands for Binary Large OBject and refers to a group of connected pixels in a binary image. The purpose of BLOB detection is to isolate the BLOBs (objects) in a binary image. The image's pixels are then labeled to extract or detect the BLOB by performing the labelling technique. In order to label image's pixels, labelling technique marks the image pixels with value 1, which are connected, with the same number.

Structure element is an important component for labelling technique. It can be 8-connectivity or 4-connectivity used to process the labelling of the object in the image. The 8-connectivity is more accurate than the 4-connectivity, but the 4-connectivity is often applied since it requires less computational resources. The chosen connectivity, however, can affect detection results. Figure 3.7 shows that the result image contains either one or two BLOBs depending on the chosen connectivity.

From the advantage and disadvantage of choosing connectivity, 4-connectivity is chosen. So that structure element with 4-connectivity is implemented for labelling technique. The object can be labelled by applying Recursive Grass-Fire Algorithm. The algorithm starts in the upper-left corner of the binary image. It then scans the image from left to right and top to bottom by using structure element with 4-connectivity. Figure 3.8 shows the process of labelling the object in the image while Figure 3.9 illustrates the result of the labelling process. As a result, the objects are labelled into two differences labels. This means there are two different objects in this image.

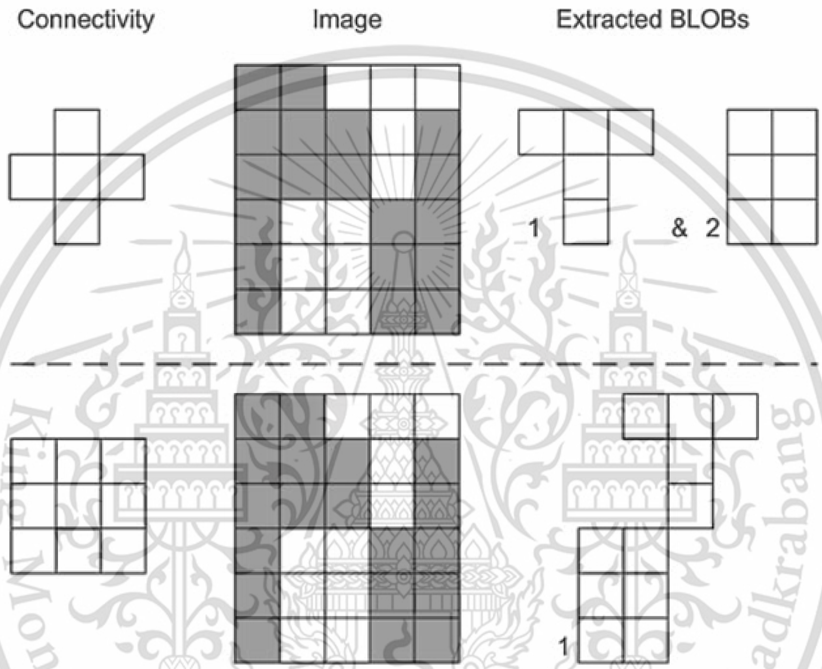


Figure 3.7: Different number of objects caused by different chosen connectivity

Based on BLOB extraction technique with Recursive Grass-Fire Algorithm for labelling process, the system is able to extract objects (vehicles in this case) from every frame of video.

After applying BLOB detection, vehicle is detected in binary image, some information are obtained; e.g. area of pixels are the number of white pixels and bounding box of the vehicle that composes of its location of upper-left corner, width, and height. However, vehicles are detected in occlusion scenarios. This occlusion scenarios affects to the result of vehicle detection. In order to solve this problem, skeleton feature extraction is needed to segment the occluded vehicle's object.

3.1.4 Occluded scenarios handling

Vehicle occlusion is one of the issues that affects to the result of vehicle detection. It may give the wrong object detection and give wrong information. From the vehicle detection in the previous section, vehicles can be detected in occluded scenarios. Figure 3.10 shows two vehicles

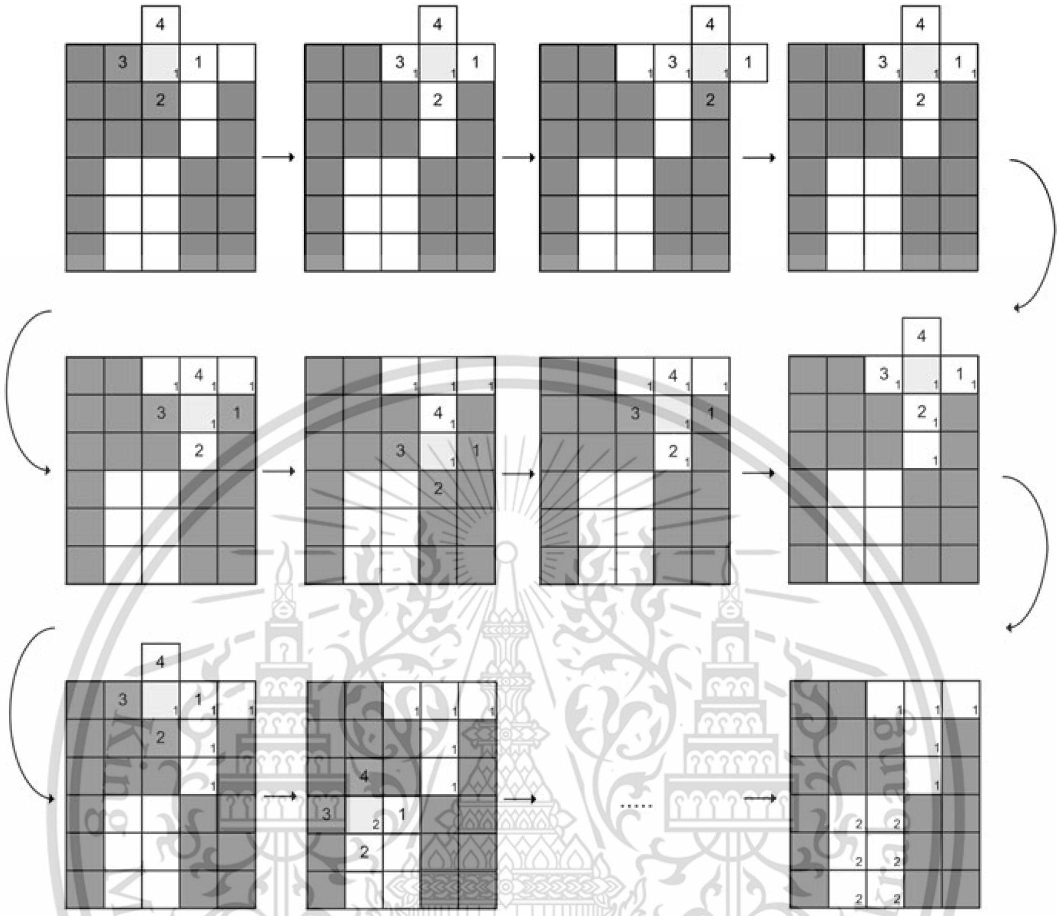


Figure 3.8: Process of BLOB labeling

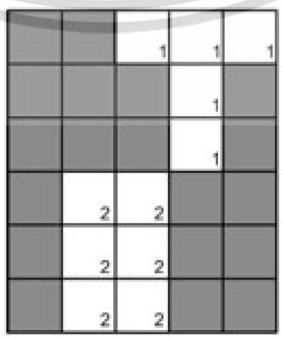


Figure 3.9: Result of the objects after labelling process

occluded with each other that make system give the wrong object detection. Hence, this part is focus on the technique to detect and segment the occluded vehicles.

Figure 3.10: Occluded vehicles

Skeleton feature extraction is one of the methods that can handle vehicles detection in occluded scenarios. This method generates the skeleton structure of the object and analyse the skeleton structure whether the original object is only one object or occluded objects. The process of skeleton feature extraction for vehicles occlusion is described in three following steps. They are skeleton structure extraction, segmented point extraction, and line segment extraction.

3.1.5 Skeleton structure extraction

Skeleton structure extraction is the way or the technique that skeleton structure of the object is extracted. The aim of the skeleton structure extraction is to extract a shape feature that represents an object. One approach to extract skeleton feature is thinning technique. Thinning is the process of peeling off a pattern as many pixels as possible without affecting the general shape of the pattern. In other words, after pixels have been peeled off, the pattern can still be recognized. Hence, the skeleton obtained must have the following properties:

- Must be as thin as possible
- Connected
- Centered

In order to apply thinning technique, Hilditch Thinning algorithm is used. The process of Hilditch thinning algorithm, the choice of each pixel in an image only has two possibilities: removed or not removed. In order to test whether the pixel is removed or not, it needs 4 conditions to verify. In this case, background region is assumed that it has value 0 (white) and the object regions have value 1 (black).

For example, Figure 3.11 shows a pixel $P1$ with its 8-connectivity or 8 neighbors pixels. Moreover, its 8 neighbors are arranged in a clock-wise order with two functions defined.

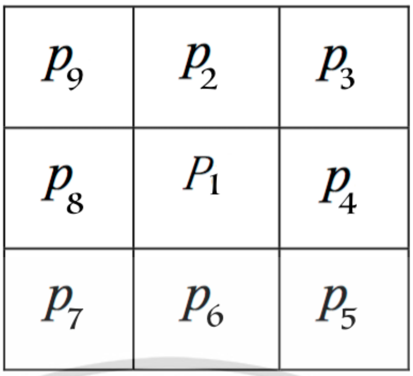


Figure 3.11: 8-connectivity's pixel

First function: $B(p1)$ = number of non-zero neighbors of $p1$

Second function: $A(p1)$ = number of pixel change from 0 to 1 patterns in the sequence $p2, p3, p4, p5, p6, p7, p8, p9, p2$

Example: From Figure 3.12, those two function about can be identified as: $B(p1) = 2, A(p1) = 2$

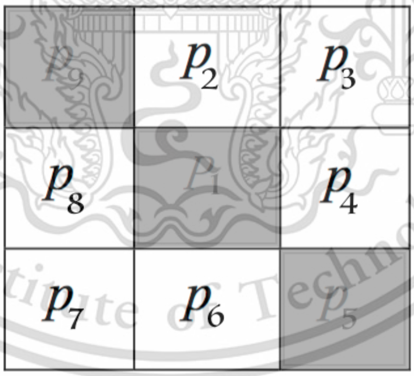


Figure 3.12: 8-connectivity's pixel

So $P1$ needs to be decided whether peel it off or keep it as part of the resulting skeleton. It is depend on the following 4 conditions:

Condition 1: $2 \leq B(p1) \leq 6$ This condition means that the number of non-zero neighbors of p1 is greater than or equal to 2 and it be less than or equal to 6. It ensures that no end-point pixel and no isolated one be deleted (any pixel with 1 black neighbor is an end-point pixel), and it ensures that the pixel is a boundary pixel.

Condition 2: $A(p1) = 1$ This is a connectivity test. If $A(p1) > 1$, the pattern will become disconnected when $p1$ is remove.

Condition 3: $p2, p4, p8 = 0 or A(p2)! = 1$ This condition ensures that 2-pixel wide vertical lines do not get completely eroded by the algorithm.

Condition 4: $p2, p4, p6 = 0 or A(p4)! = 1$ This condition ensures that 2-pixel wide horizontal lines do not get completely eroded by the algorithm.

This condition needs to be checked for each pixel of the image and it stop when nothing changes (no more pixels can be removed). So that, skeleton structure is extracted after this algorithm is implemented.

For example, Figure 3.13 shows the skeleton structure extracted from the object of letter T. Figure 3.13(a) is the original object of letter T while Figure 3.13(b) is the skeleton structure generated from the original object letter T. After skeleton structure is extracted, skeleton node is one of the important features of skeleton structure is needed to detect and segment occluded object. Skeleton node or can call junction point, it refers to the pixel that has three or more neighbors pixels (see Figure 3.14). The circle point in Figure 3.14 represents the skeleton node of the object of letter T.

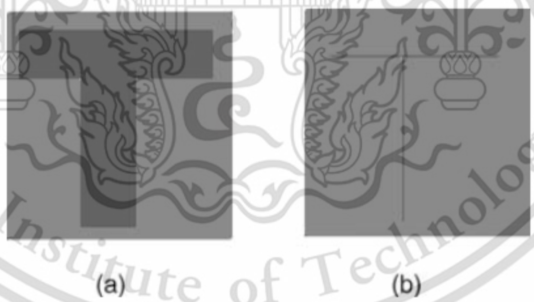


Figure 3.13: (a) Original pattern; (b) skeleton as a result of thinning

This study used skeleton structure and skeleton node to detect and separate the occluded vehicles. The main function of skeleton node is used to represent the number of objects or vehicles. Without occlusion problem, the detected vehicle has the skeleton structure with two nodes only. However, it can give skeleton structure with more than two nodes in the real situation as shown in Figure 3.15. All (*) signs are the skeleton nodes of the skeleton structure.

To solve this problem, Alternating Sequential Filter (ASF) is used. AFS is a composition of opening and closing operations. Opening is a morphological operation defined as an erosion

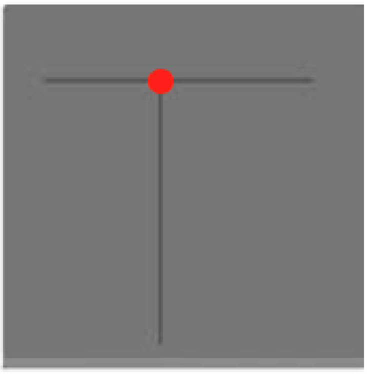
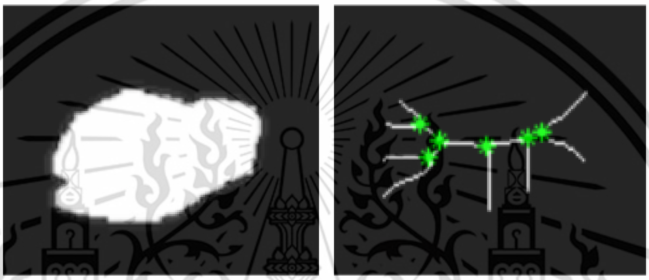


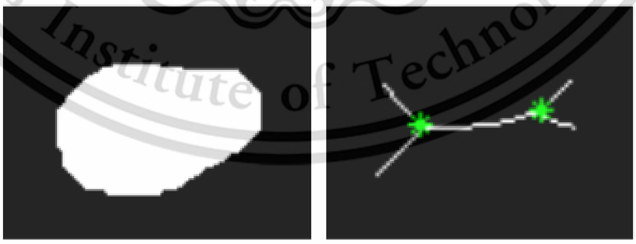
Figure 3.14: Skeleton node



(a) (b)

Figure 3.15: (a) Original object; (b) Skeleton structure

followed by a dilation with the same structuring element. Closing is a morphological operation that performed the reverse of opening. The aim of this method is to make the boundary of object become more smooth. After smoothing the object boundary, system can generate the skeleton structure of object without occlusion occur with only two nodes as shown in Figure 3.16.



(a) (b)

Figure 3.16: (a) Object after smoothing boundary; (b) Skeleton structure

In case of two vehicles occlusion after applying ASF, it gives skeleton structure with four nodes. However, it can give skeleton structure with only three nodes depending on the shape of vehicle object. Chen and Liu [7] the number of occluded vehicles can calculate by following formula:

$$N = \frac{\text{Number of nodes}}{2} \tag{3.4}$$

Where N is the number of vehicles occluded. In case the number of nodes is the odd number greater than 2, the result of N is the floating point number. Because of the number of vehicles cannot be a floating point number, the system rounds it to the next nearest integer number. From the number of skeleton nodes, the number of occluded objects or vehicles now can be calculated. In order to segment occluded vehicles, segmented point needs to be identified in the next step.

3.1.6 Segmented point extraction

After number of occluded vehicles are defined, segmented point is can be determined. A segmented point is the point that is used to segment occluded vehicles. In order to select the segmented point, it depends on number of nodes. This proposed system discusses two different cases. They are three nodes extracted and four node extracted in two vehicles occlusion problem. In case of three nodes extracted, system will find the middle node which has x value greater than the first node and less than the second node. Then, it calculates distance from the first to middle node and from the middle to the last node by using Euclidean Distance formula:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{3.5}$$

Where D is the distance between two nodes, (x_1, y_1) is the coordinate of the first node, and (x_2, y_2) is the coordinate of the second node. The segmented point is found at the middle of the two nodes with the longest distance as in the following formula.

$$\text{seg}(x, y) = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}\right) \tag{3.6}$$

Where $\text{seg}(x, y)$ is the coordinate of segmented point, (x_1, y_1) is the coordinate of the first node, and (x_2, y_2) is the coordinate of the second node. In case of four nodes extracted and counted from left to right (1 to 4), the segmented point is found at the middle of the second and the third node by using Equation 3.6. As shown in Figure 3.17, the second node is the node that has x value greater then the first node and less than the third node. The third node is the node that has x value greater than the second node and less than the fourth node. The segmented point is shown as the (+) sign (see Figure 3.17). Segmented point is generated, so line segment is then extracted in the next step.

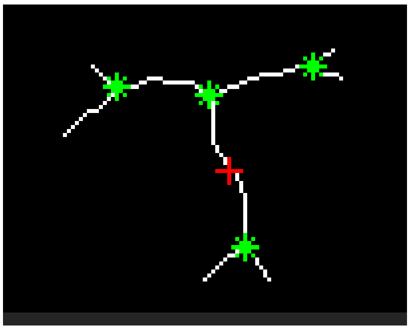


Figure 3.17: Segmented point

3.1.7 Line segment extraction

Line segment refers to the line that is used to separate occluded vehicles. It can be found after segmented point has been identified. To obtain the line segment, there are two approaches involved. The first approach used every boundary pixels to identify line segment [7] while the second approach is downsampling boundary pixels to determine line segment.

The first approach, all boundary pixels of the object are divided into two parts, upper part UP $= \{(x_1^u, y_1^u), ..., (x_n^u, y_n^u)\}$ and lower part LP $= \{(x_1^l, y_1^l), ..., (x_n^l, y_n^l)\}$. The set of UP points are the points that have y value less than y value of segmented point while the set of LP points are the points that have y value greater than y value of segmented point.

As shown in Figure 3.18, (o) sign is the segmented point while $P1$ and $P2$ are the points on boundary pixels which have y value the same with the segmented point. So that the set of UP points are the boundary pixels above $P1$ and $P2$ while the set of LP points are the boundary pixels below $P1$ and $P2$. The Euclidean distance is used to calculate distance from the segmented point to UP denoted by DU $= \{d_1^u, d_2^u, ..., d_n^u\}$ and from the segmented point to LP denoted by DL $= \{d_1^l, d_2^l, ..., d_n^l\}$. The solid lines are shown in Figure 3.18 present the distance from segmented point to the set of UP points and the set of LP points. Finally, line segment is found by connecting between two points. The first point is the one from the set of points in UP that has minimum distance from segmented point, $min(DU)$, and the second point is the one from the set of LP that has minimum distance from segmented point $min(DL)$. As the result, segment line is found as the dash-line (see Figure 3.18).

The second approach is the modified version of the first approach. It starts with segmented point (o) and four points on the boundary with sign (Δ) as shown in Figure 3.19. $P1$ and $P2$ are the boundary points that have the same y value with the segmented point while $P3$ and $P4$ are

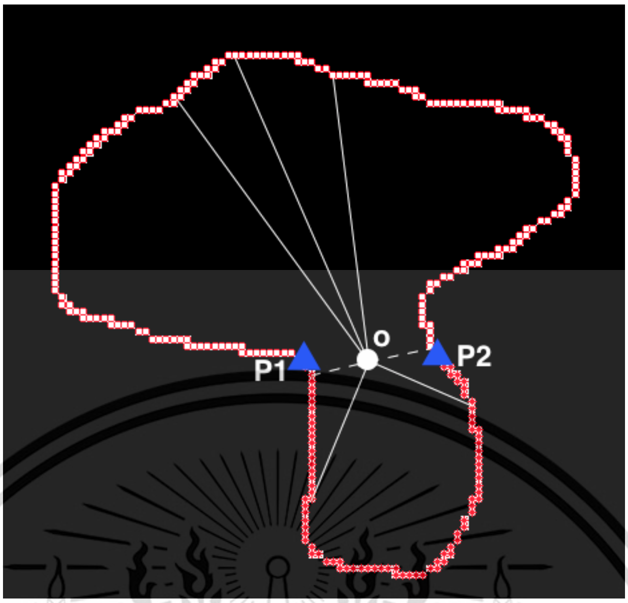


Figure 3.18: Line segment extraction first approach

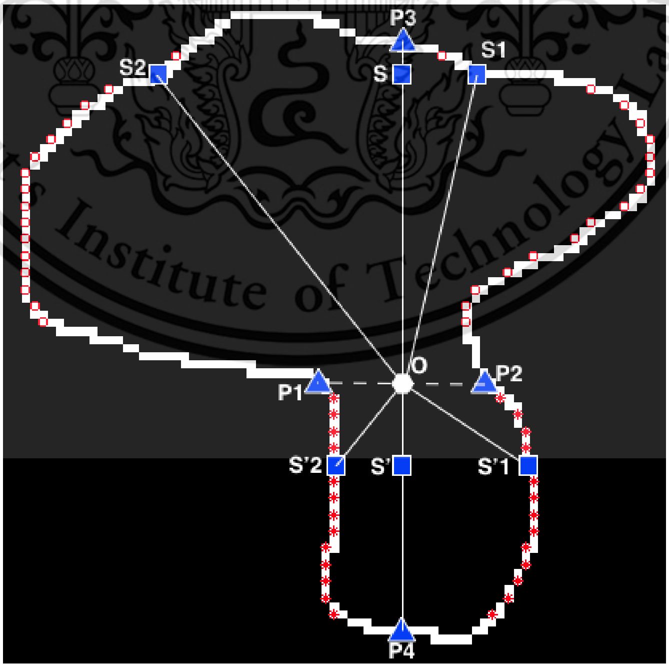


Figure 3.19: Line segment extraction second approach

the boundary points that have the same x value with the segmented point. S and S' are the points on the line $(P3P4)$. S, which is the point can move from $P3$ to segmented point, is used to do the downsampling for the upper part of the boundary pixels. S', which is the point can move from $P4$ to segmented point, is used to do downsampling for the lower part of the boundary pixels. $S1$ and $S2$ are the downsampling points where its y value equals to y value of S while $S'1$ and $S'2$ are the downsampling points where its y value equals to y value of S'.

The process of downsampling starts when S and S' is moving. The step of moving is defined by 2 pixels in this research. While S and S' are moving in each step, four of downsampling points are identified. The distance from the segmented point to those downsampling pixels is calculated by using Equation 3.5. After downsampling, the system finds one point from upper part of the boundary pixels and one point from lower part of the boundary pixels that have minimum distance from the segmented point. Finally, line segment is found by connecting those two points. The line segment is the dash-line (see Figure 3.19). By comparing these two approaches in term of computational resources the second approach is better. This is because it can reduce the computational time with the comparable result to the first approach.

By applying skeleton feature extraction, the problem can be solved. The occluded vehicles can then separated form each other.

3.2 Vehicle classification

Vehicle classification is a subsystem used to categorize vehicles into different classes [17]. The goal of vehicle classification is to know the amount of vehicles in each class which pass through the monitoring area. According to the proposed system, vehicles are classified into three different categories i.e. car, motorcycle, and tricycle (Tuk-Tuk). In order to classify vehicles, neural network back-propagation and radial basis function network are implemented. The results and performance of these two types of neural network are compared. The structure of back-propagation neural network is shown in Figure 3.20 with three layers i.e. input, hidden, and output layer. $\{x_1, ..., x_n\}$ are the inputs of the network. Hidden layer consists neurons and each neuron consists of summation function and activation function. Summation function will sum the multiplication of each input x_i with its weight w_i while activation function $f(R)$ of hidden layer is chosen to be sigmoid function presented as Equation 3.7. As a result, the output of each neuron in hidden layer is illustrated as Equation 3.8.

$$f(R) = \frac{1}{1 + e^{-R}} \tag{3.7}$$

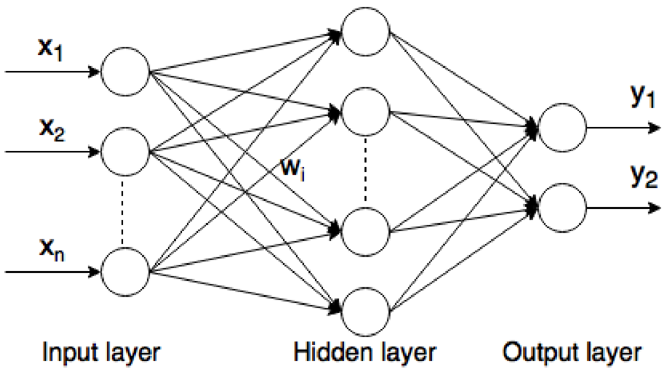


Figure 3.20: Implemented back-propagation structure

where R is the output of the summation function computed by Equation 3.8.

$$out = f(\sum_{i=1}^{n} x_i w_i) \tag{3.8}$$

where x is input and w is weight of input x.

The last layer of neural network is output layer. This layer in each neuron also consists of summation function and activation function similar to the hidden layer. The output of each neuron is also defined as Equation 3.8.

Radial Basis Function Network(RBFN) is an artificial neural network that uses radial basis functions as activation functions. $\{x_1, ..., x_m\}$ are the inputs of the network in the structure of RBFN (see Figure 3.21). RBFN has only one hidden layer with activation functions φ_i. Gaussian function will be used as the activation function of the neuron in the hidden layer. The function can be stated as shows in Equation 3.9.

$$\varphi = \exp(-\frac{r^2}{2\sigma^2}) \tag{3.9}$$

where $r = \|x_i - t_i\|$, x_i is the input and t_i is the center of the neuron in hidden layer. Moreover, t_i is generated randomly from the input and $\sigma = \frac{d}{\sqrt{n}}$, which d is the maximum distance between any two centers t and n is the number of centers.

The output y is the target class and it will be used to define the weight (see Equation 3.10).

$$w_1\varphi_1(\|x_1 - t_1\|) + ... + w_m\varphi_m(\|x_m - t_m\|) = y \tag{3.10}$$

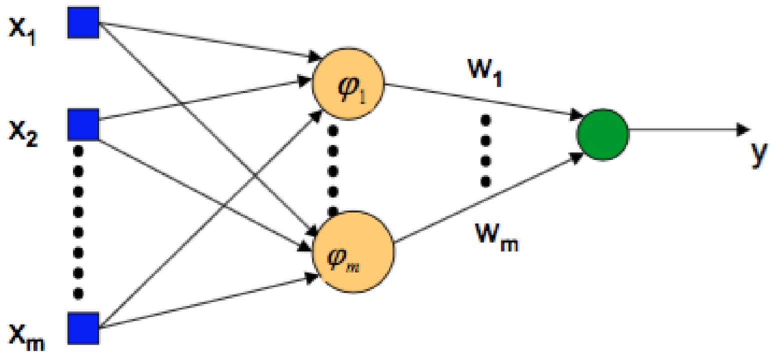


Figure 3.21: Implemented radial basis function network structure

These neural networks need to be trained with the dataset before it can classify vehicles into three different categories i.e. car, motorcycle, and tricycle(Tuk-Tuk). From the dataset, there are two different inputs x_1 and x_2 to the neural network system. x_1 represents the ratio between width and heigh of the vehicle while x_2 represents the number of pixels in the area of vehicle. These two feature inputs can be generated after vehicle detection is implemented. Vehicle detection using BLOB detection can detect vehicles and gives the value of its width, height and the number of pixels of its area. So the two feature inputs to the neural network is identified.

There are two videos used to train and test the neural networks for vehicle classification. The dataset from the first video is used to train the network while data set from the second video is used to test the network. The problem of vehicle classification occurs with this approach. This is because the distance from camera to object in those two videos are different. Such different distance affects the size of detected vehicles in the two videos. Affine transformation, which is a function that maps an object from an affine space to another space with structure preserving, is then applied to solve this problem. It consists of various operations such as translation, rotation, and scaling. In this research, scaling operation is implemented to scale the size of object of the testing video to be similar with the training video. In order to apply the scaling, nearest neighbor interpolation is the scaling methods implemented in this system. Nearest neighbor is the simplest and fastest implementation of image scaling technique. This scaling technique requires the size of the original image and the size of the output image to be scaled. In order to apply this technique two assumptions are made:

1. The original image, $I_1(v, w)$, is defined in vw coordinate system with width and height of w_1 and h_1, respectively.

2. In the similar manner, an output image, $I_2(x, y)$ is defined in xy coordinate system with

width and height of w_2 and h_2, respectively.

In this system the size of the output image is assigned to 50% smaller than the original image. Firstly, the ratio between the width of the original and output image and the ratio between the heigh of the original image and the output image are defined by Equation 3.11 and Equation 3.12.

$$ratio_w = \frac{w_1}{w_2} \tag{3.11}$$

$$ratio_h = \frac{h_1}{h_2} \tag{3.12}$$

Then the value of pixel in the output image is calculated by Equation 3.13. This method needs to swap x and y as well as v' and w' to find the value of the pixel of the output image.

$$I_2(y, x) = I_1(w', v') \tag{3.13}$$

where w' and v' are defined by Equation 3.14 and Equation 3.15, respectively.

$$w' = \frac{y}{ratio_h} \tag{3.14}$$

$$v' = \frac{x}{ratio_w} \tag{3.15}$$

After nearest neighbor interpolation is calculated, the object in the second video is scaled to 50% smaller that make its size similar to the first video. The original vehicle's object in the second video before Affine transformation is applied. This vehicle's object is large compare to the size of vehicle's objects in training video. In order to avoid the problem, affine transformation is implemented to scale down those vehicle's objects 50% . So affine transformation using nearest neighbor interpolation is implemented to scale the vehicle's objects in testing video to make its size similar to training video.

3.3 Traffic density calculation

Traffic density is implemented to illustrates the statue of the area of the road. Then the monitoring system will use this status to analyse whether that area cause traffic congestion or not. There are many techniques that can use to calculate traffic density. However, in this research proposed a technique to calculate traffic density of the area of the road. This technique calculates traffic density from each frame of the video. The system calculates traffic density by doing the summation of the area of each vehicle inside the region of interest (ROI) Figure 3.22 divided by the area of ROI. Traffic density is computed by Equation 3.16.

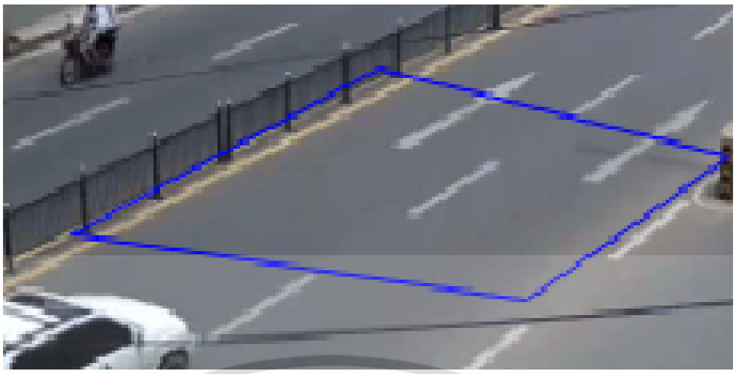


Figure 3.22: Region of interest

$$Density = \frac{\sum_{i=1}^{n} V_i}{\text{Area of ROI}} * 100 \tag{3.16}$$

Where V_i is the number of pixels in the area of vehicle i^{th} and n is the number of vehicle inside ROI.

V_i can be extracted by using BLOB detection in vehicle detection technique. After applying BLOB detection, the number of pixels is given as presented in the research study of Nuthong and Sotheany [18].

ROI is the region that set to monitor with polygon shape. So the area of ROI is calculated by using Equation 3.17

$$Area = \frac{|(x_1 y_2 - y_1 x_2) + (x_2 y_3 - y_2 x_3) + ... + (x_n y_1 - y_1 x_n)|}{2} \tag{3.17}$$

where x and y are the coordinates of vertex of the ROI.

CHAPTER 4

DISCUSSION AND EXPERIMENTAL RESULTS

This chapter presents the discussion and experimental results of the research study. It presents the vehicle detection, vehicle classification and traffic density measurement.

4.1 Experimental setup

This system runs on iMac 21.5-inch contains 2.7 GHz Intel Core i5 processor.

There are three traffic videos used in this research. The information of each video is shown in Table 4.1.

Table 4.1: Information of input videos

Videos	Number of Frames	Resolution	Frame rate (fps)
1	4301	598 x 396	60
2	395	1278 x 720	10
3	6649	456 x 288	60

4.2 Discussion

This section describes the results of the experiment on three part i.e. vehicle detection, vehicle classification, and traffic density measurement.

4.2.1 Vehicle detection

Vehicle detection uses two different videos input as shown in Table 4.1 to do the experiment. From the input video, current image and reference image are generated as the feature of background subtraction. The result of background subtraction is converted into binary image to match with the requirement of BLOB detection that need binary image as the input. Because these input videos have vehicles occlusion problem, skeleton feature extraction is performed to handle occlusion issue.

Table 4.2: Result without handling occlusion

Videos	Actual number of vehicles	Detected Vehicles	Rate %
1	48	44	91.67
2	15	14	93.33

Table 4.2 presents the experiment of vehicles detection in both videos without occlusion handling. It results in unsatisfied accuracy of vehicle detection. In contrast, Table 4.3 presents

Table 4.3: Result with handling occlusion

Videos	Number of vehicles	Occlusions	Detected Vehicles	Rate %
1	48	4	47	97.91
2	15	1	14	93.33

the results obtain from both videos with vehicles occlusion handling. It is found that detection accuracy increases for first video. The second video shows no difference. The reason is that the second video has only one occlusion and the system cannot detect it. However, this system can detect vehicles in the various videos input with the satisfied result.

Table 4.4: Compare execution of segment line technique

Segment line technique	Execution time(s)
1	0.357461
2	0.020347

Table 4.4 presents the execution time of proposed approaches to find line segment. It is applied only on the image in Figure 3.18 and Figure 3.19. It shows that the second approach uses less execution time compare to the first approach.

4.2.2 Vehicle classification

Vehicle classification do the experiment on two different videos from Table 4.1. There are two approaches that is used to select the dataset. First approach, 80% of dataset in video 1 for training and 20% for testing both Back-propagation neural network and Radial basis function network. It results almost 100% classify correctly for this approach. For second approach, dataset in one of these two videos is used for training while dataset in another video is used for testing both Back-propagation neural network and Radial basis function network.

Table 4.5: Comparison between back-propagation and RBFN. Note that AF refers to affine transformation.

Classes	Training	Testing	Classification			
			Back-propagation		RBFN	
			Before AF	After AF	Before AF	After AF
Car	20	1	0	1	0	0
Motorcycle	44	5	5	4	5	5
Tuk-Tuk	5	1	0	1	0	0
Total	69 (100%)	7 (100%)	5 (71%)	6 (86%)	5 (71%)	5 (71%)

Table 4.5 shows that the number of vehicles taken from the first video for training the net-

work is 69 which is categorised to 20 cars, 44 motorcycles, and 5 tricycles (Tuk-Tuk) while 7 dataset taken from the second video for testing consists of 1 car, 5 motorcycles, and 1 tricycles (Tuk-Tuk). The classification results of both Back-propagation and Radial basis function network are the same before applying affine transformation. They can classify correctly motorcycle but incorrectly car and tricycles. Table 4.5 shows that both Back-propagation and Radial basis function network can classify vehicle up to 71% accuracy. The misclassification occurs because the size of vehicles in testing video and training video are different. This problem is solved by using affine transformation technique to scale the size of vehicles of testing video to be similar with training video.

After affine transformation is applied, the results in Table 4.5 shows that the accuracy of classification of Back-propagation technique is increased to 86% while Radial basis function network stay the same accuracy.

From the previous experiment of classification, small dataset is taken from the second video is used for testing. So the result of classification still difficult to discuss. So new dataset is generated from the third video used for testing the classification in order to ensure the performance between both neural network algorithm i.e, Back-propagation and Radial basis function network. We found that the size of object in this new dataset is far different from the training set. From the previous experiment, applying affine transformation would give better result for classification. We then apply affine transformation for this new data before applying classification techniques. From Table 4.6 shows that amount 75 testing data that consist of 19 cars, 37 motorcycles and 19 tricycles, Back-propagation can classify 59 of vehicles correctly that are 19 of cars, 28 of motorcycles and 12 of tricycles. But for Radial basis function network can classify 55 of vehicles correctly that are 9 of cars, 32 of motorcycles and 14 of tricycles. So the accuracy of Back-propagation neural network is 78% classify vehicles correctly with Radial basis function network can classify 73% correctly.

From these experiment above, it shows that both Back-propagation and Radial basis function network can classify vehicle with the similar accuracy. So the performance of both neural network algorithm are comparable.

In case of computational, it is found that Back-propagation uses slightly less execution time compared to Radial basis function network. This is illustrated in Table 4.7. The classification results is illustrated in Table 4.5 and Table 4.6. The number of vehicles in each class provides valuable information for traffic monitoring system.

Table 4.6: Comparison between Back-propagation and Radial basis function network(RBFN)

Classes	Training	Testing	Classification	
			Back-propagation	RBFN
Car	20	19	19	9
Motorcycle	44	37	28	32
Tuk-Tuk	5	19	12	14
Total	69	75	59 (78%)	55 (73%)

Table 4.7: Training execution time comparision

Neural network types	Execution time(s)
Back-propagation	1.218395
Radial basis function	1.664398

4.2.3 Traffic density measurement

The experiment of traffic density is the calculation of traffic density inside region of interest (ROI) to measure how is the density of that zone. The result of traffic density is shown in Figure 4.7. A vehicle is passing through ROI with 38% of traffic density. After traffic density is calculated, monitoring system can use that information to predict whether the monitoring zone have congestion or not. If density value keeps increasing, it means that the monitoring area is in congestion or it is going to have congestion problem soon. Otherwise, if density value is decreasing, it means that the traffic is going to be better. So the information of traffic density information can make traffic monitoring system defines whether the monitoring area is in the bad or good traffic.

4.3 Results

In this part, the results produced from this system is presented. It is divided into vehicle detection, vehicle classification, traffic density measurement. Firstly, traffic video is used to initial the system. Then the vehicle detection part of the system uses background subtraction, morphological operation and BLOB detection to detect vehicle inside the traffic video. After background subtraction is implemented the foreground of the image is obtained (see in Figure 4.1) by removing the background.

The foreground of the image consists of unconnected object's pixel. So, morphological operation is used to fill some missing information or pixels by using dilation and erosion operation. After morphological operation the result as shown in Figure 4.2 enclosed the pixels of the vehicle's object. BLOB detection technique is used after morphological operation in order to detect or extract the information of vehicle's object. After, BLOB detection vehicle are detected as shown

Figure 4.1: Foreground object

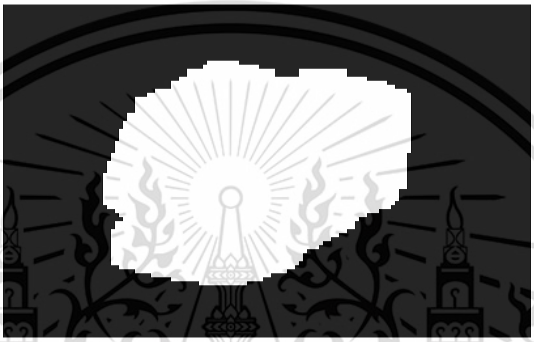


Figure 4.2: Result of morphological operation

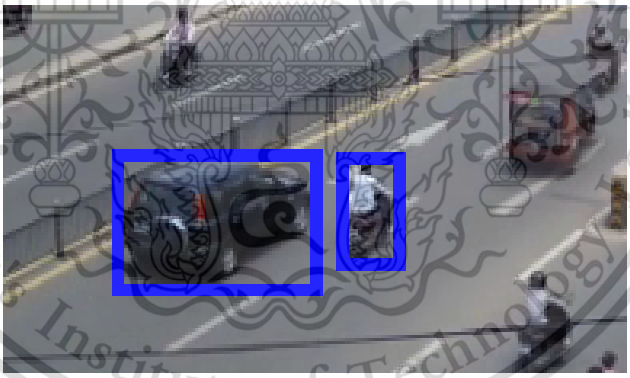


Figure 4.3: Vehicle detection result

in Figure 4.3. However, vehicle occlusion issue is occurred after vehicles are detected. Skeleton feature extraction is applied to detect and segment the occluded vehicles. After this technique is implemented, occluded vehicles are detected and separated as shown in Figure 4.4.

After vehicle occlusion problem can be solved, the information extracted from vehicles are used for vehicle classification. There are two different neural network type used to classify vehicle into classes i.e, car, motorcycle and tricycle. The results of vehicle classification are shown in Figure 4.5 and Figure 4.6.

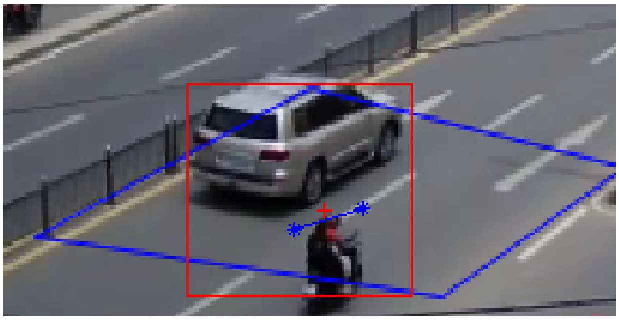


Figure 4.4: Vehicle occlusion handling

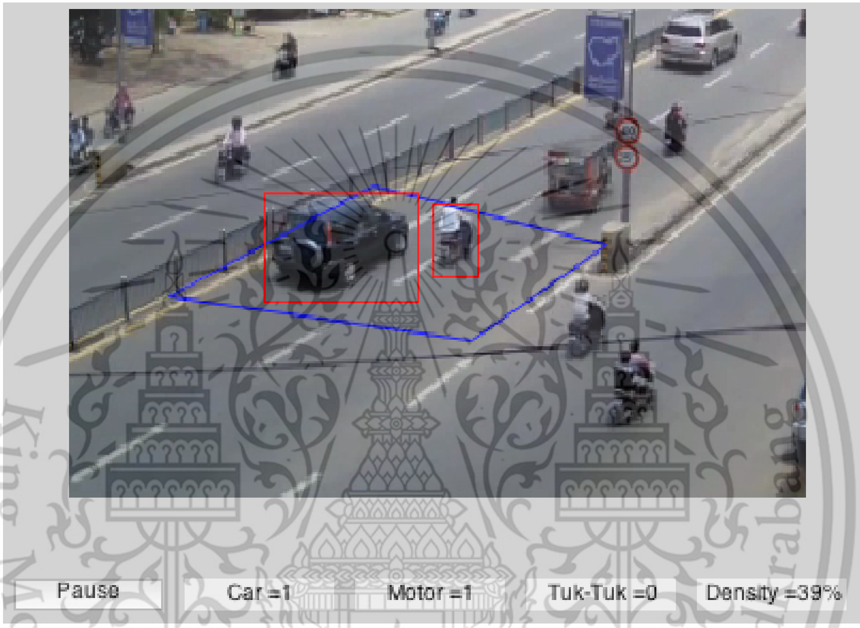


Figure 4.5: Car and motorcycle classification

Moreover, the last step of this system is traffic density measurement. This is used to calculate the density of the traffic in specific zone of the road. The result of traffic density calculation can see in Figure 4.7.

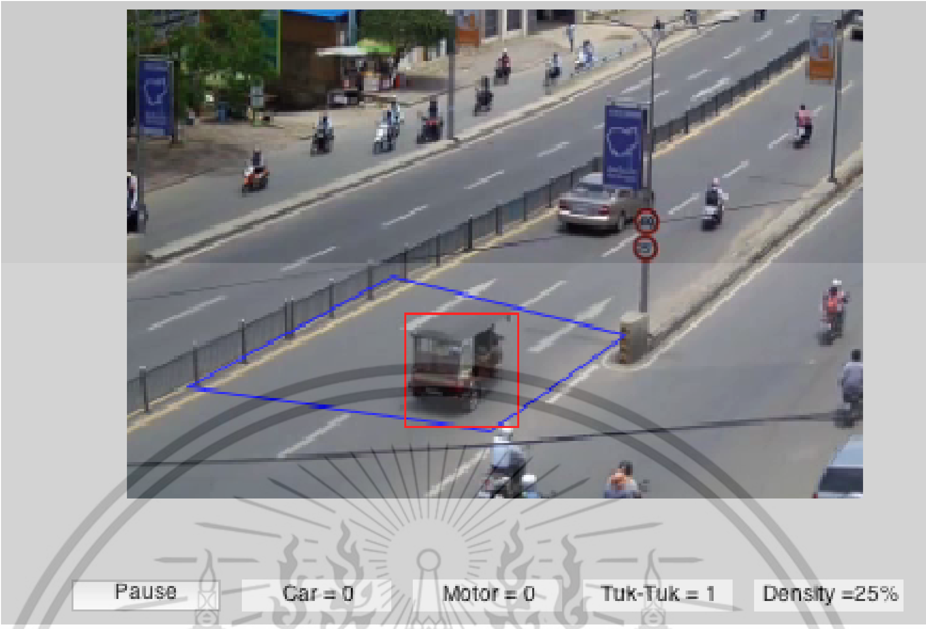


Figure 4.6: Tuk-Tuk classification

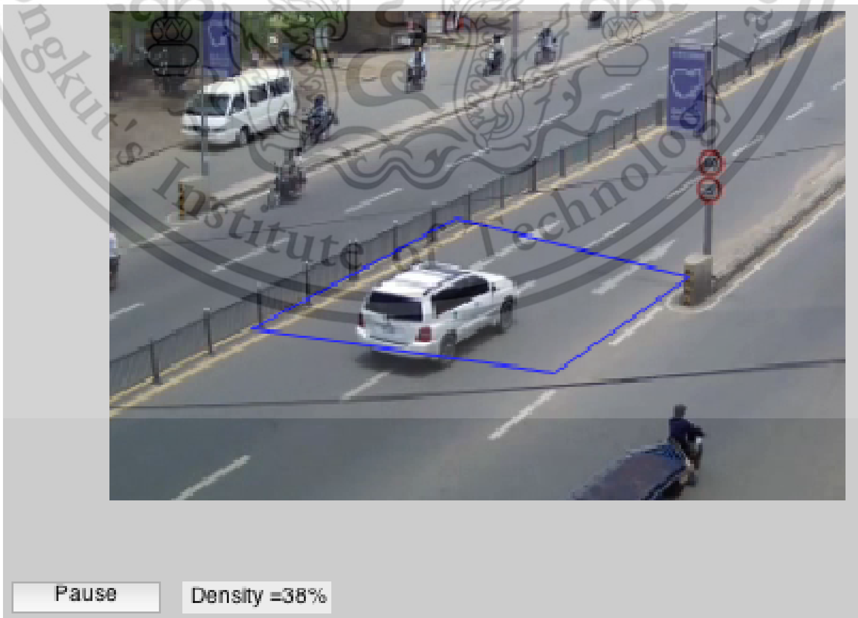


Figure 4.7: Traffic density calculation

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this research, traffic monitoring system has three subsystem i.e. vehicle detection, vehicle classification and traffic density measurement.

Vehicle detection is implemented to detect vehicle's object in the traffic videos. In order to detect vehicle's object, it is processed with four steps. The first three steps are background subtraction, morphological operation, and BLOB detection used for vehicles detection. The last step is skeleton feature extraction used for vehicles occlusion handling. In the background subtraction technique, the background image is removed and foreground image remain as the object. The morphological operation removes noise from object and encloses the object. BLOB detection method is used to extract object from image in order to receive the object information. Skeleton features extraction is the method used to handle vehicle occlusion. Results indicate that skeleton structure and skeleton nodes can handle vehicle detection in occluded scenarios very well. It performs the good result and increase the accuracy of system.

Vehicle classification categories vehicles into three different classes i.e. car, motorcycle, and Tricycle (Tuk-Tuk). The classification step can be implemented after finishing vehicle detection and vehicle occlusion handling step. Then vehicle classification is implemented by using two types of neural network to train and test the dataset from traffic videos and compare the results. From the dataset at hand, the experimental results show that Back-propagation and Radial basis function are not difference from each other both classification and execution time. Results indicate that the system can classify vehicle well in the various traffic video. The information of vehicles in each class obtained from traffic monitoring system can be used for further analysis and management in order to solve traffic congestion problems.

Traffic density measurement is also a part of traffic monitoring system that can be measure by taking the total area of pixel of all vehicle inside ROI divided by the area of ROI.

5.2 Future work

Since the traffic videos used to do the experiment consists of limited number of dataset. So the detection subsystem needs more vehicle's object in the traffic video in order to do the better evaluation. Moreover, It is still small dataset for training and testing the system for vehicle classification, the author also have plan to collect more dataset and apply the proposed method to ensure its performance in the future.

REFERENCES

[1] H. Rabiu, "Vehicle detection and classification for cluttered urban intersection," International Journal of Computer Science, Engineering and Applications (IJCSEA), vol. 3, pp. 37–47, February 2013.

[2] N. Chintalacheruvu and V. Muthukumar, "Video based vehicle detection and its application in intelligent transportation systems," Journal of Transportation Technologies, pp. 305–314, February 2012.

[3] S. M. Al-Garni and A. A. Abdennour, "Moving vehicles detection using automatic background extraction," World Academy of Science, Engineering and Technology, pp. 82–86, 2006.

[4] S. Han, Y. Han, and H. Hahn, "Vehicle detection method using haar-like feature on real time system," World Academy of Science, Engineering and Technology, pp. 455–459, 2009.

[5] B. Sharma, V. K. Katiyar, A. K. Gupta, and A. Singh, "The automated vehicle detection of highway traffic images by differential morphological profile," Journal of Transportation Technology, vol. 4, pp. 150–156, April 2014.

[6] Z. Khalid, A. Mazoul, and M. E. Ansari, "A new vehicle detection method," International Journal of Advanced Computer Science and Applications (IJACSA), pp. 72–76. Special Issue on Artificial Intelligence.

[7] C. Chen and L. Shihui, "Detection and segmentation of occluded vehicles based on skeleton features," 2012 Second International Conference on Instrumentation Measurement, Computer, Communication and Control, pp. 1055–1059.

[8] H. Zhang, H. Zheng, and K. Liu, "The detection and segmentation of vehicle occlusion based on morphology," Journal of Information Computational Science, p. 3063 3072, May 2015.

[9] W. Wu, Z. QiSen, and W. Mingjun, "A method of vehicle classificationusing models and neural networks," p. 3022 3026, 2001.

[10] R. P. Avely, Y. Wang, and G. S. Rutherford, "Length-based vehicle classification using images from uncalibrated video cameras," IEEE intelligent Transportation Systems Conference, p. 737 742, October 2004.

[11] Z. Chen, T. Ellis, and S. A. V. SMIEEE, “Vehicle detection, tracking and classification in urban traffic,” 15th International IEEE Conference on Intelligent Transportation Systems, pp. 951–956, 2012.

[12] S. Messelodi, C. M. Modena, and M. Zanin, “A computer vision system for the detection and classification of vehicles at urban road intersections,” ITC-irst Technical Report T04-02-07, February 2004.

[13] P. Ji, L. Jin, and X. Li, “Vision-based vehicle type classification using partial gabor filter bank,” Proceedings of the IEEE International Conference on Automation and Logistics, pp. 1037–1040, August 2007.

[14] C. Ozkurt and F. Camci, “Automatic traffic density estimation and vehicle classification for traffic surveillance systems using neural networks,” Mathematical and Computational Applications, vol. 14, pp. 187–196, 2009.

[15] H. A. Walid, N. Anjum, U. Tubba, L. Akter, and Z. Asfad, “Traffic density measurement using image processing: An svm approach,” International Journal of Innovative Research in Computer and Communication Engineering, vol. 3, p. 4979 4986, June 2015.

[16] D. A. Osuto, H. O. Absaloms, and E. N. Ndungu, “Automatic road traffic density estimation using image processing algorithms,” Annual Conference on Sustainable Research and Innovation, p. 80 86, May 2016.

[17] N. Sotheany and C. Nuthong, “Vehicle classification using neural network,” ECTI-CON 2017, 2017.

[18] C. Nuthong and N. Sotheany, “Vehicle detection in occluded scenarios,” AUN/SEED-Net Regional Conference for Computer and Information Engineering, pp. 27–32, October 2016.

APPENDICES

APPENDIX A Publications

RCCIE 2016 AUN/SEED-Net

Regional Conference on Computer and Information Engineering

October 3-4, 2016, Yangon, Myanmar

- Committees
- Keynote Talks
- Accepted Papers
- Technical program
- Visa / Hotels / Tours
- Conference Venue

Technical Sponsors:

Financial Sponsors:

Platinum Sponsors:

Media Partner:

Yangon Technological University

RCCIE 2016

Theme: "To enhance academic and research collaboration among ASEAN Universities in the fields of Computer Engineering, IT and Automation Engineering"

RCCIE 2016 is the Regional Conference on Computer and Information Engineering supported by Yangon Technological University and JICA Project for AUN/SEED-Net. This Regional Conference is intended as an Regional Forum in ASEAN Region for academicians, professionals and researchers who wish to share their latest research results, innovative ideas and experiences in the fields of Computer and Information Engineering. This conference will feature paper presentation as well as keynote speech by prominent speakers who will focus on related state-of-art technologies in the areas of the conference and also allows opportunities for participants to discuss future collaborations and activities related to computer engineering and information technology fields.

The potential papers will be encouraged to extend the content and published in ASEAN Engineering Journal (Published by AUN/SEED-Net).

Important Dates

- SUBMISSION DEADLINE
  - ~~15 June 2016~~
  - Extended to 25 June 2016
- AUTHOR NOTIFICATION
  - ~~15 August 2016~~
  - Extended to 22 July 2016
- CAMERA-READY SUBMISSION
  - 25 August 2016
- CONFERENCE DATES
  - 3-4 October 2016

RCCIE 2017 Organizers

Gold Sponsors:

Silver Sponsors:

Copyright © All rights reserved by CEIT, YTU.

Powered by :

Vehicle Detection in Occluded Scenarios

Chaiwat Nuthong [1]
Nou Sotheany [2]
International College
King Mongkut's Institute of Technology Ladkrabang

Abstract

Traffic congestion is one of the problems in urban traffic. There are many systems proposed to solve this problem. Traffic monitoring subsystem is a part of the Intelligent Transportation System(ITS) used to solve traffic congestion issue. Vehicle detection is such an important part of traffic monitoring subsystem that developing the vehicle detection algorithm with high accuracy becomes essential. The proposed vehicle detection method presented in this paper includes several techniques i.e. background subtraction technique, morphological operation, and BLOB detection. Although the vehicles can be well detected, they come in occluded scenarios. These scenarios result the poor vehicle counting performance. In order to solve this problem, skeleton feature extraction has been applied. There are two videos used to perform the experimental setup for system evaluation. It is found that the first video can detect vehicles up to 97% accuracy while the second video can detect up to 93% accuracy. The experimental results illustrate that the proposed system can detect vehicles in occluded scenarios for various videos input with high accuracy.

Keywords: vehicle detection, occluded vehicle detection, skeleton feature extraction

1. Introduction

Traffic congestion has been a major problem of the modern societies. One of the issues that causes traffic congestion problem is an exponential growth of vehicle number. Traffic monitoring subsystem is a part of the Intelligent Transportation System(ITS), which is used to solve traffic congestion problem. This subsystem can extract the valuable information concerning traffic. This information can then be used further in ITS. However, in order to have an efficient traffic monitoring subsystem, vehicle detection feature must be included to fulfill the task. There are many corresponding researches conducted in this filed. H. Rabiu [1] used background subtraction and Kalman filter algorithm to detect vehicles in the detection zone. Because of vehicle occlusion, the result was still unsatisfied. The author put vehicle occlusion issue as the future work. N. Chintalacheruvu, and V. Muthukumar [2] implemented Harris-Stephen Corner detector Method (HSCM) in vehicle detection system. HSCM was used to detect and track a standalone vehicle and determine vehicle counts and speeds at arterial road-ways and freeways. HSCM was implemented and compared to commercial vehicle detection. Because these methods used to detect vehicles in the lanes, HSCM produced better result compared to commercial vehicle detection only in direct angle camera field of view. Al-Garni, and Abdennour [3] proposed background subtraction and edge detection techniques for vehicle detection. The result showed that the proposed methods can detect vehicles higher than 91% accuracy but it still had miss detection problem because of vehicles occlusion. H. Sungji, H. Youngjoon, and H. Hernsoo [4] presented Haar-like feature and the symmetric feature to detect vehicle. It can also remove the shadow of a vehicle on the road effectively. According to the experiment, the result still had error because of illumination changes of road environment. B.Sharma, V. Kumar Katiyar, A. Kumar Gupta, and A. Singh [5] used differential morphology closing profile (DMP) to extract the vehicles automatically from traffic image. The experimental result showed that the DMP technique provided better result comparing to traditional image processing. Z. Khalid, A. Mazoul, and M. El Ansari [6] proposed a new vehicle detection method called AdaBoost classifier for vehicle detection in real time video. AdaBoost classifier method provided an efficient result that was capable to work in real time. Base on previously mentioned researches, vehicles occlusion affects the results of the vehicle detection. Some researches focus on vehicles occlusion detection and segmentation to solve this problem. C. Chen, and

Email: chaiwat.nu@kmitl.ac.th [1]
n.sotheany@gmail.com, 58610015@kmitl.ac.th [2]

S. Liu [7] presented a method to detect and segment occluded vehicles based on the effective skeleton nodes. This method was effectively segment occluded vehicles but it still needed to be improved because skeleton features could be very sensitive to noise. H. Zhang, H. Zheng, and K. Liu [8] proposed morphology and the concavity property of the image to detect vehicles occlusion. From the experimental result, the algorithm showed the high performance in detecting occlusion in many kinds of scenarios. It is found that the researches conducted on vehicle detection in occluded scenarios are still limited in accuracy.

The purpose of this research is to develop a system that can detect vehicle in occluded scenarios at the specific region on the road. This research paper is organized as follows: system overview is described in section II. Methods of the system is illustrated in section III. Section IV shows the experimental setup of the system. Lastly, the conclusion is described in section V.

2. System overview

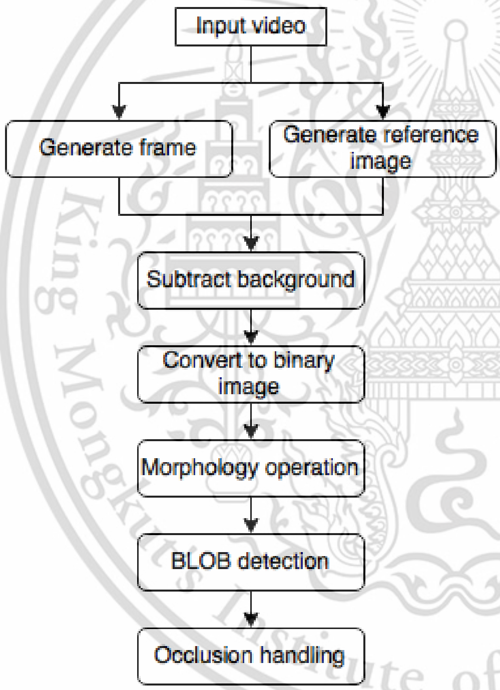


Fig. 1. Vehicle detection system overview

The system of vehicle detection in occluded scenarios is developed with the process as shown in Figure 1. Initially, the system takes a traffic video as its input. The first frame of the video is used as the reference image while the rest is generated as current image. Following this, background subtraction technique is applied by taking current image to subtract with reference image. The output of background subtraction technique is converted to binary image. The next step is morphological operation which is used to remove imperfection of the object such as noise or undesired texture. BLOB detection technique which is used to detect vehicles from binary image is applied afterward. The final process is occlusion handling which performs skeleton feature extraction technique.

3. Methods

This section discusses about the procedure to detect vehicles in each video's frame. Three methods involve are background subtraction technique, morphological operation, and BLOB detection technique. After the vehicles are detected, the occlusion handling procedure applies the skeleton features extraction technique to separate occluded vehicles.

3.1 Background subtraction

Background subtraction is an image processing technique that is used to extract image's foreground. Reference image and current image are the features for this technique. Reference image refers to the image consists only road or background without vehicles or moving objects. Current image refers to image that is extracted one by one iteratively from video. Because of the image consists not only desirable objects but also undesirable objects, region of interest (ROI) is set to focus on the specific zone.

In this research, background subtraction technique is implemented. Moreover, ROI is set to both reference image and current image. Figure 2 shows the reference image with ROI. After setting ROI, both reference image and current image are converted to grayscale images. This is because of background subtraction technique need grayscale image as the input. Background subtraction technique takes current frame as shown in Figure 3 to subtract with reference image. The process of subtraction performs pixel by pixel at the same index as the following formula:

$$\mathrm{diffImg}(i,j) = \mathrm{currImg}(i,j) - \mathrm{refImg}(i,j) \quad (1)$$

Where $\mathrm{diffImg}(i,j)$ is the intensity value at index (i,j) of the new image that obtains from the difference between current image and reference image at index (i,j). $\mathrm{currImg}(i,j)$ is the intensity value of current image at index (i,j) while $\mathrm{refImg}(i,j)$ is the intensity value of reference image at index (i,j).

The result of background subtraction is foreground or vehicles detected. Figure 4 shows the result of background subtraction technique that is already converted to binary image.

Fig. 2. Reference image

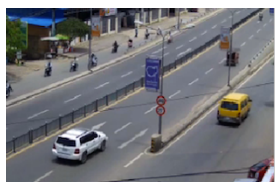


Fig. 3. Current image

Fig. 4. Detected object

3.2 Morphology

Morphology is an image processing operation that processes images based on shapes. Morphological operation pursues the goals of removing noise and undesired texture. The most basic morphological operations are dilation and erosion. Dilation indicated by Equation (2), adds pixels to the boundaries of objects in an image while erosion indicated by Equation (3), removes pixels on object boundaries.

$$g = f \oplus s \qquad (2)$$

$$g = f \ominus s \qquad (3)$$

where g is the output image, f is the input image, and s is the structure element.

Figure 4 shows the objects obtained from background subtraction technique. It still has noise and objects are not enclosed.

In this paper, morphological dilation operation is implemented. It is used to remove noise and add pixels to enclosed holes of the objects. After dilation operation is applied the object becomes completely enclosed as shown in Figure 5.

Fig. 5. Result after morphological dilation operation

3.3 BLOB detection

BLOB stands for Binary Large OBject and refers to a group of connected pixels in a binary image. BLOB detection is a method used to detect the object from the image. The method performs the labelling technique. This technique marks the image pixels with value 1, which are connected, with the same number [9].

BLOB detection technique is implemented in this proposed system. It uses 8-connectivity structure element to access image's pixels to perform labelling technique. After that, the pixels with the same label are grouped as the same object. When vehicle is detected in binary image as shown in Figure 6(a), some information are obtained; e.g. area of pixels are the number of white pixels and bounding box of the vehicle that composes of its location of upper-left corner, width, and height. From the bounding box information, the surrounding box of the vehicle can be drawn in both image as shown in Figure 6. The image is shown in Figure 6(b) presents the detected vehicle in the real video's image.

(a) (b)

Fig. 6. Result of BLOB detection (a) Detected vehicle in binary image (b) Detected vehicle in real video's image

3.4 Vehicle detection in occluded scenarios

Occlusion is one of the problems that affects the accuracy in vehicle detection. Skeleton feature extraction is one of the methods that can handle vehicles detection in occluded scenarios.

The process of skeleton feature extraction for vehicles occlusion is described in three following steps. They are skeleton structure extraction, segmented point extraction, and line segment extraction.

(i) Skeleton structure extraction: The aim of the skeleton structure extraction is to extract a shape feature that represents an object. The idea is thinning the object from its boundary until it remains only one pixel connected. Furthermore, skeleton structure extraction gives node or junction that is used to separate objects in occluded scenarios. Node or junction of skeleton refers to pixels that have more than three neighbour pixels. The number of the nodes represent the number of the vehicles. The detected vehicle has the skeleton structure with two nodes only if the occlusion problem does not occur. However, in the real scenarios the detected vehicle can give skeleton structure with more than two nodes. Skeleton structure that is shown in Figure 7(b) is generated from the vehicle object of Figure 7(a). All (*) signs, which are shown in Figure 7(b), are called the skeleton nodes. Figure 7(b) shows that the skeleton structure extracted from object without occlusion has more than two nodes. To solve this problem, Alternating Sequential Filter (ASF) is used.

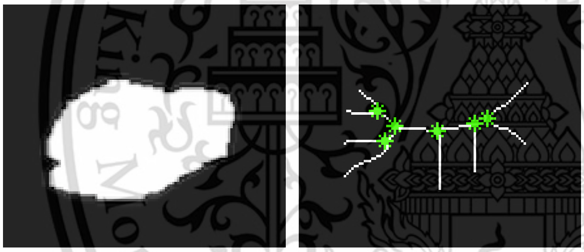
AFS is a composition of opening and closing operations. The aim of this method is to make the boundary of object become more smooth.

In this research, AFS is implemented. After applying the ASF to the object as shown in Figure 7(a), the result of smoothing is shown in Figure 8(a). After that, the skeleton structure and nodes are extracted as shown in Figure 8(b). This method gives the good result because skeleton node consists only two nodes which is the expected number of nodes in unoccluded scenarios.

The case of two vehicles occlusion is shown in Figure 9(a). After applying AFS, the skeleton structure and nodes are extracted as shown in Figure 9(b). In this case, the skeleton structure consists of four nodes. However, two vehicles occluded can give skeleton structure with only three nodes depending on the shape of vehicle object. C. Chen, and S. Liu [7] said that the number of vehicles occluded can be calculated by the following formula:

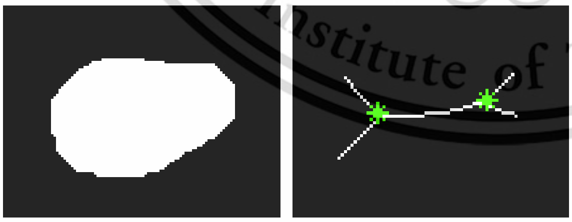
$$N = \frac{\text{Number of nodes}}{2} \quad (4)$$

Where N is the number of vehicles occluded. In case the number of nodes is the odd number greater than 2, the result of N is the floating point number. Because of the number of vehicles cannot be a floating point number, the system rounds it to the next nearest integer number.



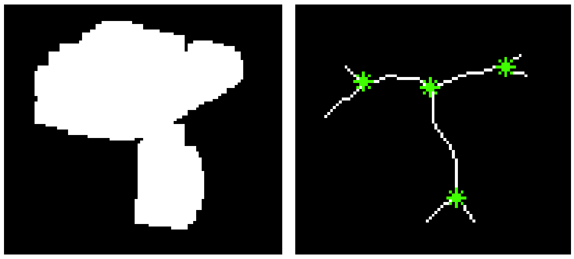
(a) (b)

Fig. 7. Skeleton structure extraction before smoothing the boundary (a) Detected vehicle before smoothing boundary (b) Skeleton structure and nodes



(a) (b)

Fig. 8. Skeleton structure extraction after smoothing the boundary (a) Detected vehicle after smoothing boundary (b) Skeleton structure and nodes



(a) (b)

Fig. 9. Skeleton structure extraction of vehicel occlusion (a) Detected vehicle in occlusion issue (b) Skeleton structure and nodes

(ii) Segmented point extraction: A segmented point is the point that is used to segment occluded vehicles. To find the segmented point depends on number of nodes.

This proposed system discusses two different cases. They are three nodes extracted and four node extracted in two vehicles occlusion problem. In case of three nodes extracted, system will find the middle node which has x value greater than the first node and less than the second node. Then, it calculates distance from the first to middle node and from the middle to the last node by using Euclidean Distance formula:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (5)$$

Where D is the distance between two nodes, (x_1, y_1) is the coordinate of the first node, and (x_2, y_2) is the coordinate of the second node.

The segmented point is found at the middle of the two nodes with the longest distance as in the following formula.

$$\text{seg}(x, y) = \left(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}\right) \quad (6)$$

Where $\text{seg}(x, y)$ is the coordinate of segmented point, (x_1, y_1) is the coordinate of the first node, and (x_2, y_2) is the coordinate of the second node.

In case of four nodes extracted, the segmented point is found at the middle of the second and the third node by using Equation (6). As shown in Figure 10, the second node is the node that has x value greater then the first node and less than the third node. The third node is the node that has x value greater than the second node and less than the fourth node. The segmented point is shown as the (+) sign (see Figure 10).

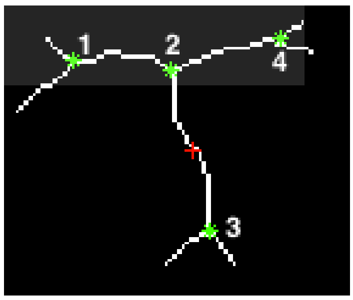


Fig. 10. Segmented point

(iii) Line segment extraction: Line segment refers to the line that is used to separate occluded vehicles. It can be found after segmented point has been identified.

To obtain the line segment, there are two approaches involved. The first approach used every boundary pixels to identify line segment while the second approach is downsampling boundary pixels to determine line segment.

The first approach, all boundary pixels of the object are divided into two parts, upper part $\mathrm{UP} = \{(x_1^u, y_1^u), \ldots, (x_n^u, y_n^u)\}$ and lower part $\mathrm{LP} = \{(x_1^l, y_1^l), \ldots, (x_n^l, y_n^l)\}$. The set of UP points are the points that have y value less than y value of segmented point while the set of LP points are the points that have y value greater than y value of segmented point.

As shown in Figure 11, (o) sign is the segmented point while P1 and P2 are the points on boundary pixels which have y value the same with the segmented point. So that the set of UP points are the boundary pixels above P1 and P2 while the set of LP points are the boundary pixels below P1 and P2. The Euclidean distance is used to calculate distance from the segmented point to UP denoted by $\mathrm{DU} = \{d_1^u, d_2^u \ldots, d_n^u\}$ and from the segmented point to LP denoted by $\mathrm{DU} = \{d_1^l, d_2^l \ldots, d_n^l\}$. The solid lines are shown in Figure 11 present the distance from segmented point to the set of UP points and the set of LP points. Finally, line segment is found by connecting between two points. The first point is the one from the set of points in UP that has minimum distance from segmented point, $min(\mathrm{DU})$, and the second point is the one from the set of LP that has minimum distance from segmented point $min(\mathrm{DL})$. As the result, segment line is found as the dash-line (see Figure 11).

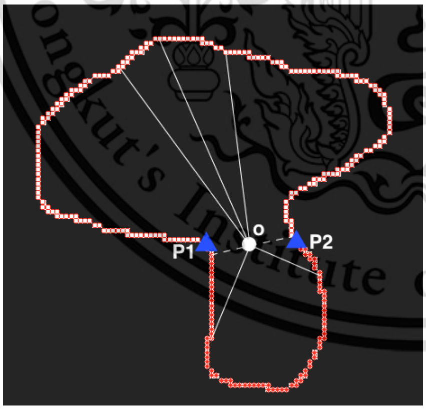


Fig. 11. Line segment of the first approach

The second approach is the modified version of the first approach. It starts with segmented point (o) and four points on the boundary with sign (Δ) as shown in Figure 12. $P1$ and $P2$ are the boundary points that have the same y value with the segmented point while $P3$ and $P4$ are the boundary points that have the same x value with the segmented point. S and S' are the points on the line ($P3P4$). S, which is the point can move from $P3$ to segmented point, is used to do the downsampling for the upper part of the boundary pixels. S', which is the point can move from $P4$ to segmented point, is used to do downsampling for the lower part of the boundary pixels. $S1$ and $S2$ are the downsampling points where its y value equals to y value of S while $S'1$ and $S'2$ are the downsampling points where its y value equals to y value of S'.

The process of downsampling starts when S and S' is moving. The step of moving is defined by 2 pixels in this research. While S and S' are moving in each step, four of downsampling points are identified. The distance from the segmented point to those downsampling pixels is calculated by using Equation (5). After downsampling, the system finds one point from upper part of the boundary pixels and one point from lower part of the boundary pixels that have minimum distance from the segmented point. Finally, line segment is found by connecting those two points. The line segment is the dash-line (see Figure 12). By comparing these two approaches in term of computational resources the second approach is better. This is because it can reduce the computational time with the comparable result to the first approach.

By applying skeleton feature extraction, the problem can be solved. The occluded vehicles can then separated form each other.

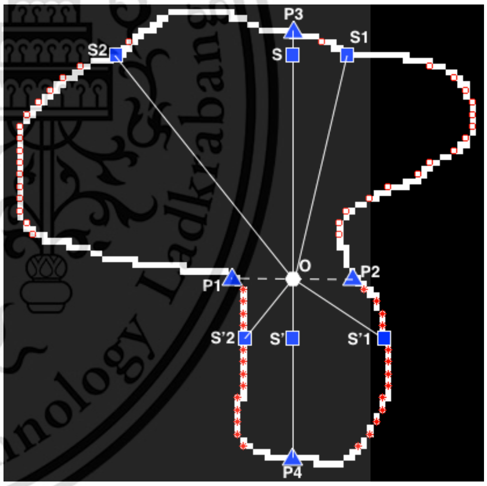


Fig. 12. Line segment of the second approach

4. Experimental results

4.1 Experimental setup

The vehicle detection system runs on iMac 21.5-inch contains 2.7 GHz Intel Core i5 processor with NVIDIA GeForce GT 640M 512 MB graphics processor.

The experiment of this system uses two different videos input as shown in Table 1. From the input video, current image and reference image are generated as the feature of background subtraction. The result of background subtraction is converted into binary image to match with the requirement of BLOB detection that need binary image as the input. Because these input videos have vehicles occlusion problem, skeleton feature extraction is performed to handle occlusion issue.

Videos	Number of Frame	Resolution	Frame rate
1	4301	598*396	60
2	395	1278*720	10

Table 1. Information of input videos

4.2 Result

It is found from experimental result shows that the system can detect vehicles very well as shown in Figure 13 and Figure 14. Table 2 presents the experiment of vehicles detection in both videos without occlusion handling. It results in the low accuracy of vehicle detection. In contrast, Table 3 presents the results obtain from both videos with vehicles occlusion handling. It is found that detection accuracy increases for first video. The second video shows no difference. The reason is that the second video has only one occlusion and the system cannot detect it. However, this system can detect vehicles in the various videos input with the satisfied result. Table 4 presents the execution time of proposed approaches to find line segment. It is applied only on the image in Figure 11 and Figure 12. It shows that the second approach uses less execution time compare to the first approach.

Videos	Real number of vehicle	Detected Vehicle	Rate %
1	48	44	91.67
2	15	14	93.33

Table 2. Result without occlusion handling

Videos	Number of vehicles	Occlusions	Detected Vehicle	Rate %
1	48	4	47	97.91
2	15	1	14	93.33

Table 3. Result with occlusion handling

Segment line technique	Execution time(s)
1	0.357461
2	0.020347

Table 4. Compare execution of segment line technique

Fig. 13. Result of video 1

Fig. 14. Result of video 2

5. Conclusion

This paper proposes the vehicle detecting approach in occluded scenarios. It has four main steps. The first three steps are background subtraction, morphological operation, and BLOB detection used for vehicles detection. The last step is skeleton feature extraction used for vehicles occlusion handling. In the background subtraction technique, the back- ground image is removed and foreground image remain as the object. The morphological operation removes noise from object and encloses the object. BLOB detection method is used to extract object from image in order to receive the object information. Skeleton features extraction is the method used to handle vehicle occlusion. Results indicate that skeleton structure and skeleton nodes can handle vehicle detection in occluded scenarios very well. It performs the good result and increase the accuracy of system.

Acknowledgment

The authors would like to thank AUN/Seed-net that provides the financial support to conduct this research.

Reference

[1] H. Rabiu, “Vehicle Detection and Classification for Cluttered Urban Intersection”, International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.3, No.1, February 2013.

[2] N. Chintalacheruvu, and V. Muthukumar, “Video Based Vehicle Detec- tion and Its Application in Intelligent Transportation Systems”, Journal of Transportation Technologies, 2012, 2, 305-314.

[3] S. M. Al-Garni, and A. A. Abdennour, “Moving Vehicles Detection using Automatic Background Extraction”, World Academy of Science, Engineering and Technology 24 2006.

[4] H. Sungji, H. Youngjoon, and H. Hernsoo, “Vehicle Detection Method using Haar-like Feature on Real Time System”, World Academy of Science, Engineering and Technology 59 2009.

[5] B. Sharma, V. Kumar Katiyar, A. Kumar Gupta, and A. Singh, “The Automated Vehicle Detection of Highway Traffic Images by Differential Morphological Profile”, Journal of Transportation Technologies, 2014, 4, 150-156.

[6] Z. Khalid, A. Mazoul, and M. El Ansari, “A new vehicle detection method”, (IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence.

[7] C. Chen, and S. Liu, “Detection and Segmentation of Occluded Vehicles Based on Skeleton Features”, 2012 Second International Conference on Instrumentation Measurement, Computer, Communication and Control.

[8] H. Zhang, H. Zheng, and K. Liu, “The Detection and Segmentation of Vehicle Occlusion Based on Morphology”, Journal of Information Computational Science 12:8 (2015) 30633072.

[9] BLOB Analysis (Introduction to Video and Image Processing) Part1. Retrieved from http://what-when-how.com/introduction-to-video- and-image-processing/blob-analysis-introduction-to-video-and-image- processing-part-1/


ECTI-CON 2017
Phuket, Thailand
Contact Us
All queries to ECTI-CON 2017 can be addressed to the email: ecticon2017@ectithailand.org
Home
Call For Papers
Important Dates
Special Sessions
Author's Kit
Paper Submission
Paper Registration
Technical Program
Keynote Speakers
Conference Committee
Conference Venue
Visa Information
Sponsors
About ECTI-CON
Contact Us
User Login Site >>
ECTI Association
Like Page
928 likes
Be the first of your friends to like this
KMITL ECTI IEEE
Ambient Backscatter Assisted Wireless Powered Communications
Prof. Dusit Niyato
ECTI-CON 2017
Like Page
222 likes
27-30 June 2017
Be the first of your friends to like this
ECTI-CON 2017
about a month ago
Good news! ECTI-CON proceedings are now available in IEEE Xplore.
24
1
Share
27-30 June 2017, Phuket, Thailand

Vehicle Classification using Neural Network

Nou Sotheany
International College
King Mongkut's Institute of Technology
Ladkrabang
Email: n.sotheany@gmail.com/58610015@kmitl.ac.th

Chaiwat Nuthong
International College
King Mongkut's Institute of Technology
Ladkrabang
Email: chaiwat.nu@kmitl.ac.th

***Abstract*—Traffic congestion is one of the major problems in the big cities. Many systems are proposed to solve this problem, for example, Intelligent Transportation System(ITS). In general, ITS consists of many subsystems including traffic monitoring. The monitoring system can provide the information of vehicles that pass through the monitoring area. This information can suggests about the status of the traffic. For this reason, developing a system that can classify vehicle with high accuracy plays important role to support ITS. The proposed system of vehicle classification using neural network presents in this paper includes other techniques i.e. vehicle detection and vehicle occlusion handling. Furthermore, Back-propagation neural network and Radial basis function network are implemented for vehicle classification. The training dataset of these two neural network classifiers is taken from a traffic video while the testing dataset is taken from another traffic video. The experimental results show that both techniques can classify vehicles only 71% accuracy because the different size of detected vehicles in those two videos. In order to solve this problem, affine transform function is implemented to scale the object of the testing video. After implementing affine transformation to the system, the accuracy of Back-propagation technique increases to 86% while Radial basis function remains the same. The experimental results illustrate that Back-propagation neural network can classify vehicles with higher accuracy and better performance than Radial basis function network.**

I. Introduction

Many big cities have problem with traffic congestion. The growth of number of vehicle is one of the issues that causes traffic congestion problem. As the solution to current issue, Intelligent Transportation System(ITS) is implemented. ITS generally consists of a traffic monitoring system that provides traffic information. The information of the number of vehicle in each class can then be used for monitoring the traffic congestion problem. In order to classify vehicles into calsses, vehicle detection is needed. C. Nuthong and N. Sotheany [1] implemented vehicle detection in occluded scenarios. Background subtraction technique, morphological operation, and BLOB detection are used to detection vehicles in the surveillance video. Skeleton feature extraction is applied to solve vehicle occlusion problem. The experimental results illustrated that the system can detect vehicle up to 97% accuracy in occluded scenarios. There are various researches conducted on vehicles classification issues. Z. QiSen and W. Mingjun [2] presented a novel method of vehicle classification using parameterized model and neural networks. Multi-layer perceptron networks is implemented as the classifier. This novel method can recognise vehicles with the correct rate more than 91%. However, the wrong classification happened with the effect of vehicle occlusion and the shadow of vehicle. They planned to improve their system in the future by collecting more and better data for training and also find better features. Z. Chen, T. Ellis and S. A. V. SMIEEE [3] used Support Vector Machine(SVM) as a classifier to classify vehicles into four categories i.e. car, van, bus, and motorcycle. This system can classify vehicle with 94.69% accuracy under varying illumination and weather conditions. The problem is the multiple vehicles are too close and it causes occlusion problem. Moreover, the similarity of vehicle's colour and background's colour also causes the classification problem. S. Messelodi, C. M. Modena and M. Zanin [4] presented vehicle classification based on a model-based matching technique to classify vehicle into seven categories. According to the experimental result, the system can classify vehicle with 87.3% accuracy. This problem happened because of shadows, reflections, occlusions, and pedestrian. They planned to solve the problem by separating a vehicle from its shadow, identifying and excluding pedestrian and uninteresting moving objects from the traffic statistics, and solve the occlusion problem. P. Ji, L. Jin, X. Li [5] proposed vehicle classification method using partial Gabor filter bank to classify five vehicles categorization: sedan, van, hatchback sedan, bus and van truck. The experimental results showed that the classification rate can reach to 95.17%. The problem happened because of the sampling method is still limited. C. Ozkurt and F. Camci [6] presented vehicle classification and traffic density calculation methods using neural networks. They classified vehicles into three categories i.e. car, van, and bus with 98% accuracy. The problem that affected the result is vehicle misclassification. After vehicles was identified, traffic density was calculated. In this study, they calculated traffic density of each type of vehicle by taking number of vehicle in each type divided by a period of time. Based on the experimental result, traffic density was calculated without error. However, the classification of vehicle types is performed with some error. Some of the vehicles have been misclassified.

Due to the limitations of the conducted researches described above, this paper proposes two different types of neural network to classify vehicles by categories and compare the results.

II. System overview

The system is developed with the process as shown in Figure 1. Initially, the proposed system takes a video as an input. Then, vehicle detection is implemented by using background subtraction, morphological operation, and BLOB detection. After vehicles are detected, the problem of occlusion scenarios is found. In order to handle vehicle occlusion

978-1-5386-0449-6/17/$31.00 ©2017 IEEE

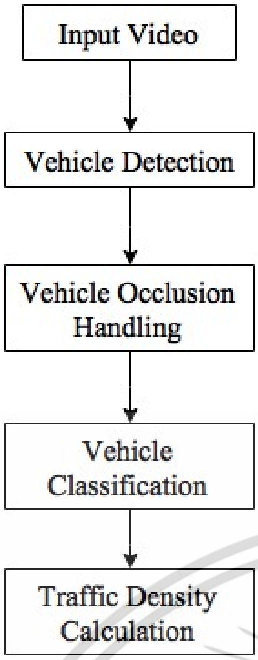


Fig. 1: System overview

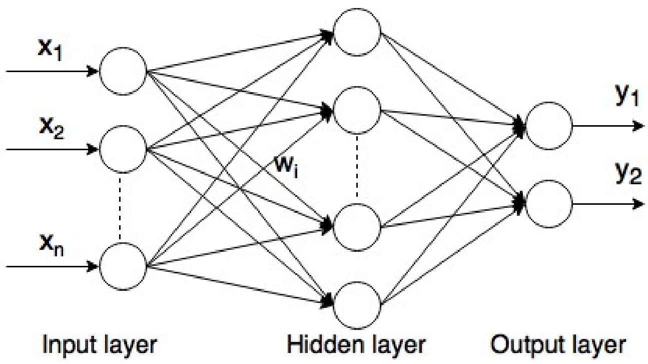


Fig. 2: Implemented Back-propagation Structure

problem, Skeleton feature extraction method is implemented [1]. Following this scheme, vehicle classification is used to classify vehicles into categories in order to provide information of vehicle's types that pass the monitoring area. Lastly, traffic density calculation calculates the density of the traffic in the area of interest and measure the status of the traffic whether it is busy or not. However, this paper focus on vehicle classification that neural network is implemented.

III. VEHICLE CLASSIFICATION

Vehicle classification is a subsystem used to categorize vehicles into different classes. The goal of vehicle classification is to know the amount of vehicles in each class which pass through the monitoring area. According to the proposed system, vehicles are classified into three different categories i.e. car, motorcycle, and tricycle(Tuk-Tuk). In order to classify vehicles, neural network back-propagation and radial basis function network are implemented. The results and performance of these two types of neural network are compared. The structure of back-propagation neural network is shown in Figure 2 with three layers i.e. input, hidden, and output layer. $\{x_1, ..., x_n\}$ are the inputs of the network. Hidden layer consists neurons and each neuron consists of summation function and activation function. Summation function will sum the multiplication of each input x_i with its weight w_i while activation function $f(R)$ of hidden layer is chosen to be sigmoid function presented as Equation (1). As a result, the output of each neuron in hidden layer is illustrated as Equation (2).

$$f(R) = \frac{1}{1 + e^{-R}} \tag{1}$$

where R is the output of the summation function computed by Equation (2).

$$out = f(\sum_{i=1}^{n} x_i w_i) \tag{2}$$

where x is input and w is weight of input x.

The last layer of neural network is output layer. This layer in each neuron also consists of summation function and activation function similar to the hidden layer. The output of each neuron is also defined as Equation (2).

Radial Basis Function Network(RBFN) is an artificial neural network that uses radial basis functions as activation functions. $\{x_1, ..., x_m\}$ are the inputs of the network in the structure of RBFN (see Figure 3). RBFN has only one hidden layer with activation functions φ_i. Gaussian function will be used as the activation function of the neuron in the hidden layer. The function can be stated as shows in Equation (3).

$$\varphi = \exp\left(-\frac{r^2}{2\sigma^2}\right) \tag{3}$$

where $r = \|x_i - t_i\|$, x_i is the input and t_i is the center of the neuron in hidden layer. Moreover, t_i is generated randomly from the input and $\sigma = \frac{d}{\sqrt{n}}$, which d is the maximum distance between any two centers t and n is the number of centers.

The output y is the target class and it will be used to define the weight (see Equation (4)).

$$w_1\varphi_1(\|x_1 - t_1\|) + ... + w_m\varphi_m(\|x_m - t_m\|) = y \tag{4}$$

These neural networks need to be trained with the dataset before it can classify vehicles into three different categories i.e. car, motorcycle, and tricycle(Tuk-Tuk). From the dataset, there are two different inputs x_1 and x_2 to the neural network system. x_1 represents the ratio between width and heigh of the vehicle while x_2 represents the number of pixels in the area of vehicle. These two feature inputs can be generated after vehicle detection is implemented. Vehicle detection using BLOB detection can detect vehicles and gives the value of its width, height and the number of pixels of its area. So the two feature inputs to the neural network is identified.

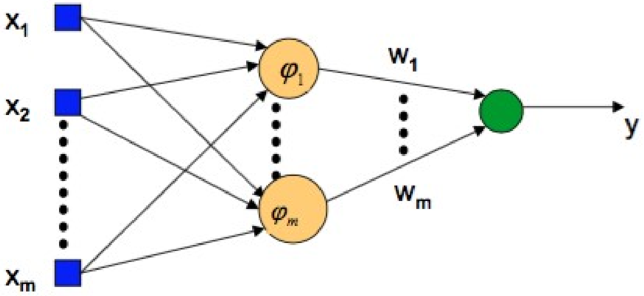


Fig. 3: Implemented Radial Basis Function Network Structure

There are two videos used to train and test the neural networks for vehicle classification. The dataset from the first video is used to train the network while data set from the second video is used to test the network. The problem of vehicle classification occurs with this approach. This is because the distance from camera to object in those two videos are different. Such different distance affects the size of detected vehicles in the two videos. Affine transformation, which is a function that maps an object from an affine space to another space with structure preserving, is then applied to solve this problem. It consists of various operations such as translation, rotation, and scaling. In this research, scaling operation is implemented to scale the size of object of the testing video to be similar with the training video. In order to apply the scaling, nearest neighbor interpolation is the scaling methods implemented in this system. Nearest neighbor is the simplest and fastest implementation of image scaling technique. This scaling technique requires the size of the original image and the size of the output image to be scaled. In order to apply this technique two assumptions are made:

1) The original image, $I_1(v, w)$, is defined in vw coordinate system with width and height of w_1 and h_1, respectively.
2) In the similar manner, an output image, $I_2(x, y)$ is defined in xy coordinate system with width and height of w_2 and h_2, respectively.

In this system the size of the output image is assigned to 50% smaller than the original image. Firstly, the ratio between the width of the original and output image and the ratio between the heigh of the original image and the output image are defined by Equation (5) and Equation (6).

$$ratio_w = \frac{w_1}{w_2} \quad (5)$$

$$ratio_h = \frac{h_1}{h_2} \quad (6)$$

Then the value of pixel in the output image is calculated by Equation (7). This method needs to swap x and y as well as v' and w' to find the value of the pixel of the output image.

$$I_2(y, x) = I_1(w', v') \quad (7)$$

where w' and v' are defined by Equation (8) and Equation (9), respectively.

$$w' = \frac{y}{ratio_h} \quad (8)$$

$$v' = \frac{x}{ratio_w} \quad (9)$$

After nearest neighbor interpolation is calculated, the object in the second video is scaled to 50% smaller that make its size similar to the first video as shown in Figure 4. The original vehicle's object in the second video before Affine transformation is applied shown in Figure 4(a). This vehicle's object is large compare to the size of vehicle's objects in training video. In order to avoid the problem, affine transformation is implemented to scale those vehicle's objects 50% and the result of scaling shown in Figure 4(b). So affine transformation using nearest neighbor interpolation is implemented to scale the vehicle's objects in testing video to make its size similar to training video.

(a) Before (b) After

Fig. 4: Vehicle's object before and after Affine transformation

IV. Experimental Result

A. Experimental setup

This system runs on iMac 21.5-inch contains 2.7 GHz Intel Core i5 processor with NVIDIA GeForce GT 640M 512 MB graphics processor.

There are two traffic videos used in this research. The information of each video is shown in Table I. In order to perform vehicle classification, vehicle detection and vehicle occlusion handling are needed. This is because a classifier needs data of vehicle's object from detection step.

Videos	Number of Frame	Resolution	Frame rate (fps)
1	4301	598*396	60
2	395	1278*720	10

TABLE I: Information of input videos

B. Results

One of two videos is used for training both Back-propagation neural network and Radial basis function network while one another video is used to test these two neural network. Training set is the dataset taken from the first video while testing set is taken from the second video. Table II shows that the number of vehicles taken from the first video for training the network is 69 which is categorized to 20 cars, 44 motorcycles, and 5 tricycles (Tuk-Tuk) while 7 dataset taken from the second video for testing consists of 1 car, 5 motorcycles, and 1 tricycles (Tuk-Tuk). The classification results of both Back-propagation and Radial basis function network are the same before applying affine transformation. They can classify correctly motorcycle but incorrectly car and tricycles. Table II shows that both Back-propagation and Radial basis function network can classify vehicle up to 71% accuracy. The misclassification occurs because the size of vehicles in testing video and training video are different. This problem is solved by using affine transformation technique to scale the size of vehicles of testing video to be similar with training video.

Classes	Training	Testing	Classification			
			Back-propagation		RBFN	
			Before AF	After AF	Before AF	After AF
Car	20	1	0	1	0	0
Motorcycle	44	5	5	4	5	5
Tuk-Tuk	5	1	0	1	0	0
Total	69 (100%)	7 (100%)	5 (71%)	6 (86%)	5 (71%)	5 (71%)

TABLE II: Comparison between Back-propagation and RBFN. Note that AF refers to Affine Transformation.

After affine transformation is applied, the results in Table II shows that the accuracy of classification of Back-propagation technique is increate to 86% while Radial basis function network stay the same accuracy. It is also found that Back-propagation uses slightly less execution time compared to Radial basis function network. This is illustrated in Table III. The experimental results suggest that Back-propagation neural network is preferred for both accuracy and execution time criterion. The classification results is illustrated in Figure 5 and Figure 6. The amount of vehicles in each class provide valuable information for traffic monitoring system.

Neural network types	Execution time(s)
Back-propagation	1.218395
Radial basis function	1.664398

TABLE III: Training execution time comparision

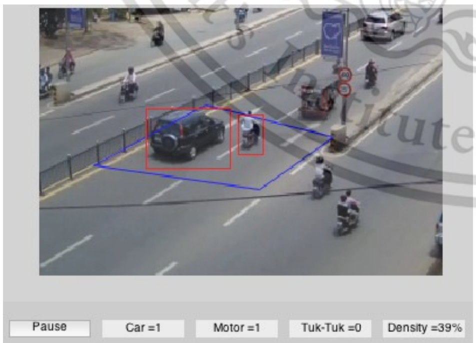


Fig. 5: Car and motorcycle classification

V. Conclusion

This research paper proposed the technique which classify vehicles into three different classes i.e car, motorcycle, and

Fig. 6: Tuk-Tuk classification

Tuk-Tuk. The classification step can be implemented after finishing vehicle detection and vehicle occlusion handling step. Then vehicle classification is implemented by using two types of neural network to train and test the dataset from traffic videos and compare the results. From the dataset at hand, the experimental results show that Back-propagation better than Radial basis function both classification and execution time. Moreover, Traffic density measurement is also a part of traffic monitoring system that can be measure by taking the total area of pixel of all vehicle inside ROI divided by the area of ROI. Results indicate that the system can classify vehicle well in the various traffic video.

The information of vehicles in each class obtained from traffic monitoring system can be used for further analysis and management in order to solve traffic congestion problems. It is still small dataset for training and testing the system, the author also have plan to collect more dataset and apply the proposed method to ensure its performance it the future.

Acknowledgment

The authors would like to thank AUN/Seed-net and King Mongkuts Institute of Technology Ladkrabang Research Fund (KREF015808).

References

[1] C. Nuthong and N. Sotheany, *Vehicle Detection in Occluded Scenarios*, AUN/SEED-Net Regional Conference for Computer and Information Engineering, 27-32, 2016

[2] Z. QiSen and W. Mingjun, *A Method of Vehicle ClassificationUsing Models and Neural Networks*, IEEE, 3022-3026, 2001

[3] Z. Chen, T. Ellis and S. A. V. SMIEEE, *Vehicle Detection, Tracking and Classification in Urban Traffic*, 15th International IEEE Conference on Intelligent Transportation Systems, 951-956, 2012

[4] S. Messelodi, C. M. Modena and M. Zanin,*A computer vision system for the detection and classification of vehicles at urban road intersections*, ITC-irst Technical Report T04-02-07, February 2004

[5] P. Ji, L. Jin, X. Li, *Vision-based Vehicle Type Classification Using Partial Gabor Filter Bank*, Proceedings of the IEEE International Conference on Automation and Logistics,1037-1040 ,August 2007

[6] C. Ozkurt and F. Camci, *AUTOMATIC TRAFFIC DENSITY ESTIMATION AND VEHICLE CLASSIFICATION FOR TRAFFIC SURVEILLANCE SYSTEMS USING NEURAL NETWORKS*, Mathematical and Computational Applications, Vol. 14, No.3, 187-196, 2009

AUTHOR BIOGRAPHY

Name	Sotheany Nou
Degree	Master of Engineering
Date of graduation	15th December 2017
Date of Birth	07th November 1992
Place of Birth	Kampong Siem District, Kampong Cham Province, Cambodia

Undergraduate and Graduate Education:

Master of Engineering in Computing in Engineering Systems , King Mongkuts Institute of Technology Ladkrabang, Bangkok, 2017

Bachelor degree in Information and Communication Engineering , Institute of Technology of Cambodia, Phnom Penh, Cambodia, 2015

Major: Computing in Engineering Systems

Presentations and Publications:

[1] C. Nuthong and N. Sotheany, Vehicle detection in occluded scenarios, AUN/SEED-Net Regional Conference for Computer and Information Engineering, pp. 2732, October 2016.

[2] N. Sotheany and C. Nuthong, Vehicle classification using neural network, ECTI-CON 2017, 2017.