

國立交通大學

資訊科學與工程研究所

碩士論文

交通監控影片中日夜車流壅塞之分析及評估

Traffic Congestion Classification for Daytime and Nighttime

Surveillance Videos

研究生：蔡立武

指導教授：李素瑛 教授

中華民國 一百年八月

交通監控影片中日夜車流壅塞之分析及評估

研究生:蔡立武

指導教授:李素瑛 教授

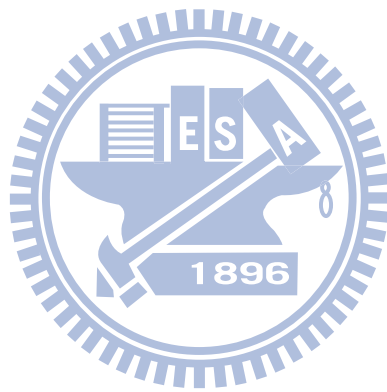
國立交通大學資訊科學與工程研究所

摘 要

近年來，各國致力於智慧型運輸系統的開發，以期能透過即時的交通訊息傳輸與整合來提升交通運輸品質。對於一般大眾，若能從中獲得即時的交通壅塞資訊則是更加的實用。目前，由攝影機所構成的交通監控系統已成為交通訊息偵測的主流，然而大部分的研究僅限於監控影片中交通事件的自動分析，例如交通事故與違規事件，但這無法幫助我們得知當下車流的壅塞情形。所以，處理監控影片以提供民眾最即時的交通壅塞資訊是迫切需要的。在本論文中，我們提出了一個適用於白天與晚上交通監控影片的壅塞程度評估系統能將壅塞程度分為五個等級。

想要從影片中評估交通壅塞程度，視訊處理的技巧與相關的知識是不可或缺的。對於白天的影片，我們利用背景相減法來找出道路上的車子，而在晚上的影片中，則是透過車頭燈的偵測來找出車子的位置。當車子擷取出來之後，我們使用虛擬的偵測器來蒐集交通訊息，藉此即時地評估所偵測到的車流壅塞程度。除此之外，我們也針對道路偵測、車流方向判斷與車道偵測等問題提出解決方法，以提升整個系統的即時性與完整性。最後，我們利用高速公路上的交通監控影片來驗證我們所提出系統的性能，並獲得了令人滿意的結果。

關鍵字：交通壅塞、車道偵測、夜間監控、車燈偵測、虛擬偵測器、智慧型運輸系統



Traffic Congestion Evaluation for Daytime and Nighttime Surveillance Videos

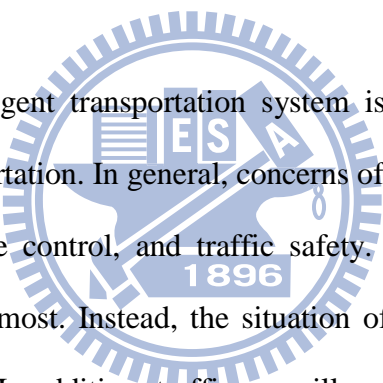
Student: Li-Wu Tsai

Advisor: Prof. Suh-Yin Lee

Institute of Computer Science and Engineering

National Chiao Tung University

Abstract

The logo of National Chiao Tung University is a circular seal. It features a central shield with a book and a torch, flanked by the letters 'E', 'S', and 'A'. Below the shield is the year '1896'. The entire seal is surrounded by a gear-like border.

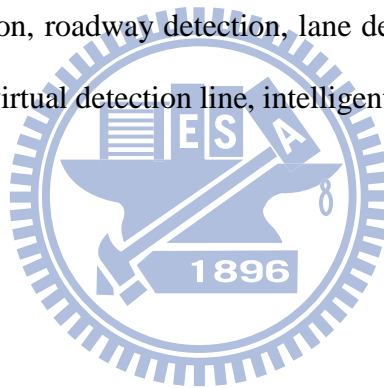
In recent years, intelligent transportation system is developed to promote the quality of the traffic transportation. In general, concerns of the traffic control center are traffic management, vehicle control, and traffic safety. However, they are not the issues that people concern most. Instead, the situation of traffic congestion is much more useful for the public. In addition, traffic surveillance systems have been widely used for monitoring the roadways. There have been many researches on video analysis of traffic activities such as traffic accidents and violations, but these researches still cannot help people get to know the traffic congestion situation. Therefore, we intend to develop techniques to process traffic surveillance videos for providing people with instant traffic congestion information. In this thesis, a traffic congestion classification framework is proposed for identifying the traffic congestion levels in daytime and nighttime surveillance videos. The degrees of traffic congestion are classified into five levels: jam, heavy, medium, mild and low.

In order to analyze the traffic congestion levels from videos, image processing techniques and the knowledge of classification are indispensable. In the proposed

framework, moving vehicles are extracted by background subtraction during the day and by headlight detection at night. Afterward, virtual detectors and virtual detection line are utilized to evaluate and classify the traffic congestion levels in daytime and nighttime surveillance videos, respectively. Moreover, methods of bidirectional roadway detection and lane detection are proposed to extract the consistent features of roadway for the requirements of real-time response and robustness of the frameworks.

In the experiments, we use real freeway surveillance videos captured at day and night to demonstrate the performances on accuracy and computation. Satisfactory experimental results validate the effectiveness of the proposed framework.

Keywords: traffic congestion, roadway detection, lane detection, nighttime, headlight detection, virtual detector, virtual detection line, intelligent transportation system



Acknowledgement

I greatly appreciate the patient guidance, the encouragement, and valuable comments of my advisor, Prof. Suh-Yin Lee. Without her graceful advices and help, I cannot complete this thesis. Besides, I want to extend my thanks to my friends and all members in the Information System laboratory, Yee-Choy Chean, Hui-Zhen Gu, Chien-Peng Ho, Hua-Tsung Chen especially. They gave me a lot of suggestions and shared their experiences.

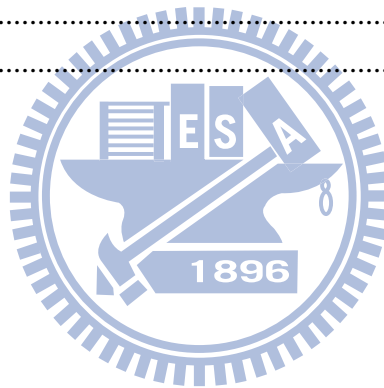
Finally, I would like to express my appreciation to my dear family for their supports. This thesis is dedicated to them.



Table of Contents

Abstract (Chinese)	i
Abstract (English)	iii
Acknowledgement	v
Table of Contents	vi
List of Figures	viii
List of Tables.....	ix
Chapter 1. Introduction	1
Chapter 2. Related Work	5
2.1 Related work on Daytime Surveillance	5
2.1.1 Roadway Detection	5
2.1.2 Object Detection	7
2.1.3 Shadow Elimination.....	9
2.1.4 Traffic Surveillance Video Analysis	11
2.2 Related Work on Nighttime Surveillance	13
2.2.1 Nighttime Image Enhancement.....	13
2.2.2 Object Detection and Tracking	14
Chapter 3. Proposed Framework Overview	16
3.1 Daytime Traffic Congestion Classification.....	17
3.2 Nighttime Traffic Congestion Classification	18
Chapter 4. Daytime Traffic Congestion Classification	20
4.1 Initialization Procedure	20
4.1.1 Roadway Detection	21
4.1.2 Bidirectional Roadway Analysis.....	23
4.1.3 Virtual Detector Installation.....	27
4.2 Vehicle Detection	30
4.2.1 Mixture of Gaussians	30
4.2.2 Shadow Elimination.....	33
4.3 Traffic Congestion Classification	37
4.3.1 Traffic Parameter Estimation	37
4.3.2 Traffic Congestion Evaluation and Classification	40
Chapter 5. Nighttime Traffic Congestion Classification.....	42
5.1 Headlight Detection	42
5.1.1 Bright Region Detection	42
5.1.2 Headlight Shape Validation.....	45
5.2 Vehicle Detection	46

5.2.1 Headlight Correlation.....	46
5.2.2 Headlight Grouping	49
5.3 Traffic Congestion Classification	50
Chapter 6. Experimental Results and Discussions.....	52
6.1 Daytime Traffic Congestion Classification.....	52
6.1.1 Roadway Detection.....	52
6.1.2 Bidirectional Roadway Analysis.....	55
6.1.3 Virtual Detector Installation.....	57
6.1.4 Traffic Congestion Classification	59
6.1.5 Performance of Execution Time	64
6.2 Nighttime Traffic Congestion Classification	65
6.2.1 Headlight Detection	65
6.2.2 Vehicle Detection	68
6.2.3 Traffic Congestion Classification	70
6.2.4 Performance of Execution Time	72
Chapter 7. Conclusions	73
Bibliography	75

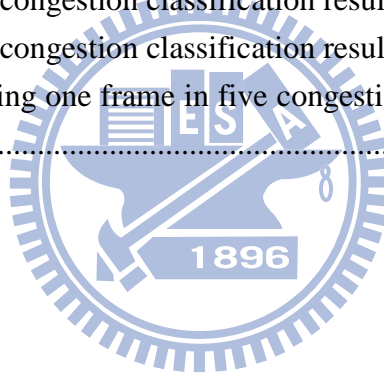


List of Figures

Figure 1. Captured frames in five congestion levels.....	2
Figure 2. Lane markings with different shapes and colors on the roadway.....	6
Figure 3. Three roadway types.....	7
Figure 4. Cast Shadow and self shadow	10
Figure 5. Generation procedure of time-spatial image.....	13
Figure 6. Overview of the proposed framework	16
Figure 7. A traffic scene of a bidirectional roadway consisting of six lanes.....	20
Figure 8. An example of roadway detection	23
Figure 9. An example of bidirectional roadway analysis.....	26
Figure 10. MBSAS processing on the history of the central points of vehicles	29
Figure 11. Results of virtual detectors installation on a bidirectional roadway	30
Figure 12. Gradient images of foreground and its relevant background.....	35
Figure 13. Result of shadow elimination by using gradient feature	36
Figure 14. Operators of Sobel filter	36
Figure 15. Determination of traffic congestion thresholds for NCD	41
Figure 16. Headlights with different color lights	43
Figure 17. Results of bright region detection.....	44
Figure 18. The parameters of height relationship for two headlights	48
Figure 19. The parameters of edge relationship of two headlights	48
Figure 20. Experimental results of roadway detection.....	54
Figure 21. Experimental results of bidirectional roadway analysis	56
Figure 22. Experimental results of virtual detector installation.....	58
Figure 23. Classification accuracy in five levels by different parameters (NCD).	61
Figure 24. Classification accuracy in five levels by different parameters (SVM).	63
Figure 25. Experimental results of headlight detection	67
Figure 26. Experimental results of vehicle detection by headlight grouping	69

List of Tables

Table 1. Luminance and color variation from the center to boundary of headlights. .43	.43
Table 2. Distribution of experimental daytime surveillance video data.....59	59
Table 3. Comparison of classification performance with different features (NCD). ..61	61
Table 4. Daytime traffic congestion classification results by using <i>Cong_{day}</i> (NCD). .62	62
Table 5. Daytime traffic congestion classification results by using <i>Dens</i> (NCD).....62	62
Table 6. Daytime traffic congestion classification results by using <i>Sp</i> (NCD).62	62
Table 7. Comparison of classification performance with different features (SVM). ..63	63
Table 8. Daytime traffic congestion classification results by <i>Cong_{day}</i> (SVM).....64	64
Table 9. Daytime traffic congestion classification results by <i>Dens</i> (SVM).64	64
Table 10. Daytime traffic congestion classification results by <i>Sp</i> (SVM).64	64
Table 11. Time for processing one frame in five congestion levels in daytime video.65	65
Table 12. Distribution of experimental nighttime surveillance video data.70	70
Table 13. Nighttime traffic congestion classification results by <i>Cong_{night}</i> (NCD).71	71
Table 14. Nighttime traffic congestion classification results by <i>Cong_{night}</i> (SVM).....72	72
Table 15. Time for processing one frame in five congestion levels in nighttime video.72	72



Chapter 1. Introduction

Traffic surveillance systems have been widely used for monitoring roadways in recent years. Unfortunately, the repository of the captured videos is so large that it is almost impossible to manually understand the contents of the videos. In fact, it is useful to utilize these traffic video data which can be processed to extract abundant traffic information for real-time intelligent transportation applications. Therefore, plenty of researches have been focused on automatic traffic events analysis such as traffic accidents, violation, and congestion. In this thesis, we investigate the event that the public concern most: roadway traffic congestion.

In the past, when traffic jams occurred, the police or drivers would inform the traffic control centers, and people detoured to avoid the traffic jams after radio station broadcasted that information. Nowadays, a variety of sensors such as loop detector, infrared detector, and Closed Circuit Television camera are used to gather the instant traffic information in traffic control system. However, the cameras are the particular devices that not only can observe the traffic situation but also record all events that happen on roadways all the time, which provides us with more plentiful traffic information. Moreover, due to the advantage of non-invasive installation, the cameras have distributed over all freeways and the main roadways in metropolises. Thus, it facilitates the possibility of the establishment of the complete traffic information. If a traffic surveillance system can automatically analyze the level of traffic congestion from traffic surveillance videos, the congestion message can be immediately provided for the public. Moreover, with the rapid development of intelligent mobile devices such as smart phone and personal digital assistant, drivers can earlier get traffic information and recommended alternate routes for avoiding traffic jam.

Aiming at the benefits arising from integration of video analysis technique and intelligent mobile devices, real-time classifying the traffic congestion in daytime and nighttime surveillance videos is the goal we are going to achieve in this thesis. The traffic congestion is classified into five levels: *jam*, *heavy*, *medium*, *mild*, and *low*. *Jam* is the situation that the vehicles fully occupy the roadways and almost all of the vehicles move slowly or completely stop. *Heavy* indicates that most of the vehicles on the roadway run slowly but seldom stop. In *medium* level, the difference from aforementioned levels is that all vehicles can move smoothly, and there still are many vehicles moving on the roadway. *Mild* means that the number of vehicles is much less than that in medium level and the vehicles move at normal speed. *Low* denotes that only few vehicles pass through roadway. Figure 1 shows the examples of video frames in five congestion levels from surveillance videos.

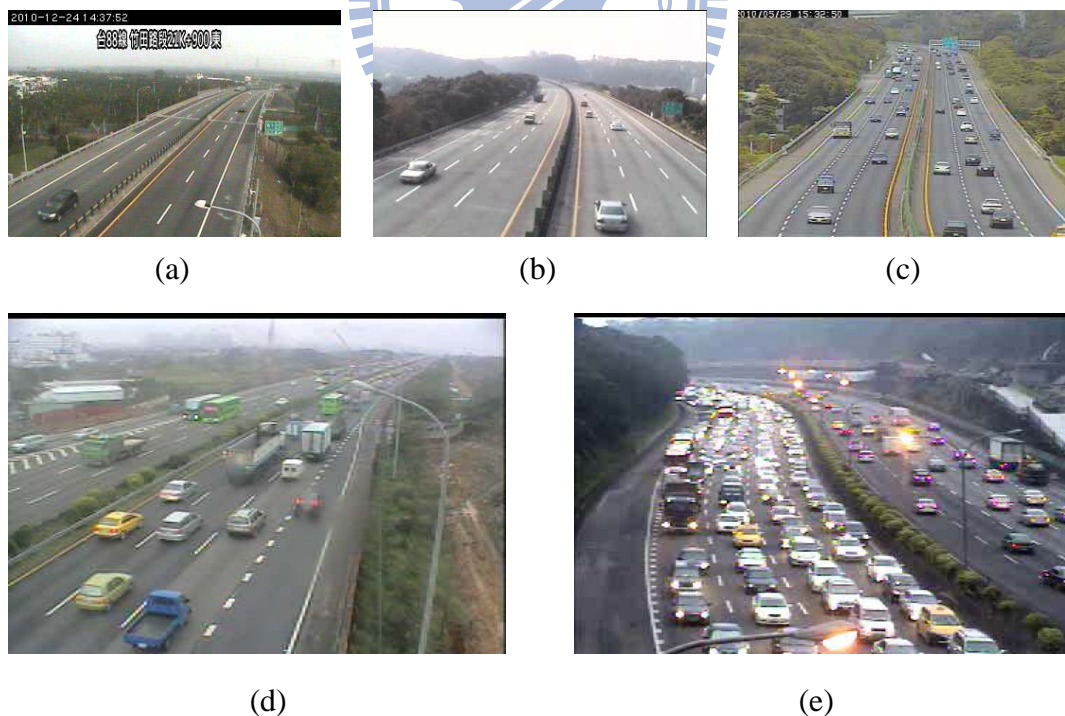
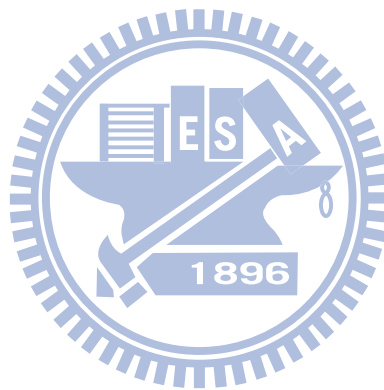


Figure 1. Captured frames in five congestion levels. (a) Low. (b) Mild. (c) Medium. (d) Heavy in the right side of roadway. (e) Jam in the left side of roadway.

In order to analyze the traffic information from video, image processing techniques and knowledge of classification are essential. In general, procedure of analyzing video comprises selection of the region of interest, vehicle detection, vehicle tracking, and activity analysis. Nevertheless, in most of the existing works, a fundamental problem is that the performance of video analysis may not be stable with the varied environments. For example, a large number of vehicles may lead vehicle occlusion and cause the failures of vehicle segmentation. Moreover, vehicles which have similar features such as color, shape, texture, and moving direction increase the difficulty in vehicle tracking. On the other hand, a critical issue is that image processing is always time-consuming. For the requirement of real-time response, developing efficient frameworks and algorithms of video analysis is an important and inevitable challenge. Consequently, how to quickly and accurately evaluate the traffic congestion from traffic surveillance videos is the core problem in our work.

In this thesis, we propose a real-time traffic congestion classification framework which consists of daytime and nighttime modules to automatically process the daytime and nighttime surveillance videos for identifying the traffic congestion levels. For daytime surveillance videos, the moving vehicles on the roadway are extracted by background subtraction and shadow elimination technique. Afterward the extracted vehicles are used to calculate the important traffic parameters including traffic flow, traffic speed, and traffic density. Then the traffic parameters are utilized to evaluate and classify traffic congestion levels. For nighttime surveillance videos, the moving vehicles are detected by extracting and grouping the headlight candidates. Subsequently a virtual detection line is utilized for evaluating the traffic congestion levels. Finally, we examine the proposed framework on real freeway surveillance videos captured at day and night data to demonstrate the accuracy and real-time response of traffic congestion classification.

The rest of this paper is organized as follow. Some related works on daytime and nighttime video processing are reviewed in Chapter 2. In Chapter 3, we introduce the proposed framework of traffic congestion classification. After that, we present the daytime module that includes initialization procedure, vehicle detection and traffic congestion classification in Chapter 4. In Chapter 5, the module of nighttime traffic congestion classification which is composed of headlight extraction, vehicle detection and traffic congestion classification is described. The experimental results of daytime module and nighttime module are shown and discussed in Chapter 6. In Chapter 7, we conclude this thesis and discuss the future works.



Chapter 2. Related Work

In this chapter, we review the previous research works on daytime surveillance and nighttime surveillance video processing. The details are described as follows.

2.1 Related work on Daytime Surveillance

In the past, plenty of works related to daytime traffic surveillance had been proposed [1]. In the following sections, some methods of roadway detection, object detection, shadow elimination, and traffic surveillance video analysis are introduced.

2.1.1 Roadway Detection

In traffic surveillance videos, roadway is the only region that we are interested in, and the rest regions in video frames are worthless. Therefore, finding out the region in advance reduces the computation of video processing and decreases errors caused by moving objects outside the roadway. In addition, discovering a center line of bidirectional roadway is helpful to monitor respective traffic flows in two directions.

Li and Chen [2] propose an algorithm to detect the lane boundaries of roadway by using Multi-resolution Hough Transform [3] without a priori knowledge of road geometry or training data. Then the region between the lane boundaries is regarded as roadway region. Lai et al. [4] put forward a method to detect multiple lanes from a traffic scene by using lane marking information and orientation. However, there are many types of lane markings like solid line, double solid line, and dotted line on different kinds of roadway as shown in Figure 2. Therefore, finding the correct lane



Figure 2. Lane markings with different shapes and colors on the roadway.

markings is not a simple task. Furthermore, the lane markings of the roadway are not always visible in some cases. Hence, there are other researches on roadway detection without lane marking information.

Stewart et al. [5] present an automatic lane finding algorithm based on detecting a region with significant changes. The roadway region in a traffic scene is generated by accumulating the differences between two consecutive frames after removing noises and sudden changes in brightness. However, a limitation of their algorithm is that the roadway must be parallel to the direction of camera shooting. Afterwards, Pumrin and Dailey [6] improve the algorithm to detect the roadway region from a variety of camera angles. For generating a roadway region mask, a hundred frames of moving edge images are accumulated and then holes are filled with a convex hull algorithm. In addition, two successive activity region masks generated from two successive sets of hundred frames are compared to detect the camera's motion. In [7], Lee and Ran put forward a method to detect bidirectional roadway by accumulating moving parts in a difference image between two consecutive frames and find a center line to separate the roadway into two parts with different directions. Nevertheless, their methods are affected by unbalanced traffic flow in different lanes and constrained by three roadway types. As shown in Figure 3, the roadway may extend to a) bottom-right, b) bottom-mid, c) bottom-left. Moreover, a clear gap must exist between the two respective parts in two directions for center line estimation.

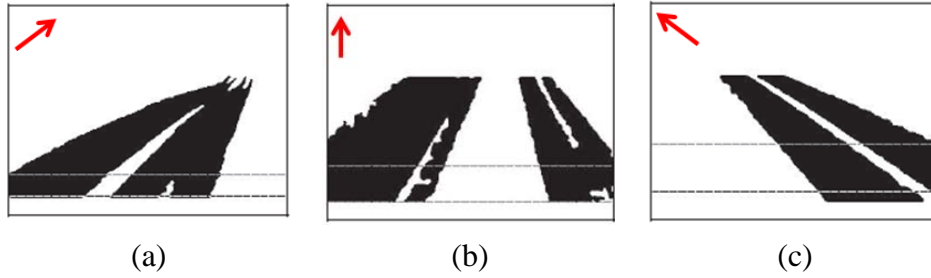


Figure 3. Three roadway types. (a) Bottom-right. (b) Bottom-mid. (c) Bottom-left.

Therefore, in order to conquer the drawbacks in previous works, the roadway detection and bidirectional roadway analysis without any lane marking information and a limitation of specific roadway types are proposed in our framework.

2.1.2 Object Detection

Detecting the moving objects is an important and useful technique for video understanding. Thus, many techniques are developed and can be classified into four categories: background subtraction, segmentation, pointer detectors and supervised learning [8]. Among the four categories, background subtraction is a widely used method for detecting moving objects in videos captured by static cameras. The rationale of the method is to detect the moving objects from the significant differences between the current frame and a reference frame, often called “background image” or “background model”. However, this method suffers from the background varying. Thus, the background image must be a representation of the scene without moving objects and keeps regularly updated so as to adapt to the changing geometry setting and luminance condition [9]. Since roadway surface is even and smooth in most cases and traffic surveillance is stable without camera motion, we employ background subtraction technique for the advantages of integrity of foreground and low computation complexity in extracting the moving vehicles on the roadway.

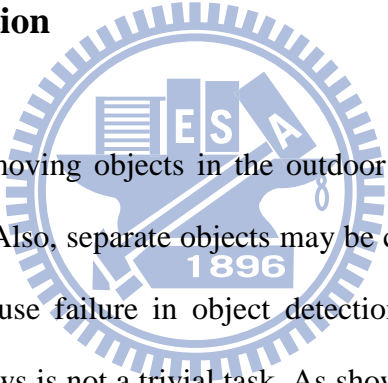
A variety of algorithms and techniques for performing background subtraction have been developed to detect vehicles. Averaging [10] and finding the median values [11] of a sequence of frames are the most basic ways to construct the background image in the past. Subsequently, Chen et al. [12] put forward a background image construction by calculating the frequency of pixel intensity value at training period. The frequency ratios of intensity values for each pixel at same position in frames are calculated and the intensity values with biggest ratio are incorporated to model a background image. Then, the background image is updated by repeating initialization operation. However, the aforementioned methods are fast but memory consuming and do not provide explicit methods to choose a threshold for segmenting out the foreground. Hence, Wren et al. [13] propose a running Gaussian average background model which fits a Gaussian probability density function on the latest n values of a pixel location for each pixel in a frame. In addition to the low memory requirement, that the threshold is determined automatically by standard deviation is the most significant improvement.

In some conditions, different objects are likely to appear at a same location over time. Therefore, some researches are proposed to deal with multiple modal background distributions. Stauffer and Grimson [14] raise a case for a multi-valued background model that is capable of coping with multiple background objects. The recent history of each pixel, called pixel process, is modeled by a mixture of K Gaussian distributions, and each pixel is classified into foreground or background according to whether the pixel matches one distribution of its pixel process. In a highly volatile environment, Elgammal et al. [15] propose a method to model the background by a non-parametric model based on kernel density estimation on the last n values. The method rapidly forgets the past and concentrates on recent observation and is more accurate because of avoiding the inevitable errors in parameter estimation,

which requires great amount of data to be accurate and unbiased.

On the other hand, traditional background subtraction approaches model only temporal variation of each pixel. However, there is also spatial variation in real world due to dynamic background such as waving trees and camera jitters, which causes a significant performance degradation of traditional methods. A novel spatial-temporal nonparametric background subtraction approach is proposed by Zhang et al. [16] to handle dynamic background by modeling the spatial and temporal variations at the same time. Moreover, other various background subtraction methods suitable for different environments have been reviewed and discussed in [1, 16, 17].

2.1.3 Shadow Elimination



When we detect the moving objects in the outdoor images, shadows are often extracted with the objects. Also, separate objects may be connected through shadows. Both conditions always cause failure in object detection. However, separating the moving objects from shadows is not a trivial task. As shown in Figure 4, shadows can be generally categorized cast shadow and self shadow [18]. Referring to Figure 4, the self shadow is a portion of the object not illuminated by the light source. The cast shadow lying beside the object does not belong to the object. For object detection and many applications, cast shadow is undesired and should be eliminated, while self shadow is a part of the object and need to be preserved. However, cast shadow and self shadow are similar in intensity. Thus, how to distinguish between them becomes a serious challenge. Moreover, if an object has intensity close to its shadows, shadow elimination is extremely difficult. Sometimes even though object and shadows can be separated, object shape is often incomplete due to imprecise shadow removal.



Figure 4. Cast Shadow and self shadow.[18]

Confronting these knotty problems, various methods for shadow elimination have been proposed for suppression of cast shadow in recent years. The intensity, color and texture are the most remarkable features of shadow. Because the distribution of intensity within a shadow is not uniform in real environments, Wang et al. [18] develop a method to estimate attributes of shadow by sampling points on edges of cast shadow and remove the shadow by the attributes. Afterwards, a process is executed to recover the object shape on the basis of information of object edges and attributes of shadow for avoiding over-elimination. Song et al. [19] remove the shadow in good use of the different properties between shadows and objects based on the RGB chroma model. Liu [20] introduces a method which uses gradient feature to eliminate shadow based on the observation that shadow region presents the same textual characteristics as in the corresponding background image.

Based on the prior knowledge, Yoneyama et al. [21] simplify 3D solid cuboids model to a 2D joint vehicle-shadow model for eliminating cast shadow. Six types of vehicle-shadow models are employed to match the extracted vehicle by utilizing luminance of shadow for differentiating the vehicle and shadow. Besides, Chien et al. [22] remove the shadow by a mathematical analysis model. Different from the methods mentioned above, Hsieh et al. [23] use lane geometries as an important cue to help eliminate all undesirable shadows even though the intensity, color and texture of vehicles are similar to cast shadow.

A summary of general observations with respect to cast shadow and background in roadway scene is given by Xie et al. [24] : (1) Pixels of cast shadow fall on the same surface as the background; (2) Cast shadow pixels are darker than their background in all three color channels; (3) Background is mostly roadway surface which is often monochrome in traffic scene. As a result, the values of hue channel are small in the cast shadow region; (4) The edge pixels of the cast shadow are significantly less than that of the vehicle.

2.1.4 Traffic Surveillance Video Analysis

For analyzing the content of video, the trajectories of moving objects can provide much information, and object tracking is an important and unavoidable way to extract the trajectories. Take traffic surveillance videos for example, if we intend to realize the action of moving vehicles, we have to analyze how the vehicles move. That is, we must track vehicles during the traffic monitoring in order to obtain their trajectories. Generally, tracking methods can be classified into two categories. One category estimates motions of moving objects and minimizes the error function to track objects. Another category calculates the similarities between current objects and previous objects and maximizes the similarity measures to track the objects. A variety of object tracking methods developed in past decades are reviewed in [8].

In traffic surveillance videos, a traditional approach of moving vehicles tracking is to model the moving object properties such as position, velocity and acceleration. Measurements usually include the object positions in the frame, which is obtained by an object detection algorithm. In particular, Kalman filter [25] and particle filter [26] are popularly used in many research works. However, in practical applications, it is difficult to track all vehicles on the roadway. For examples, if the viewpoint of a

camera is low or there are plenty of vehicles on the roadway, the vehicle occlusion problem results in failing to extract and track correct individual vehicles. Moreover, that the effective resolution of perspective is reduced in a frame makes insufficient features of vehicles for object tracking. Even though many researches on occlusion problem have been proposed, the complexity of vehicle tracking is surged and accuracy of vehicle tracking is sagged while a great number of occluded vehicles need to be tracked at one time.

Along with the trajectories of vehicles are extracted, numerous works deal with vehicle activity analysis. The features such as size, speed, and moving direction of vehicles are helpful to understand the situation of traffic flow in the surveillance videos [2]. Generally, more complicated events are mostly detected with machine learning algorithms. In [27], the issue of event detection in time series data is addressed using neural network. In [28], Hidden Markov Models are used to form the basis of activity recognition and anomaly detection.

For the purpose of traffic flow analysis, the usage of virtual line detectors without tracking all vehicles on the roadway was developed in [29-33]. In [29], the authors present an approach to evaluate traffic-flow parameters under urban road environment. Virtual line based time-spatial image as shown in Figure 5 is used for vehicle counting. The vehicles are extracted from the time-spatial image after edge detection and morphological operation. In [30], the time-spatial image is processed to evaluate the traffic congestion level. However, these methods do not work well in the frames with low contrast, small vehicle blocks, and irregular driving behaviors. Afterwards, the virtual line group methods are proposed as an improvement in [32, 33]. Actually, virtual line based algorithms are more suitable to analyze the traffic flow while there are a large number of vehicles for real-time performance.

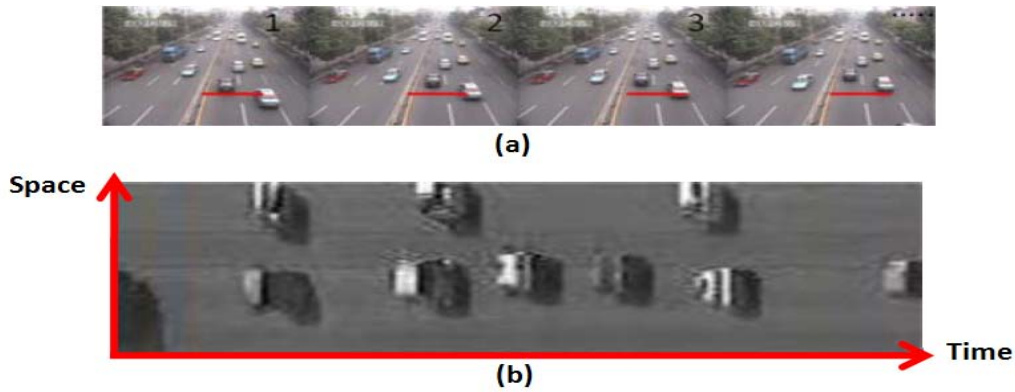


Figure 5. Generation procedure of time-spatial image.[29] (a) A frame sequence. (b) Time-spatial image generated by virtual line iteration.

2.2 Related Work on Nighttime Surveillance

Due to low illumination, understanding the activities in nighttime videos becomes more difficult than in daytime for the frames captured from a camera have lower contrast and higher noise than their corresponding daytime frames. Thus, nighttime video analysis is still quite a challenging task up to the present. In this section, we will review some studies on nighttime surveillance video processing.

2.2.1 Nighttime Image Enhancement

In order to solve the low contrast problem, some researches focus on nighttime image enhancement. Histogram equalization is a commonly used method for image enhancement in luminance of image. Hence, Sayed et al. [34] propose an efficient algorithm that modifies traditional histogram equalization to maintain the color information of the original nighttime image. Each color channel is enhanced separately by multiplying the ratio of enhanced luminance to original luminance.

Nevertheless, the performance of enhancement is limited due to the detailed information of nighttime frames has been lost. Therefore, Cai et al. [35] combine

daytime image and nighttime image together based on the object extraction technique. The low quality static parts of a nighttime image can be replaced by the high quality counterpart in the daytime image. However, if errors occur in object extraction, unnatural mixture images may be generated. In [36], authors can produce natural-appearing enhanced images that do not appear to be fake. The image is decomposed into luminance and reflectance components, and only the luminance of the image is modified by referring to daytime background.

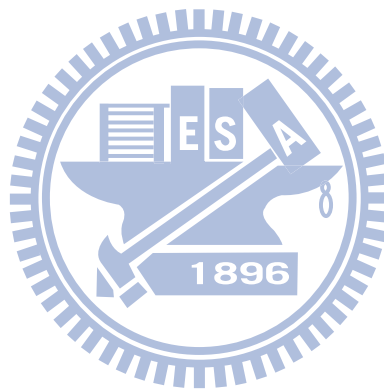
2.2.2 Object Detection and Tracking

As mentioned in previous section, due to the low contrast problem in nighttime video, detecting moving objects from the dark scenes becomes difficult. Even though the nighttime image enhancement technique improves appearance of nighttime images, object detection based on background subtraction is quite arduous. Hence, in [37], the authors put forward an algorithm that is based on contrast analysis to detect moving objects. They use the local contrast change over time to detect potential moving objects, which is called Salient Contrast Change (SCC). Then motion prediction and spatial nearest neighbor data association are used to suppress false alarm. Wang et al. [38] propose a model based on SCC feature which applies learning process to strength adaptability and analyze trajectories to improve the effectiveness of detection.

For nighttime traffic surveillance, Kostia Robert [39] presents a framework to detect multiple vehicles at night by headlight detection. He adopts HSV color model and uses the ratio of value to saturation (V/S) with white top-hat transform operation to obtain the bright blobs as headlights. Then the blobs are analyzed to generate hypothesis of vehicles. In [40], in order to extract headlights, a bright object

segmentation process based on automatic multilevel histogram thresholding is applied on the nighttime scenes of roadway.

In the case of object tracking, in general, conventional tracking methods such as model-based, appearance-based, and feature-based cannot work well at night due to insufficient detailed information. Thus, an appropriate solution is to track vehicles by using position and velocity. In [39], the vehicles are tracked over frames by using a Kalman filter associated with a reasoning module. Also the tracking processing in [37, 40] are based on these features.



Chapter 3. Proposed Framework Overview

In this chapter, we make an introduction of the proposed traffic congestion classification framework. The framework, which consists of daytime module and nighttime module, is able to classify the traffic congestion in traffic surveillance videos captured during day and night into five levels.

As shown in Figure 6, in the framework, a video clip is firstly determined whether it is captured at day or at night, and the corresponding modules are applied on the video according to the capturing time. The brief descriptions of the two modules are described in the following sections. As for the details of each component in the modules, we are going to expound them in Chapter 4 and Chapter 5.

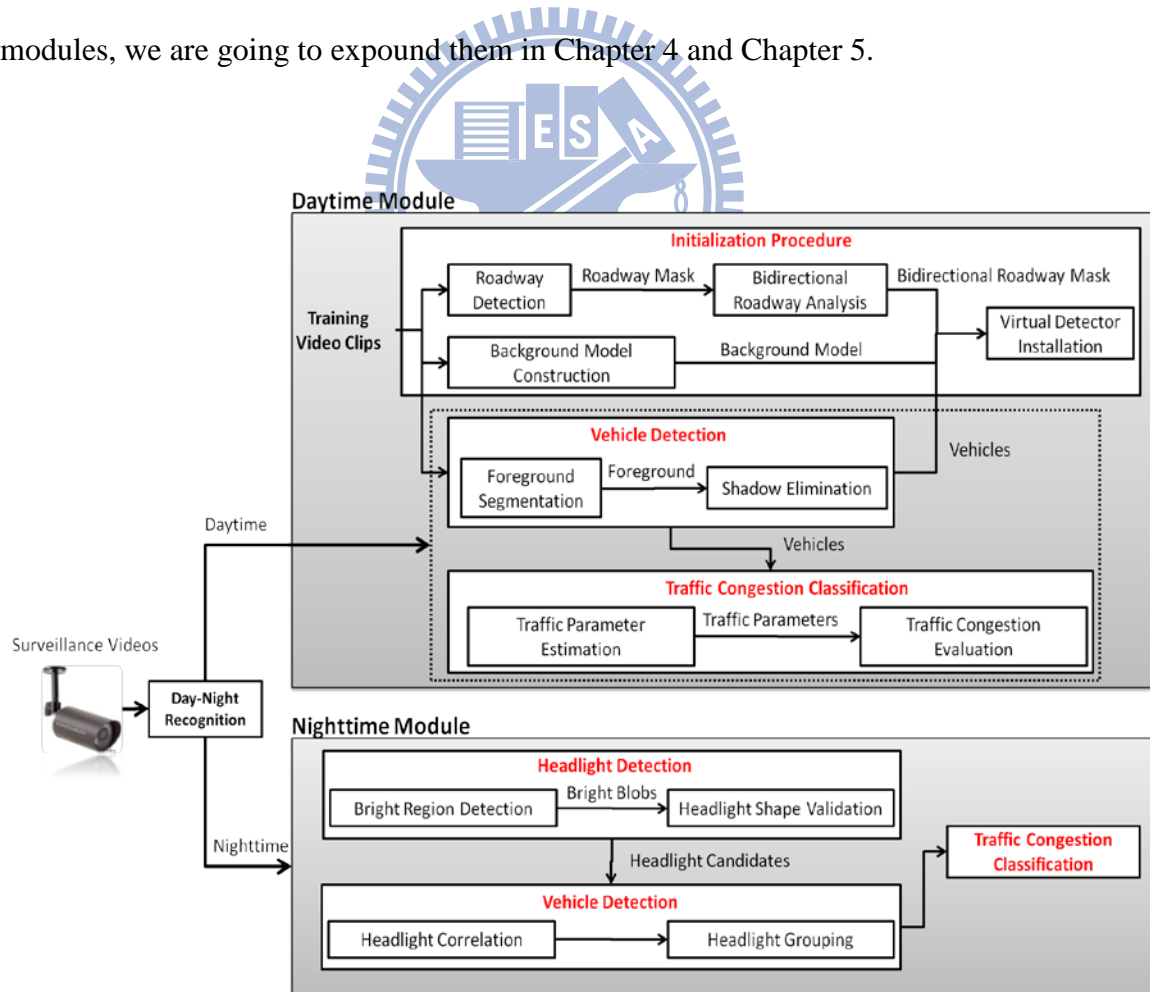


Figure 6. Overview of the proposed framework.

3.1 Daytime Traffic Congestion Classification

The goal of the daytime module is to automatically analyze the level of traffic congestion in daytime surveillance videos. In general, vehicle detection and tracking are inevitable ways to realize the content of traffic surveillance videos. However, some troublesome factors such as a great number of moving vehicles with serious vehicle occlusion not only increase the difficulty in detecting and tracking the moving vehicles, but also lower the efficiency of video processing. Therefore, in order to understand the traffic condition, we adopt a novel strategy that uses virtual detectors to simplify the processes of vehicle detection and tracking. Based on this strategy, it is unnecessary to extract and track all moving vehicles on the roadway for analyzing the surveillance videos. Furthermore, we also put forward two methods to detect the roadway region and distinguish the moving direction of roadway to enhance efficiency and robustness of the module. As shown in Figure 6, the module contains three major parts: initialization procedure, vehicle detection, and traffic congestion classification.

Initialization procedure is performed at the beginning of traffic monitoring in order to obtain the consistent characteristics of roadway including the region and moving direction of roadway, positions of virtual detectors and background model. As Figure 6 illustrates, we first extract the roadway region because only monitoring the area that we are interested in is helpful for the efficiency and accuracy enhancement. Second, bidirectional roadway analysis is applied on the roadway region in order to monitor both directions of roadway at the same time. Moreover, the background model is also constructed along with the above processes, and the model is updated with incoming frames over time. Finally, virtual detectors are automatically installed on each lane of roadway for gathering traffic flow information. In addition, since

initialization procedure executes only once in advance, there is no requirement of strict real-time response.

Vehicle detection is to extract the moving vehicles from videos. We use background model to segment the foreground image and eliminate the shadow by using two characteristics of shadow: edge and color reflectance. Next, connected component analysis clusters all adjacent pixels in foreground image as isolated moving vehicles. Because the virtual detectors are adopted to gather traffic information in the module, only the moving vehicles that pass through the virtual detectors are need to be detected and tracked. In this way, the efficiency of video processing is promoted a lot.

In traffic congestion classification, the virtual detectors estimate three traffic parameters: traffic flow, traffic speed and traffic density by tracking the moving vehicles that pass through the detectors. For a sequence of moving vehicles over a period of time, the parameters are calculated and utilized to evaluate traffic congestion degree. Finally, we use classifier to make an accurate classification of traffic congestion levels. With bidirectional analysis, the proposed module is capable of analyzing both directions of roadway at the same time.

3.2 Nighttime Traffic Congestion Classification

The proposed night module is able to classify the congestion level of traffic flow in nighttime surveillance videos. In order to understand the content of traffic surveillance videos, in general, vehicle detection is usually the first step. However, some tough factors such as poor visibility and higher noise increase difficulty in detecting the moving vehicles under nighttime condition. Especially, typical daytime surveillance framework based on background subtraction cannot work at night due to the low contrast foreground objects against the background, which is an obstacle to

vehicle detection at night. On the other hand, less color and texture information may lower the ability of vehicle tracking to distinguish between different moving vehicles.

To overcome these limitations, an appropriate solution for vehicle detection is to find headlight that is a salient feature to all vehicles at night. Hence, in the nighttime module, headlight detection and grouping technique are developed to extract the moving vehicles on the roadway. In addition, we abandon vehicle tracking approaches for analyzing traffic congestion due to the reasons below. Headlights of all vehicles are similar to each other. The large number of vehicles in crowded situation causes high complexity in vehicle tracking. Therefore, a virtual detection line is utilized to evaluate traffic congestion degree for avoiding errors caused by vehicle tracking and enhancing the execution efficiency.

The module includes three stages: headlight extraction, vehicle detection, and traffic congestion classification. Headlight extraction is performed at the beginning of each frame processing to recognize the circle-shaped bright blobs that is most similar to headlight. At vehicle detection stage, correlation of two headlights is calculated by using three features: width, height and edge relationships between them. After that, the values of correlation are used in our headlights grouping mechanism to detect the moving vehicles. For a sequence of video frames, traffic congestion is evaluated when the moving vehicles pass through a virtual detection line. A virtual detection line is a virtual horizontal line that crosses a roadway and is employed to gather the traffic information. In other words, only the headlights that touch the virtual detection line need to be extracted and grouped. Then, the possible mistakes from headlights grouping are decreased and the efficiency for video processing is raised.

Chapter 4. Daytime Traffic Congestion Classification

In this chapter, we describe the daytime traffic congestion classification module. The module contains initialization procedure, vehicle detection and traffic congestion classification as presented in the following sections.

4.1 Initialization Procedure

Initialization procedure is executed at the beginning of traffic monitoring and utilizes training video data to obtain the consistent characteristics of roadway including the region and moving direction of roadway, positions of virtual detectors and background model. In this section, we present our approaches of roadway detection, bidirectional roadway analysis and virtual detector installation in details. To make it easier to understand the proposed approaches, a traffic scene is used to be an example for demonstration as shown in Figure 7.



Figure 7. A traffic scene of a bidirectional roadway consisting of six lanes.

4.1.1 Roadway Detection

Some peculiar phenomena exist in the traffic surveillance videos. For example, shaking trees outside the roadway causes frequent motions over time. We realize two important facts by observation. First, all moving pixels outside the roadway must not belong to any vehicles. Second, surface of roadway is much stable than other regions outside the roadway. In other words, most moving pixels on roadway are a part of vehicles and actually need to process. Hence, an efficient way to remove most of useless pixels is to detect a region of roadway in advance, and then we can focus the further video processing only on the roadway region. This approach not only avoids unnecessary mistakes, but also reduces the time-consuming video processing.

In our framework, we develop a process to accomplish the roadway detection based on the concept that roadway is a region where most movements occur because of vehicle motion. Therefore, we calculate the differences between two consecutive frames to detect the movements and accumulate the movements to obtain the position and shape of roadway. The difference image D is defined as follows:

$$D^t(x, y) = D^{t-1}(x, y) + \begin{cases} 1, & \text{if } |F^t(x, y) - F^{t-1}(x, y)| > th_M \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $F^t(x, y)$ denotes the current frame t and th_M is the pre-defined threshold for identifying the pixels with movements. After accumulation of movements for a sequence of frames, the pixels that have a value in difference image D larger than 0 are regarded as a roadway candidate R_c for the roadway region. The result of the example is shown as Figure 8(a) and the white pixels are the pixels of R_c .

$$R_c(x, y) = \begin{cases} 1, & \text{if } D(x, y) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Then, we discover that some fragmentary movements outside the roadway are also detected because of the shaking trees. In order to eliminate the fragmentary movements, the pixels which satisfy the following noise filtering equation are removed from R_c :

$$\frac{1}{(2p+1)^2} \sum_{i=x-p}^{x+p} \sum_{j=y-p}^{y+p} R_c(i, j) < th_R \quad (3)$$

where p control the size of filter and th_R is a threshold determined by the characteristic of roadway. Afterward, we employ connected component analysis on R_c as shown Figure 8(b) to extract all isolated connected areas and sort the isolated connected areas in descending order according to their size from S_1 to S_n , where n is the number of isolated connected areas. Then choose the first L largest areas as the roadway region.

$$L = \arg \min_r \sum_{i=1}^r \frac{S_i}{S} > T_R \quad (4)$$

where r is a value ranging from 1 to n , S is the summation of S_1 to S_n and T_R is a measure to maximize the number of areas that should be accounted for the roadway. This operation reserves the main areas of roadway as shown in Figure 8(c). Some small but not fragmentary noises are eliminated successfully. Finally, using Closing morphological operation to fill out the holes and a hole-filling algorithm to obtain the complete roadway mask R as shown in Figure 8(d).

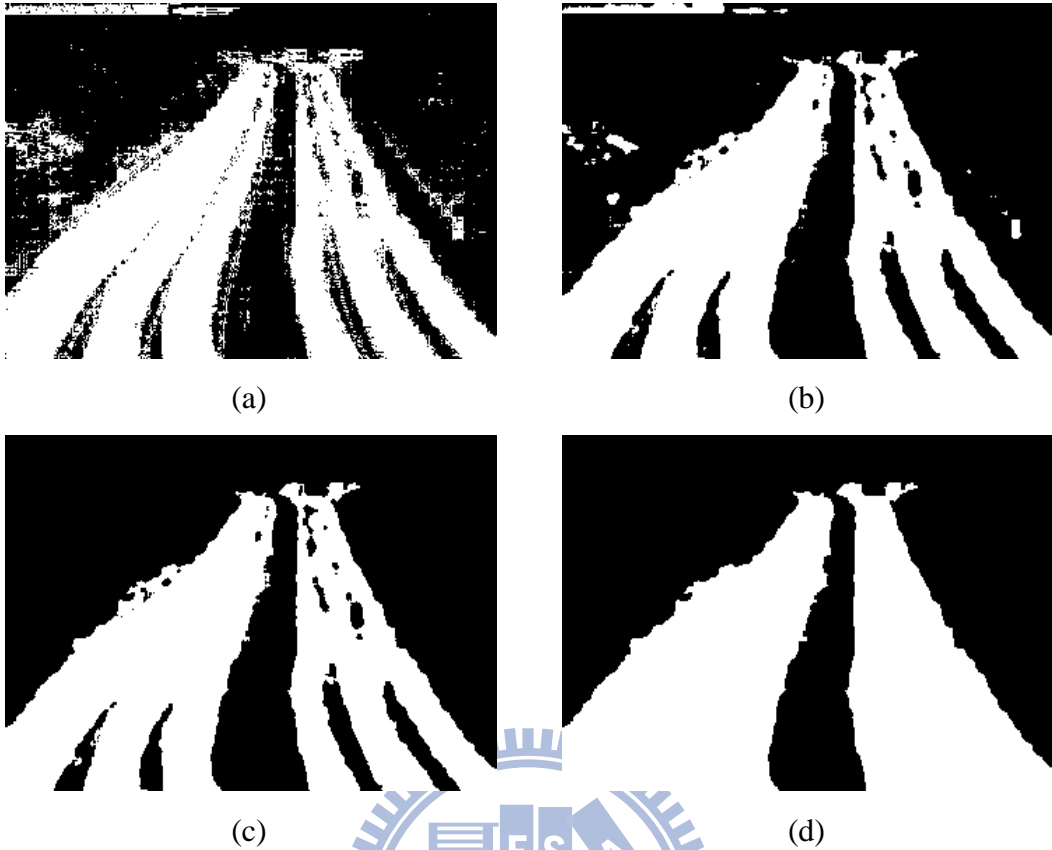


Figure 8. An example of roadway detection. (a) The candidate of roadway region R_c . (b) The roadway candidate R_c after noise filtering. (c) The first L largest areas. (d) Complete roadway mask R .

4.1.2 Bidirectional Roadway Analysis

Most surveillance cameras capture a video at a specified angle and range that contains multiple lanes of traffic in both directions. The bidirectional roadway analysis can be applied to monitor traffic flow for both directions individually. During the period of movement accumulation in roadway detection, the occurrence of movements means some motions appear at the same time. Accumulation of those motions is able to approximately reveal the moving direction of roadway. Hence, we estimate and accumulate the motion vectors for those pixels that have movements in roadway detection. A Motion Vector image MV is defined as follows:

$$MV^t(x, y) = MV^{t-1}(x, y) + \begin{cases} 1, & m_y^t(x, y) > 0 \\ -1, & m_y^t(x, y) < 0 \end{cases} \quad (5)$$

where $m_y^t(x, y)$ denotes the motion vector [41] in y -axis between frame t and $t-1$.

The method to estimate motion vector is described in Algorithm 4.1. Since bidirectional roadway contains two kinds of directions, DOWN and UP, in most cases, the motion vectors in y -axis are only considered. After accumulation of the motion vectors, the Motion Vector image is simplified into a Motion image M :

$$M(x, y) = \begin{cases} \text{DOWN}, & MV(x, y) > 0 \\ \text{UP}, & MV(x, y) < 0 \end{cases} \quad (6)$$

As shown in Figure 9(a), the white pixels denote the DOWN and the gray pixels denote UP. In order to separate the roadway into two respective parts in different directions, a center line of the roadway for separating the motions should be calculated. A motion classification method is used to obtain this center line. First, the positions of both directions need to be decided by the average x -position of two kinds of motions.

$$X_{DOWN} = \frac{1}{N_{DOWN}} \sum_{(x,y) \in R} x, \quad \text{if } M(x, y) = \text{DOWN} \quad (7)$$

$$X_{UP} = \frac{1}{N_{UP}} \sum_{(x,y) \in R} x, \quad \text{if } M(x, y) = \text{UP} \quad (8)$$

where N_{DOWN} and N_{UP} are the number of pixels of motion DOWN and UP in motion image, respectively. If X_{DOWN} is smaller than X_{UP} , the moving direction of the left side of roadway is DOWN and the right side is UP. Otherwise, the left side is UP and the right side is DOWN.

Then, a center line CL for separating bidirectional roadway is evaluated by rotating a separator line SL . As illustrated in Figure 9(b), each pixel at the top

boundary of roadway region is regarded as a core for SL rotation, and the SL rotates from left boundary to right boundary of roadway mask for each core. For each rotation, we evaluate the separation degree SD of motion classification that bases on the SL . The smaller the SD is, the better the motion classification is. The SL with minimum SD is the actual center line CL of the roadway.

In the SD evaluation, the ratios of error and recall for motion classification are considered at the same time. From the result in the first step, we can know what the well-classified pixels are and what the mis-classified pixels are. For instance, if the left side of roadway is DOWN ($X_{DOWN} < X_{UP}$), the pixels of motion DOWN in the left side of SL are well-classified and the pixels of motion UP in the left side of SL are mis-classified. Therefore, the error ratio of left side e_{left} is defined as the ratio between the number of motion UP pixels and the number of total pixels in the left side of SL . The recall ratio of left side r_{left} is defined as the ratio between the number of motion DOWN pixels and the number of total DOWN pixels in whole roadway mask. Analogously, the rationale is the same for the right side. Separation degree SD is defined as follow:

$$SD = \frac{e_{left} + e_{right}}{r_{left} + r_{right}} \quad (9)$$

where e_{left} is the error ratio for classification in the left side of SL , e_{right} is the error ratio for classification in the right side of SL , r_{left} is the recall ratio for classification in the left side of SL , and r_{right} is the recall ratio for classification in the right side of SL . According to the center line CL , the roadway mask is divided into two parts as a bidirectional roadway mask. An example of the center line evaluation and bidirectional roadway mask are shown in Figure 9(c) and Figure 9(d).

Algorithm 4.1: Motion Vector Estimation [41]

Input: The position (x, y) of a pixel, current frame F^t , and previous frame F^{t-1}

Output: The motion vector in y-axis v for the pixel (x, y)

```
1  for  $i := -p$  to  $p$  do
2      for  $j := -q$  to  $q$  do
3           $diff = |F^t(x, y) - F^{t-1}(x+i, y+j)|$ 
4          if  $diff < mini\_diff$  then
5               $mini\_diff = diff$ ;
6               $v = j$ ;
7          end for
8  end for
```

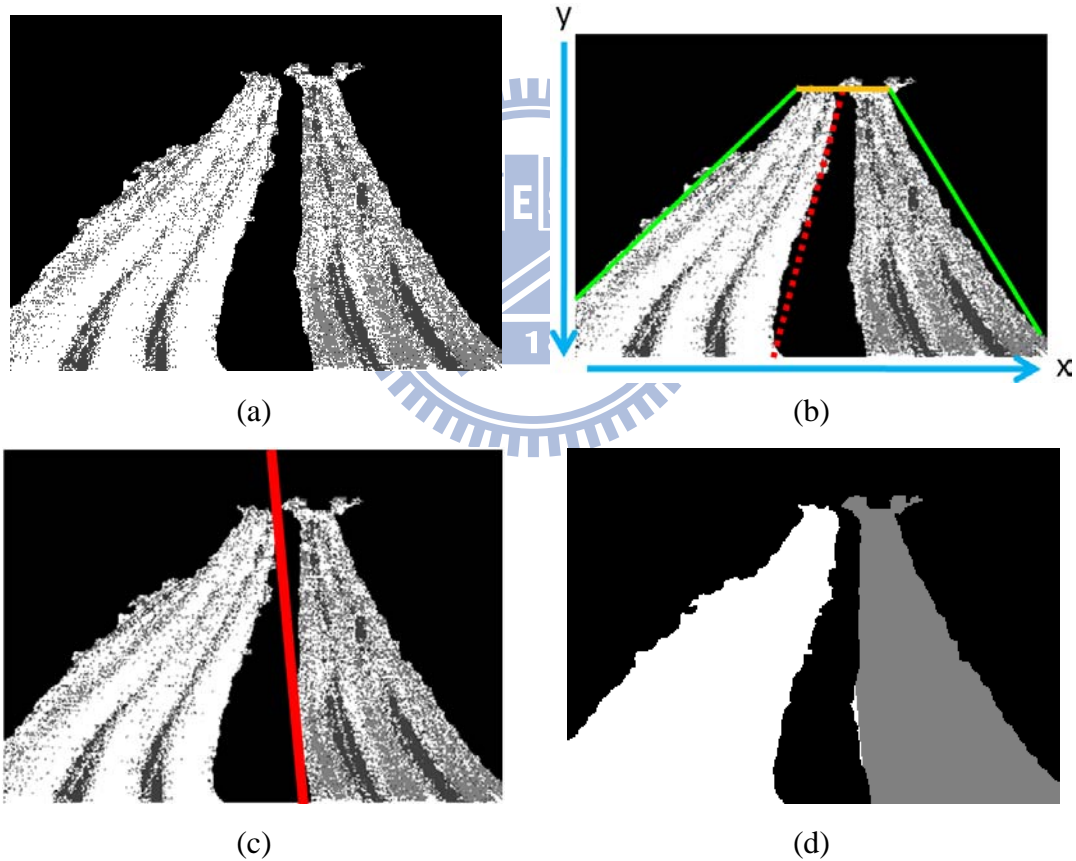


Figure 9. An example of bidirectional roadway analysis. (a) Motion image M , the motion of white pixels is DOWN, the motion of gray pixels is UP. (b) The separator line SL which is represented by a red dotted line rotates between two boundaries to calculate the separation degree SD for motion classification. (c) The center line CL which has the minimum SD . (d) Bidirectional roadway mask.

4.1.3 Virtual Detector Installation

As discussed in section 2.1.4, we realize that vehicle tracking technique is not an appropriate way to analyze actions of all vehicles on roadway due to the vehicle occlusion and reduction of effective resolution. Hence, the virtual line based algorithms are the better approaches to analyze the traffic situation for its real-time response and low difficulty in vehicle tracking.

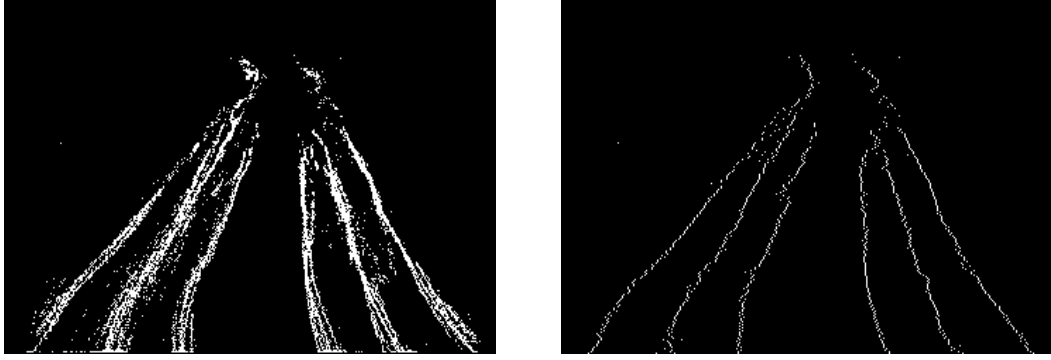
In order to overcome the disadvantages of previous works, a method adopting virtual detectors as shown in Figure 11 is proposed in our framework. The virtual detectors are set up on each lane for traffic information collection. The appropriate positions for the detectors are the central point between two lane markings. However, if the type of lane marking is a dashed-like line or the lane marking is not visible, the lane markings always fail to be detected. Based on our observations, we know the lane center usually is the central point of a moving vehicle. So the central points of the vehicles are retrieved from a clip of video to detect the lane centers.

To obtain the central points of the moving vehicles, vehicle detection based on background subtraction technique, as described in section 4.2, is applied in this step. Every extracted vehicle is identified by a bounding box after detection. The central points of vehicles and the average width of vehicles are gathered from the bottom line of bounding box. Figure 10(a) shows an example of the central points of moving vehicles over a period of time. After collecting all central points of vehicles, Modified Basic Sequential Algorithm Scheme (MBSAS) [42] clustering algorithm, as described in Algorithm 4.2, is used to cluster the x-coordinate of central points in at every row of a video frame, and the average width of vehicles is chosen as the threshold θ for MBSAS. The result of the example is shown in Figure 10(b). Afterward, each center of

clusters is considered as the center of each lane. In other words, the actual positions of virtual detectors are determined by the cluster centers.

A detection row d in a frame is regarded as an expected row for virtual detector placement. To avoid a problem that the vehicles in training data is not dense enough, the $2 \times m$ neighboring cluster results of row d are simultaneously considered for determining the correct positions of virtual detectors. Let $num_cluster_d$ be the number of clusters of row d . From $num_cluster_{d-m}$ to $num_cluster_{d+m}$, the value with maximum count is the actual number of virtual detectors. That a row contains num_vt clusters and is closest to the row d is chosen as the actual detection row d' . Finally, we set the virtual detectors on each cluster centers at row d' and identify their monitoring direction of traffic flow according to the center line CL . The result of example is illustrated in Figure 11. The red and blue rectangles are the virtual detectors. The different color means different monitoring directions.

Moreover, choosing an appropriate detection row d is dependent on roadway location and curvature with respect to the camera capturing perspective. In our framework, the detection row can be selected automatically or manually. The principle of choosing the detection row is to find an area where there is much less vehicle occlusion and the important features of vehicle are visible as much as possible. Hence, the position of detection row should be always at the bottom area of a frame. Nevertheless, it would incur the problem of incomplete vehicles if the position is too close to the bottom. Therefore, two-thirds frame height from the top of a frame is a suitable position empirically.



(a)

(b)

Figure 10. MBSAS processing on the history of the central points of vehicles. (a) All central points of moving vehicles in a period of time. (b) Cluster centers calculated by central points of moving vehicles at each row of a frame.

Algorithm 4.2: Modified Basic Sequential Algorithm Scheme (MBSAS)

Input: N patterns from x_1 to x_N and a maximum number of clusters: q

Output: clusters of patterns x_1 to x_N

Cluster Determination

```

1   $m = 1$            //  $m$  is the number of clusters
2   $C_m = \{x_1\}$      //  $C_m$  is the  $m$ th cluster
3  for  $i = 2$  to  $N$  do
4      Find  $C_k$ :  $d(x_i, C_k) = \min_{1 \leq j < m} d(x_i, C_j)$  //  $d(x_i, C_j)$  is the distance between
5          if  $d(x_i, C_k) > \theta$  AND  $m < q$  then // pattern  $x_i$  and cluster  $C_j$ 
6               $m = m + 1$ ;
7               $C_m = \{x_i\}$ ;
8          end if
9  end for

```

Pattern Classification

```

1  for  $i = 1$  to  $N$ 
2      if  $x_i$  has not been assigned to a cluster, then
3          Find  $C_k$ :  $d(x_i, C_k) = \min_{1 \leq j < m} d(x_i, C_j)$ 
4           $C_k = C_k \cup \{x_i\}$ 
5          Update the cluster center of  $C_k$ 
6      end if
7  end for

```



Figure 11. Results of virtual detectors installation on a bidirectional roadway.

4.2 Vehicle Detection

Vehicle detection is usually an important step to analyze traffic surveillance video. Since a characteristic of the roadway surface is stable, the background subtraction method is an appropriate way to segment the foreground image for the advantages of integrity of information and low computation. However, the shadow of vehicles is always detected with vehicles in outdoor scene. To solve this problem, the popular background subtraction model: Mixture of Gaussians and shadow elimination method based on gradient feature and color reflectance are adopted in our framework. The details of the methods are described in this section.

4.2.1 Mixture of Gaussians

Different background objects may appear at a same location in a frame over time. A representative example is that a traffic surveillance scene with trees and vehicles partially covering a roadway, then a same pixel location shows the values from tree leaves, vehicles, and the roadway itself. Thus, the background is not single modal in this case. So Stauffer and Grimson [14] propose a multi-valued background model to cope with multiple background objects.


They consider the values of a particular pixel location over time as a “pixel process”, $\{X_1, \dots, X_t\}$, which is modeled by a mixture of K Gaussians. The probability of the current pixel X_t value is

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (10)$$

where K is the number of Gaussian distributions and is determined by the various scenes in the different applications, $\omega_{i,t}$ is the weight of i th Gaussian at time t , $\mu_{i,t}$ is the mean value of the i th Gaussian at time t , $\Sigma_{i,t}$ is the covariance matrix of the i th Gaussian at time t , and η is a Gaussian probability density function:

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu)^T \Sigma^{-1} (X_t - \mu)} \quad (11)$$

In addition, for the reason of computation efficiency, the covariance matrix is assumed to be of the form:

$$\Sigma_{k,t} = \sigma_k^2 I \quad (12)$$


where k is an integer ranging from 1 to K . This means that the red, green, and blue pixel values are independent and have the same variances. Although this is not certainly the case, the assumption allows us to avoid the costly matrix inversion.

At each t frame time, a criterion is needed to provide discrimination between the foreground and background distributions. Therefore, each current pixel, X_t , in the frame is checked against the existing K Gaussian distributions until a match is found. Once the pixel matches one distribution of the existing K Gaussian distributions, which means the pixel belongs to the background. Otherwise, the pixel belongs to the foreground. A match is defined as follows:

$$(X_t - \mu_{i,t}) / \sigma_{i,t} < 2.5 \quad (13)$$

To solve the changes of geometry setting and luminance condition in most video sequences, it is necessary to track those changes of the K Gaussian distributions. In other words, the mixture of Gaussians background model has to be updated with new coming frames.

The authors implement an on-line K-means approximation instead of a costly expectation-maximization (EM) algorithm on a window of the recent data to estimate the updated model parameters. The weights of the k th Gaussian at time t , $w_{k,t}$, are adjusted as follows:

$$\bar{w}_{k,t} = (1 - \alpha) \bar{w}_{k,t-1} + \alpha (Mat_{k,t}) \quad (14)$$

$$Mat_{k,t} = \begin{cases} 1, & \text{the model is matched} \\ 0, & \text{the remaining models} \end{cases} \quad (15)$$

where α is the learning rate which determines the speed for the distribution's parameters updating. After this approximation, the weights should be normalized. In addition, the μ_t and σ_t parameters for unmatched distributions remain the same. The two parameters of a distribution which matches the current pixel X_t are updated as follows:

$$\mu_t = (1 - \rho) \mu_t + \rho X_t \quad (16)$$

$$\sigma_t^2 = (1 - \rho) \sigma_{t-1}^2 + \rho (X_t - \mu_t)^T (X_t - \mu_t) \quad (17)$$

where another learning rate, ρ , is

$$\rho = \alpha \eta(X_t | \mu_k, \sigma_k) \quad (18)$$

If none of the K distributions match current pixel X_t , the least probable distribution is replaced by a distribution which has the current pixel as its mean value μ , an initially high variance σ^2 and an initially low weight w .

While the parameters of the mixture model of each pixel change, we would like to determine which Gaussian distributions are most likely produced for the multi-modal background. To model this, a manner is required for deciding what parts of the mixture model best represents background. First, all the distributions are ranked based on the ratio between their weight, $w_{k,t}$, and standard deviation, $\sigma_{k,t}$. This assumes that the higher and more compact the distributions are likely to belong to the background. Then, the first B distributions in ranking order which satisfy

$$B = \arg \min_b \left(\sum_{i=1}^b \varpi_i > T_B \right), \quad (19)$$

where b is a value ranging from 1 to K , T_B is a measure of the minimum portion of the distributions that should be accounted for the background, are accepted as background.

4.2.2 Shadow Elimination

Shadow elimination is a critical issue to distinguish between the moving objects and the moving shadows for the robust vision-based systems. The shadow can cause various undesirable behaviors such as object shape distortion and object merging. To solve these problems, we combine the previous shadow removal works based on color reflectance [22] and gradient feature [20] to eliminate the cast shadow in our

framework.

First, we introduce a shadow elimination technique based on color reflectance. The principle of color reflectance can be modeled as the multiplication of light energy and reflectance of object and expressed by the following equation.

$$val^C = ener^C * refl^C \quad (20)$$

where C stands for color channels: red, green, and blue, val^C is the value of color C , $ener^C$ is the light energy of color C , and $refl^C$ is the reflectance of color C . From a relationship between shadow, and background, the following relationship would be obtained [43].

$$\frac{val_s^C}{bg_val^C} = 1 - \frac{Q}{bg_ener^C}, 0 \leq Q \leq bg_ener^C \quad (21)$$

where val_s^C is the value of color C in foreground shadow, bg_val^C is the value of color C in background, bg_ener^C is the light energy of color C in background. Thus, we realize that if a pixel belongs to shadow, the values must satisfy the following equation:

$$th_s \leq \frac{val^C}{bg_val^C} \leq 1, \quad (22)$$

where th_s is the threshold for identifying the shadow. Afterward, most of shadows would be removed from extracted object by the method. However, it is possible that some parts of vehicles are considered as shadow at the same time, which causes the broken vehicles. In general, morphological operation is a common approach to recover the broken vehicles, but it cannot recover those vehicles with serious damage. Thus, we use an approach to recover the broken vehicles based on gradient feature of the moving vehicles before morphological operation.

The approach to get the gradient feature of the moving vehicles is proposed in [20]. First, calculate gradient images of the moving vehicle and its relevant background. Gradient of the moving vehicle contains gradient of moving vehicles and its shadows. Moreover, gradient of relevant background contains gradient of only background. The example gradient images of moving foreground and relevant background are shown in Figure 12. Based on observation, we can discover the gradient of the moving vehicles is different from that of relevant background, while the gradient of the moving shadow is similar to that of relevant background. Thus, the difference of the two gradient images can reserve most gradient information at the moving vehicles area which presents skeleton of the vehicles, and the shadow gradient at shadow region is removed as shown in Figure 13.

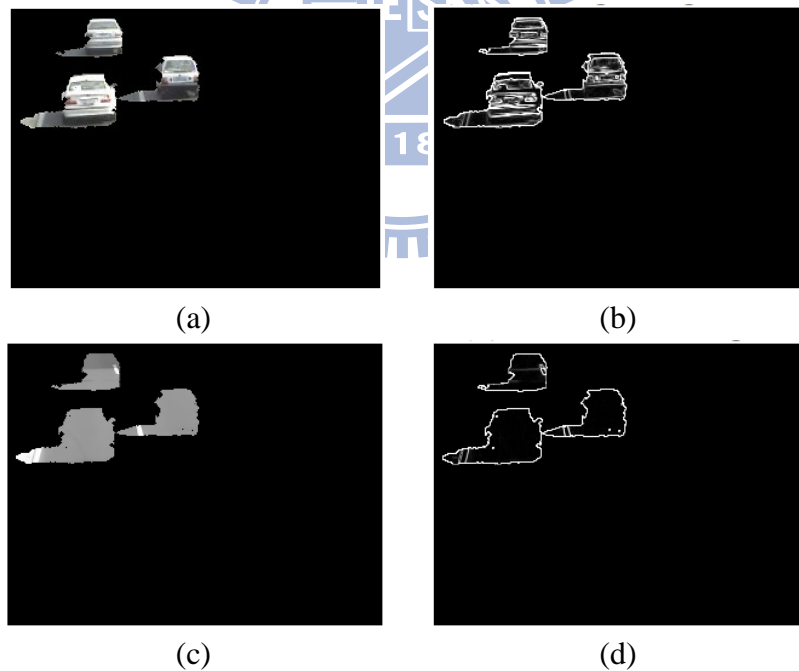


Figure 12. Gradient images of foreground and its relevant background. (a) The moving foreground. (b) The gradient image of the moving foreground. (c) The relevant background. (d) The gradient image of the relevant background.[20]

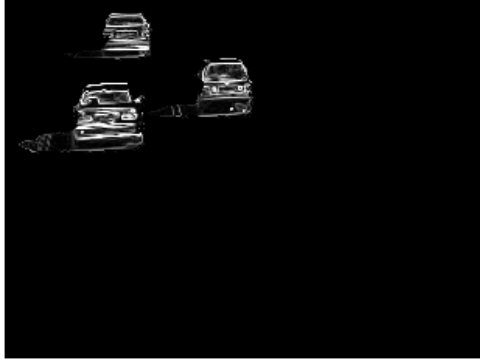


Figure 13. Result of shadow elimination by using gradient feature.[20]

Finally, we integrate the detected gradient of the moving vehicles with the moving vehicles which are executed shadow removal by color reflectance to construct more complete moving vehicles. In this way, since the vehicle's body is seriously damaged, the skeleton of the moving vehicles is able to make up for information loss. Thus, the morphological operation still can be applied for recovering the vehicle according to its gradient data.

In addition, in order to obtain the gradient information of the moving foreground and the relevant background, Sobel filter is used to detect the gradient information with horizontal and vertical operators as shown in Figure 14.

-1	0	1
-2	0	2
-1	0	1

(a)

1	2	1
0	0	0
-1	-2	-1

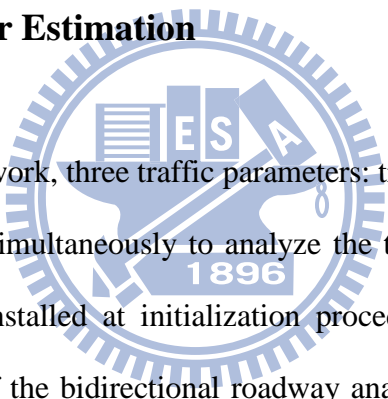
(b)

Figure 14. Operators of Sobel filter. (a) Operator for horizontal changes. (b) Operator for vertical changes.

4.3 Traffic Congestion Classification

Traffic congestion is the most useful information for the drivers among all traffic information. To reveal the degree of traffic congestion, traffic flow, traffic density, and traffic speed are the important and useful traffic parameters. Thus, we process a traffic surveillance video to estimate the traffic parameters and classify the traffic congestion into five levels: jam, heavy, medium, mild, and low. In the following sections, methods of traffic parameters estimation and traffic congestion evaluation are proposed and described.

4.3.1 Traffic Parameter Estimation



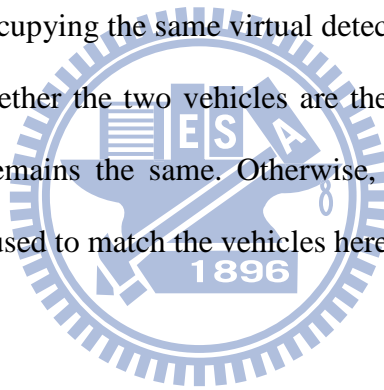
In the proposed framework, three traffic parameters: traffic flow, traffic speed and traffic density are needed simultaneously to analyze the traffic congestion. Thus, we use the virtual detectors installed at initialization procedure to estimate the traffic parameters. On the basis of the bidirectional roadway analysis, the traffic parameters can be calculated for both directions of the roadway individually.

(1) *Traffic Flow*

Traffic flow Fl is defined as the number of moving vehicles passing through the scene in a time interval. In traditional methods, tracking all the moving vehicles on the roadway is a conventional way to calculate the flow. However, tracking all the vehicles on the roadway is extremely complicated and time-consuming for vehicle occlusion problem and lane changing behavior. Therefore, in our proposed framework, the virtual detectors on each lane of roadway are utilized for counting the traffic flow. So traffic flow is defined as how many vehicles trigger the virtual detectors in a time

interval. The vehicle triggers the virtual detector when it passes through the detector. To ensure that the vehicle truly triggers the detector, the foreground pixels of the vehicles have to occupy at least a portion (say a quarter) of the triggered virtual detector. This limitation can reduce the erroneous judgment caused by noises. The ratio can be changed with the quality of surveillance videos.

For the usage of virtual detectors, the vehicles are tracked when they trigger the virtual detectors, so the vehicle tracking on the whole roadway is unnecessary. A useful property is that only one virtual detector is installed on each lane, which simplifies the vehicles matching in vehicle tracking procedure. Because only one vehicle can occupy one virtual detector at the same time in normal situation, we just match the vehicle that is occupying the same virtual detector in two consecutive video frames for determining whether the two vehicles are the same. If they are the same vehicles, traffic flow Fl remains the same. Otherwise, it increases one. The color histograms of vehicles are used to match the vehicles here.



(2) Traffic Speed

Traffic speed Sp is the average speed of the moving vehicles in a time interval. Generally speaking, speed is a ratio between moving distance and the time spent. To achieve this goal, it is necessary to track the vehicle for the length of its trajectories and to record the time to generate the trajectories. As discussed in previous sections, vehicle tracking on the whole roadway is not feasible in the complicated traffic situation. Hence, to obtain the speed of the moving vehicles, we estimate the speed of a vehicle when it triggers a virtual detector. In principle, a slower moving vehicle will trigger a virtual detector for more consecutive frames, but the situation for a fast moving vehicle is opposite. Hence, the speed approximation can be done by counting the number of frames that a moving vehicle triggers a virtual detector. Therefore,

traffic speed is refined as the average of frames for vehicles triggering the virtual detector in a time interval. The following equation is the definition of traffic speed S :

$$Sp = \frac{fps}{\frac{1}{Fl} \sum_{i=1}^{num_veh} fv_i} \quad (23)$$

where Fl is traffic flow, fv_i denotes the number of frames for that i th vehicle triggers the virtual detector, num_veh is the number of vehicles triggering the virtual detector in a time interval, and fps (frames per second) stands for sample rate of the surveillance video.

(3) Traffic Density

In general, traffic density is a ratio between the number of vehicles and the area of roadway. However, it is a difficult task to correctly segment all vehicles on roadway due to vehicle occlusion problem which is common at the far side of camera capturing. Consequently, we choose another way to calculate the density of traffic.

After segmenting out all pixels of foreground image in current frame, the traffic density is the ratio between the number of pixels in foreground and the number of pixels of roadway. Thus, the more the foreground pixels are, the higher the traffic density is. Nevertheless, there is still a problem that the moving vehicle which is near the camera is much larger than it is far from the camera. This situation makes the defined density out of reality. Because a vehicle occupy the same ratio of width of roadway regardless of distance from camera, the ratio between foreground pixels and roadway are calculated row by row to reduce the influence of camera's vision depth. The average of the ratios for frames over a period of time is the traffic density $Dens$ is defined below:

$$Dens = \frac{1}{num_fr} \times \sum_{f=1}^{num_fr} \sum_{i=1}^g \frac{PF_{fi}}{PR_{fi}} \quad (24)$$

where num_fr is the number of frames in a time interval, g is the height of roadway mask, PF_{fi} is the number of foreground pixels of row i in frame f , PR_{fi} is the width of roadway mask of row i in frame f .

4.3.2 Traffic Congestion Evaluation and Classification

To evaluate the traffic congestion for traffic flow over a period of time, how to use the traffic parameters estimated during traffic monitoring is an important issue. In general, the higher the traffic density is, the more crowded the traffic is; the slower the traffic speed is, the more crowded the traffic flow is. Thus, the congestion evaluation can be designed on the basis of relation between the two traffic parameters. Therefore, the traffic congestion value $Cong$ is defined as a ratio between traffic density and traffic speed. Thus, the higher the value is, the more crowded the traffic flow is. The congestion value becomes large while traffic density is larger and traffic speed is smaller. The equation of traffic congestion value is

$$Cong_{day} = \frac{Dens}{Sp} \quad (25)$$

where $Dens$ is traffic density and Sp is traffic speed, and the value will be used to make traffic congestion classification. As for traffic flow, it increases as the traffic congestion becomes crowded. However, it decreases when traffic is too crowded to move. Hence, the value is not directly employed to evaluate traffic congestion but used in traffic speed Sp evaluation.

For a clip of traffic surveillance video, congestion value is the only feature used for classification. Based on this feature, the traffic congestion levels are classified into five levels: jam, heavy, medium, mild, and low. In our framework, we adopt two methods for classification: Neighboring Class Distinguishing (NCD) and SVM classifier.

For NCD, the congestion values are divided into five intervals which stand for five levels of congestion respectively, and the congestion level of a congestion value is the corresponding interval where the congestion value locates. Moreover, thresholds, τ_{d1} , τ_{d2} , τ_{d3} , τ_{d4} , between five congestion levels are determined by training in advance. As shown in Figure 15, for two neighboring congestion levels, the mean value of the average congestion values $Cong_{day}$ in two level training data is the threshold between the two levels. For SVM classifier, the training data in five congestion levels are used to train a classifier model which is the base to classify the traffic congestion in testing video.

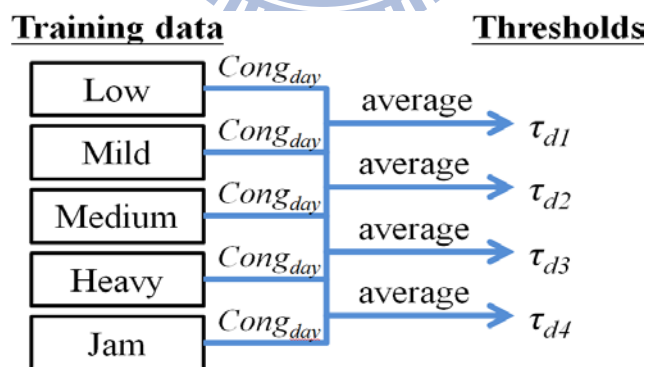


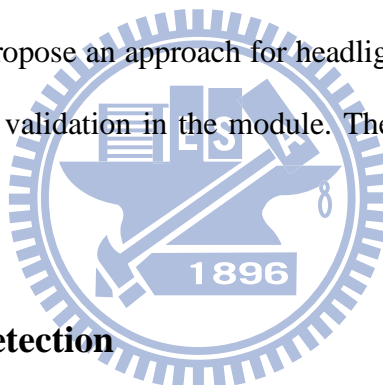
Figure 15. Determination of traffic congestion thresholds for NCD.

Chapter 5. Nighttime Traffic Congestion Classification

In this chapter, a nighttime traffic congestion classification framework is presented. The details of the module consisting of headlights detection, vehicle detection, and traffic congestion classification are described in the following sections.

5.1 Headlight Detection

For the purpose of recognizing moving vehicles at night, the headlights must be detected in advance. The most important features of headlight at night are luminance and shape. Therefore, we propose an approach for headlight detection including bright region detection and shape validation in the module. The details of the approach are described as follows.



5.1.1 Bright Region Detection

During night, the most consistent and powerful feature of headlights is the luminance. The bright region that has high luminance is the most possible area of headlights. Unfortunately, headlights are usually not the only bright region in a frame, which influences the headlight detection seriously. Hence, how to segment the headlight pixel from other bright pixels is the difficulty in bright region detection.

In Figure 16, we discover that the center of a headlight is always the area with highest luminance regardless of different color lights. Moreover, the central area looks white and non-colored in human vision. Based on this observation, as shown in Table 1, we examine the headlights in Figure 16 considering the changes of luminance and

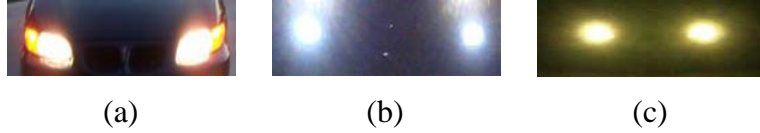


Figure 16. Headlights with different color lights. (a) Orange. (b) Blue. (c) Yellow.

Table 1. Luminance and color variation from the center to boundary of headlights.

SEA	Figure 16(a)				Figure 16(b)				Figure 16(c)			
	left		right		left		right		left		right	
	L	CV	L	CV	L	CV	L	CV	L	CV	L	CV
9	253	1	254	1	255	0	254	0	252	2	252	2
121	252	2	253	1	254	0	254	0	252	2	251	4
225	237	9	246	6	254	0	253	1	245	16	242	19

*SEA: the size of examination area; L: luminance; CV: color variation

color variation of red, green, blue channels from its center to the boundary. Through the results, the color variation is low at the center and increases from center to boundary, which means that the boundary is much colorful than center. Therefore, the region which has high luminance and low color variation is most likely the headlights. Instead, the area with low luminance or high color variance does not belong to headlights. This property is useful to identify the non-headlights area with high luminance such as light reflection on the roadway and the roadside objects.

Therefore, luminance and color variance are utilized simultaneously to detect the bright region in our framework. For a nighttime video frame, the luminance Y and the color variation σ in pixel (x, y) are calculated by the following equations:

$$Y_{(x,y)} = 0.229 \times I_{(x,y)}^{red} + 0.587 \times I_{(x,y)}^{green} + 0.114 \times I_{(x,y)}^{blue} \quad (26)$$

$$\sigma_{(x,y)} = \left(\frac{1}{3} \times \sum_{Z \in \{red, green, blue\}} (I_{(x,y)}^Z - I_{(x,y)})^2 \right)^{1/2} \quad (27)$$

where I^{red} , I^{green} , and I^{blue} stand for the pixel value in three color channels red, green, blue respectively and I denotes intensity. Then the pixels with high luminance and low color variation are the bright pixels.

$$B_{pixel} = \{(x, y) | Y_{(x,y)} > th_Y \text{ AND } \sigma_{(x,y)} < th_{CV}\} \quad (28)$$

where th_Y is the threshold for luminance and th_{CV} is the threshold for color variation. Consequently, only the pixels with high luminance and low color variation are regarded as bright pixels. The values of the two thresholds are changed according to the scene in nighttime video. Then, the fragmentary bright pixels are eliminated by filtering noises and manipulating morphological operation. Finally, we employ connected component analysis to extract all bright blobs B_{blob} :

$$B_{Blob} = \{B_{blob_h} | h = 1, 2, \dots, z\} \quad (29)$$

where z is the number of bright blobs. Figure 17 demonstrates an example process of bright region detection. Figure 17(a) shows a frame captured on freeway at night. Figure 17(b) shows the bright blobs detected from Figure 17(a). The headlights and some reflections of light on the roadside fence are extracted. Besides, we discover a phenomenon that all headlights far from the camera are connected together so that they cannot be separated.



Figure 17. Results of bright region detection.

5.1.2 Headlight Shape Validation

After bright region detection, all bright blobs are possible to be a headlight. However, some bright blobs are results from reflection of light on the roadside objects, and not all bright blobs are headlights. Hence, it is necessary to apply another constraint to the bright blobs. Shape is also a strong feature for headlight which usually looks round in the nighttime video. Therefore, all bright blobs are verified by its roundness. To evaluate the roundness of the blob, first we define rim_h as the pixels at the rim of bright blob B_{blobh} and (BX_h, BY_h) as the center of B_{blobh} .

$$BX_h = \frac{1}{N_h} \sum_{(x,y) \in B_{blobh}} x \quad (30)$$

$$BY_h = \frac{1}{N_h} \sum_{(x,y) \in B_{blobh}} y \quad (31)$$

where N_h is the number of pixels in bright blob B_{blobh} . Then, all the distances d_{hi} between the center (BX_h, BY_h) and the pixels in rim_h are calculated in Euclidean distance. D_h is the set of all distances for B_{blobh} .

$$D_h = \left\{ d_{hi} = \left((x_i - BX_h)^2 + (y_i - BY_h)^2 \right)^{1/2} \mid (x_i, y_i) \in rim_h \right\} \quad (32)$$

If all distances d_{hi} in D_h are the same, the bright blob B_{blobh} is a circle. Hence, we use variance of D_h to reveal roundness of the blob. The lower the variance is, the rounder the blob is. Thus, the roundness of B_{blobh} is defined as follow:

$$Roundness_h = \left(\frac{1}{NR_h} \sum_{i=1}^{NR_h} (d_{hi} - \overline{D_h})^2 \right)^{1/2} \quad (33)$$

where NR_h is the number of pixels at rim of B_{blobh} and $\overline{D_h}$ is the mean value of D_h .

Finally, the bright blobs that satisfy the following conditions are regarded as the headlight candidates B_{blob} '.

$$B_{blob}' = \{B_{blobh} \text{ whose } Roundness_h < th_S \mid h = 1, 2, \dots, z\} \quad (34)$$

$$th_S = 0.1 \times (W_{B_{blobh}} \times H_{B_{blobh}})^{1/2} \quad (35)$$

where th_S is the adaptive threshold for roundness which adjusts with different blobs,

$H_{B_{blobh}}$ is the height of blob B_{blobh} , and $W_{B_{blobh}}$ is the width of blob B_{blobh} .

5.2 Vehicle Detection

Due to low contrast in nighttime surveillance video, typical vehicle detection technique used in daytime cannot work well during night. In order to overcome these restrictions, we realize that the salient feature of vehicles, headlights, can reveal the presence of a vehicle at night. As discussed in Section 5.1, the candidates of headlight are extracted using luminance and shape. Next, how to group the headlights into individual vehicles is discussed in this section.

5.2.1 Headlight Correlation

The relationships between a pair of headlight candidates can be estimated with three criteria: the width, height, and the number of edge pixels between two headlights. In general, the width should be similar to the width of vehicle; the height is close to zero if the roadway surface is even; the edge between two headlights of a vehicle is much complicated than that between two different vehicles due to textured

front of the vehicles and smoothness of the roadway surface. Thus, all these three conditions are evaluated for the relationship of two headlights.

The relationship of the width is defined in the following equation. The value is from 0 to 1. If the value is much higher, the width of two headlights is more suitable for the headlights belonging to a same vehicle. Otherwise, they may be on different vehicles.

$$RW_{ij} = \frac{1}{1 + \frac{|W_{ij} - W_{expected}|}{E_{allowable}}}, \quad (36)$$

where W_{ij} is the width between headlights i and j in x-axis, $W_{expected}$ is the pre-decided value for expected width of vehicle, and $E_{allowable}$ is the allowable inaccuracy for width W_{ij} headlights. $W_{expected}$ is varied depending on the scene of surveillance and $E_{allowable}$ is determined according to the quality of the surveillance video.

The relationship of the height between two headlights is defined in Eq.(37). The value is also from 0 to 1. If the value is much lower, the detected height of two headlights is more possible for the headlights on a same vehicle. Otherwise, they may belong to different vehicles.

$$RH_{ij} = \frac{1}{1 + \frac{H_{ij}}{H_{left} + H_{right}}}, \quad (37)$$

where H_{ij} is height between headlights i and j . As shown in Figure 18(a), if right headlight is lower than left headlight in y-axis. H_{left} is the height between headlight center to its bottom, and H_{right} is the height between the headlight center to its top. Otherwise, as shown in Figure 18(b), H_{left} is the height between headlight center to its top, and H_{right} is the height between the headlight center to its bottom.

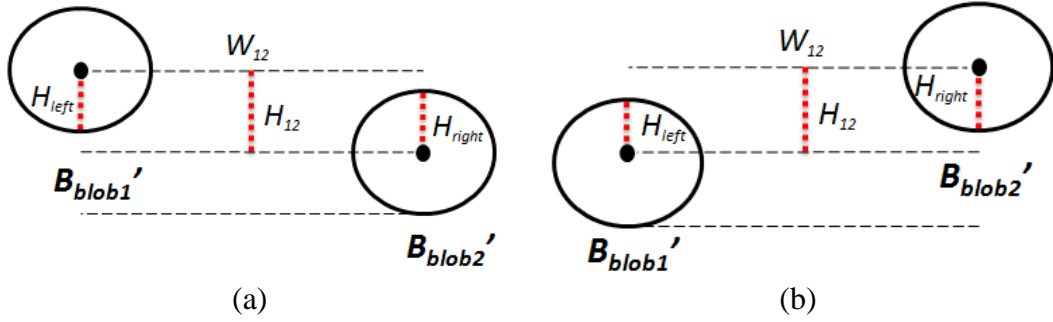


Figure 18. The parameters of height relationship for two headlights. (a) Right headlight is lower than left headlight in y-axis. (b) Left headlight is lower than right headlight in y-axis.

Furthermore, the relationship of the edge is evaluated by calculating the number of edge pixels between two headlights. The estimated value is between 0 and 1. The higher the value is, the more possible the headlights belong to a vehicle.

$$E_{ij} = \frac{NE_{ij}}{W'_{ij} \times H'_{ij}} \quad (38)$$

where NE_{ij} is the number of edge pixels between headlights i and j , W'_{ij} is the width of two headlights and H'_{ij} is the height of two headlights. The Figure 19 illustrates the related parameters for relationship of the edge. Moreover, in a bright scene, this value is remarkable because of clear textures on front of vehicle. In much darker scene, the value is always small regardless of the pair of headlight candidates. Thus, the value is more useful in bright scene rather than darker scene.

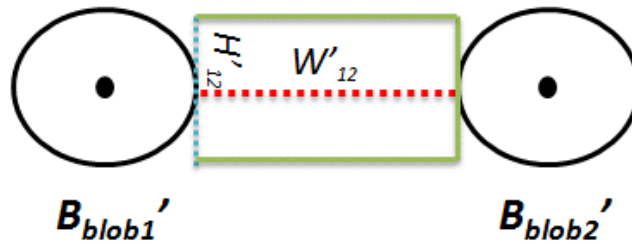
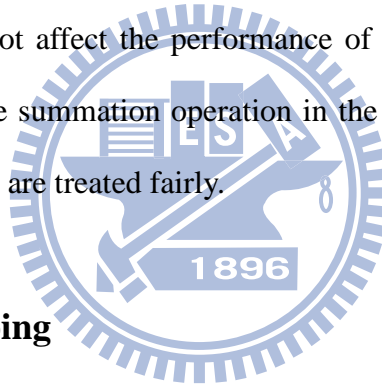


Figure 19. The parameters of edge relationship of two headlights.

After evaluating three relationships, we combine all three values for estimating correlation between two headlights by Eq.(39). The estimated value is ranging from 0 to 3. The higher the value is, the more possible the headlights belong to a vehicle.

$$Correlation_{ij} = RW_{ij} + RH_{ij} + E_{ij} \quad (39)$$

where ij stands for the headlight i and j , RW_{ij} is the relationship of width, RH_{ij} is the relationship of height, and E_{ij} is the relationship of edge. The higher the value is, the closer the relationship is. This means the possibility of these two headlights belonging to a same vehicle is higher than other pairs with lower values. As discussed in previous paragraph, the E_{ij} is much notable in bright scene. However, it can work well in bright scene and does not affect the performance of correlation evaluation under darker scene because of the summation operation in the equation. So, all vehicles in the same surveillance video are treated fairly.



5.2.2 Headlight Grouping

Due to all pairs of headlights are the candidates of vehicles, once correlations of headlights are calculated, the headlights are going to be grouped into individual vehicles. Based on the headlight correlations, our proposed headlights grouping algorithm checks all pairs of headlights and determines the pair whether it is a single vehicle. All pairs of headlight candidates are defined as the vehicle candidate set, called VC . As described in Algorithm 5.1, first the headlight pair which has the highest correlation in VC is regarded as a single vehicle if both its width relationship and height relationship are higher than 0.5. This constraint can avoid unreasonable pair being considered as a vehicle. Then, the pair of the detected vehicle is removed

from VC . Second, because no headlight can be on two different vehicles at one time, if two headlights are detected as a vehicle, the other headlight pairs that contain the two headlights must be discarded from VC . Afterward, repeat the two steps until the VC is empty.

As a result of adopting virtual detection line in the proposed module, only the headlights that pass through the virtual detection line need to be detected and grouped. It is unnecessary to process all headlights on the roadway so that the complexity of headlight grouping algorithm is reduced and accuracy is increased.

Algorithm 5.1: Headlights Grouping Mechanism

Input: All pairs of headlights with correlation, width and height relationships

Output: The pairs of headlights stand for a single vehicle.

```

1  while ( $VC$  is not empty)
2      do from  $VC$ , select the pair  $(i, j)$  with maximum  $Correlation_{ij}$ 
3          if  $RW_{ij} > 0.5$  AND  $RH_{ij} > 0.5$ 
4              then
5                   $B_{blob_i}$  ' and  $B_{blob_j}$  ' are a pair of headlights AND
6                   $VC = VC - \{ \text{pair } (a,b) \mid (a,b) \in \{ (i, j), (i-1, i), (j, j+1) \} \}$ 
7              else  $VC = VC - \{ \text{pair } (i, j) \}$ 

```

5.3 Traffic Congestion Classification

In order to evaluate the traffic congestion degree, traffic flow, traffic density, and traffic speed are useful traffic parameters for achieving this goal. However, it is difficult to estimate the parameters in nighttime traffic surveillance. Because the vehicle detection based on headlight detection also does not work on the moving vehicles which are far from the camera. These headlights scatter the lights so that all headlights are connected together. Hence, the headlights cannot reveal accurate size

and position of moving vehicles on roadway. Moreover, tracking is difficult particularly while a large number of vehicles existing on the roadway and insufficient information to distinguish vehicles.

Therefore, we use the virtual detection line which is placed in the near side of camera in the frames to evaluate the traffic congestion. The virtual line counts the number of times that vehicles touch the line during traffic monitoring. The more times the line is touched by vehicles, either the more the vehicles run on the roadway or the slower the vehicles move on the roadway. Hence, the value of traffic congestion $Cong_{night}$ for nighttime video is defined as follow:

$$Cong_{night} = \frac{num_touch}{fps \times N_{lane}} \quad (40)$$

where num_touch stands for the number of that vehicles touch the virtual detection line over a period of time, N_{lane} is the number of lanes on roadway, and fps denotes the sample rate of video. For a clip of video, traffic congestion levels are classified into five levels: jam, heavy, medium, mild, and low by congestion value. For classification, the range of congestion values is divided into five intervals standing for five levels of congestion, respectively. The threshold between two congestion levels is determined by the training data in two corresponding levels. The rationale is same as daytime traffic congestion classification as described in Section 4.3.2.

Chapter 6. Experimental Results and Discussions

In this chapter, we are going to demonstrate the performance of the proposed framework. In section 6.1, the experimental results of daytime traffic congestion classification are shown and discussed, and those of nighttime traffic congestion classification are presented in section 6.2.

6.1 Daytime Traffic Congestion Classification

In this section, we present our experiments of the proposed daytime traffic surveillance congestion classification module. The experimental results of roadway detection, bidirectional roadway analysis, virtual detectors installation, and traffic congestion classification are shown in the following sections. Moreover, the traffic surveillance videos captured on different freeways by Taiwan Area National Freeway Bureau [44] are used as the experimental data. The resolution of video frames is 352×240 , and sample rates vary from video to video. For the experiments, we use a computer with AMD 2.8 GHz dual-core CPU and 2.0 GB memory.

6.1.1 Roadway Detection

In this section, we present and discuss the results of roadway detection. The setting of threshold values: th_M for identifying movements is 40, th_R for noise filter is 0.6, and T_R for reserving main roadway areas is 0.95. These parameters are mentioned in Section 4.1.1. Figure 20 shows some results of roadway detection which use seven traffic surveillance videos captured in different scenes as the training data. Figure 20(a)

demonstrates original scene of traffic surveillance, Figure 20(b) shows roadway candidate R_c accumulated from a sequence of frames, Figure 20(c) illustrates the mask of detected roadway region.

Through the results, the shape and position of roadway have roughly revealed after accumulating the vehicle movements. After noise filtering, it is obvious that the noises mostly coming from the swaying trees outside the roadway region would be eliminated from the roadway candidate R_c . In addition, the holes in the roadway and the small gaps between two roadways in the different directions are filled to extract complete roadway region. Compared to the previous work proposed in [7], our algorithm solves the problems that the slight changes between consecutive two frames are also regarded as portions of roadway mask and unbalance traffic flow in different lanes results in disappearances of partial roadway region.

The traffic volume in the training data is a key factor to accumulate the vehicle movements. If we use a traffic surveillance video which has low traffic volume to detect the roadway, the length of the video should be much longer for avoiding insufficient information. Furthermore, the videos that has unstable environment outside the roadway may cause some large noises that is unable to be filtered. Therefore, our approach is relatively suitable in the suburbs and countryside.

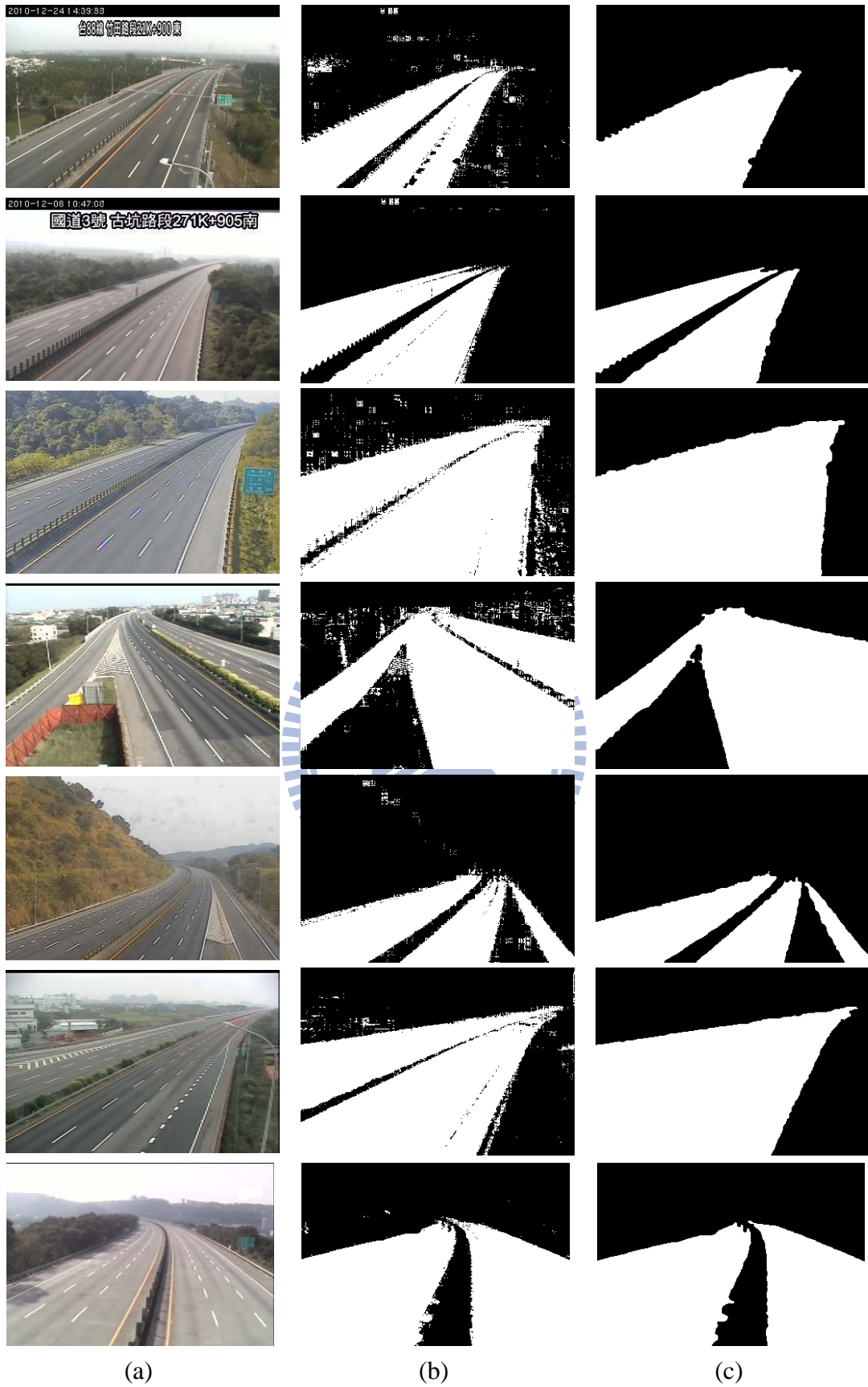


Figure 20. Experimental results of roadway detection. (a) Original scene. (b) Roadway candidate R_c . (c) Complete roadway mask R .

6.1.2 Bidirectional Roadway Analysis

In this section, we show some results of bidirectional roadway analysis. The demonstrations including original scene of traffic surveillance, motion image accumulated from a sequence of frames, and the mask of detected bidirectional roadway region in seven different traffic scenes are shown in Figure 21.

In Figure 21(b) and Figure 21(c), the pixels on roadway region are illustrated by two different colors for two kinds of motions. The white pixels stand for the motion DOWN and gray pixels denotes motion UP. From Figure 21(b), we clearly realize that the motions have roughly revealed the different directions of the roadway. As shown in Figure 21(c), a red center line CL is calculated to successfully divide the motions into two parts for identifying the two directions of roadways. In previous work [7], their mechanism to separate bidirectional roadway is limited in the three roadway types only and requires a clear gap between two parts of the roadway. Instead, in our approach, the bidirectional roadway can work well even if the roadway contains interchanges as shown in Figure 21(4) and Figure 21(5) or even there is no clear gap between two roadways in the different directions as shown in Figure 21(3) and Figure 21(6).

In some cases, unsuccessful bidirectional analysis may result from insufficient motions in the construction of motion image. In addition, a shortcoming of our algorithm is that the center line for separating bidirectional roadway must be a straight line, which causes some mistakes if the roadway is a curved shape.

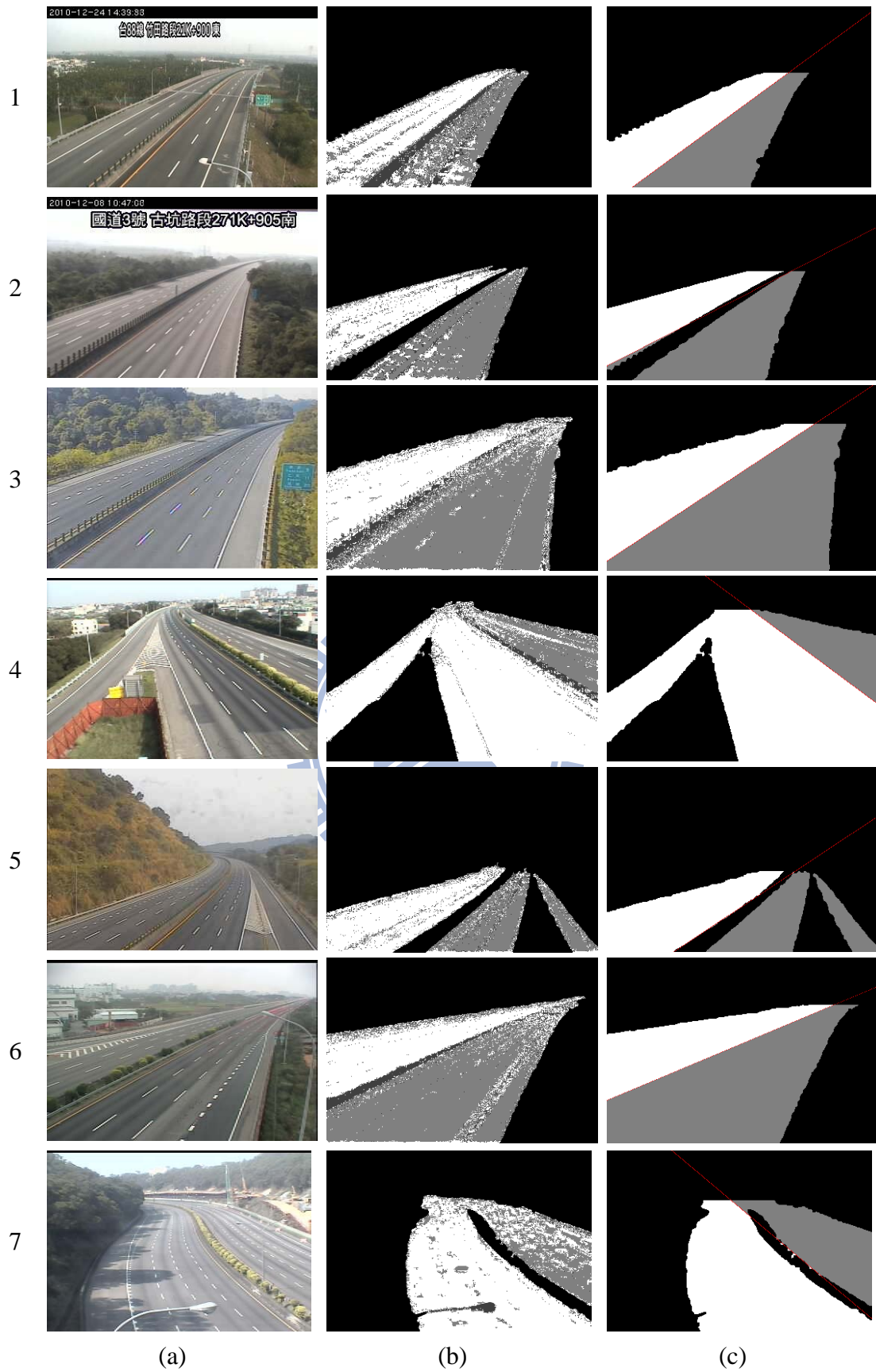


Figure 21. Experimental results of bidirectional roadway analysis. (a) Original scene. (b) Motion image M . (c) Bidirectional roadway mask.

6.1.3 Virtual Detector Installation

The experimental results of virtual detectors installation are displayed and discussed in this section. Figure 22 shows some results of virtual detector installation which use seven traffic surveillance videos captured in different scenes as the training data. Figure 22(b) is the record of the detected central points of moving vehicles. Those points display the broadly trajectories of vehicle moving. As shown in Figure 22(c), the virtual detectors are placed on each lane as a color rectangle. According to the bidirectional analysis, the virtual detectors in different directions are illustrated with different colors. The red rectangles are the virtual detectors for monitoring direction DOWN, and the blue rectangles are for direction UP.

The experimental results show that our approach has the ability to detect the lanes even though the roadway has lane markings in different shapes or not clear lane markings. Besides, in Figure 22(4), we discover that the positions of virtual detectors are not actually in the center of lane due to some inaccuracy caused by driving behaviors.

In this experiment, the detection row d is chosen at two-third height of a frame from top automatically or selected manually according to the effective resolution of perspective.

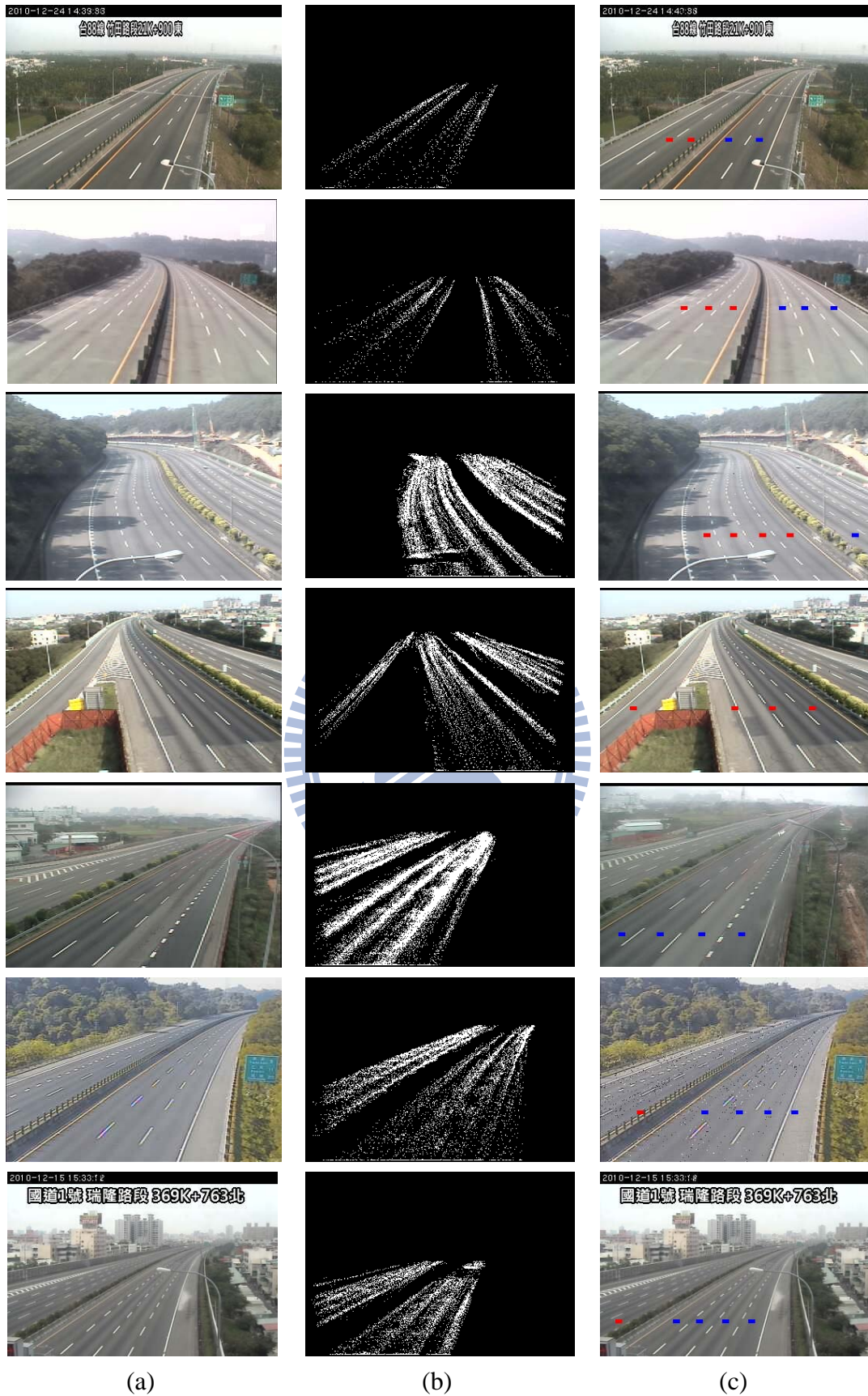


Figure 22. Experimental results of virtual detector installation. (a) Original scene. (b) Central points of moving vehicles. (c) Result of virtual detector installation.

6.1.4 Traffic Congestion Classification

In this section, we demonstrate the experimental results of classification for traffic congestion. In our experiment, a unidirectional traffic flow is the unit of our experimental data. Thus, for a bidirectional roadway monitoring surveillance video, two traffic flows in different directions are analyzed individually. In other words, we can get two experimental video clips in a bidirectional traffic surveillance video. In the experiment, 226 video clips captured on different freeways from traffic surveillance cameras are used as the experimental data. 159 video clips are used for testing data, and others are for training data. The length of a video clip is 60 seconds. After evaluating the traffic congestion $Cong_{day}$, we use two approaches, Neighboring Class Distinguishing (NCD) and Support Vector Machine (SVM) [45], to classify traffic congestion into five levels.

The distribution of training data and testing data is shown in Table 2. The ground truths of the experimental data are determined manually according to the principles described in Chapter 1. In order to determine the thresholds between the different congestion levels, the values of $Cong_{day}$ of each training video clip in five congestion levels are estimated by our proposed framework in advance. The approach of threshold determination is mentioned in Section 4.3.2.

Table 2. Distribution of experimental daytime surveillance video data.

<i>Congestion Levels</i>	Jam	Heavy	Medium	Mild	Low
<i>No. of Testing data</i>	12	30	26	35	56
<i>No. of Training data</i>	6	12	10	15	24

For evaluating the performance of traffic congestion classification, the classification accuracy of the video data in one congestion level is defined as follow:

$$accuracy_{level} = \frac{num_{classified}}{num_{all}} \quad (41)$$

where $num_{classified}$ is the number of well-classified testing video clips in one congestion level and num_{all} is the number of all testing video clips in the level. The well-classified video clips denote the videos that are successfully classified into the congestion level as same as the ground truth.

(1) Neighboring Class Distinguishing (NCD)

By using the traffic parameter $Cong_{day}$ to classify the traffic congestion level, accuracy of congestion classification is 89.2%, which is the average of $accuracy_{level}$ of five levels.

In order to present the robustness of the $Cong_{day}$, we compare the results of classification by three different traffic parameters: traffic congestion $Cong_{day}$, traffic density $Dens$, and traffic speed Sp . Table 3 shows the average $accuracy_{level}$ in five congestion levels by using different parameters. This validates that $Cong_{day}$ is the best among the three parameters for congestion classification. Moreover, for getting to know the impact of three traffic parameters on different congestion levels, Figure 23 shows the classification accuracies by using different parameters in five different congestion levels. We can discover that $Cong_{day}$ is also the best choice for classification in all congestion levels except jam condition. The reason is that the precision of traffic speed estimation is relatively lower than other levels because of vehicle occlusion. Instead, in jam level, traffic density is able to present the condition of traffic congestion correctly. On the other hand, traffic speed is the worst parameters

to reveal the correct traffic congestion situation. Because the vehicle speed must be much slower in the congestive condition but be uncertain in the unobstructed situation due to driving behaviors, the traffic speed Sp is not reliable enough to distinguish traffic congestion levels.

Table 3. Comparison of classification performance with different features (NCD).

<i>Traffic parameter</i>	Traffic congestion ($Cong_{day}$)	Traffic density ($Dens$)	Traffic speed (Sp)
Accuracy	89.2%	87.6%	59.4%

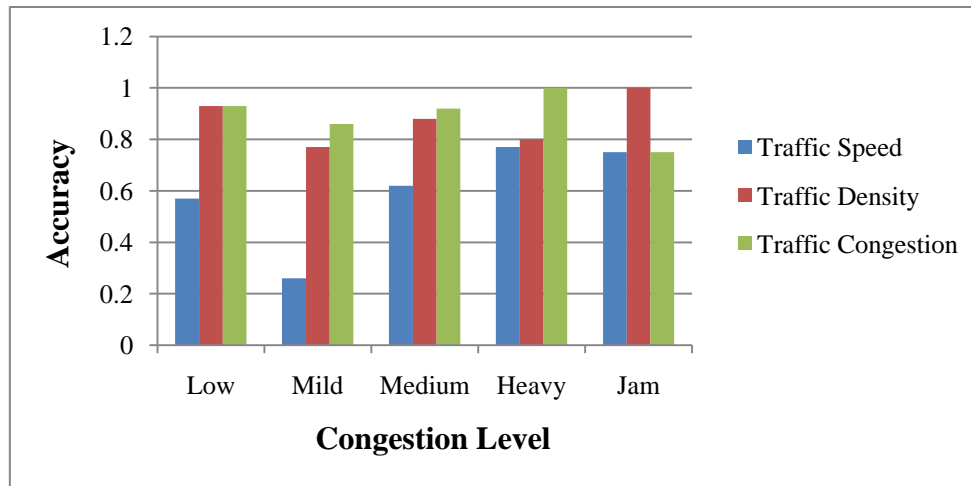


Figure 23. Classification accuracy in five levels by different parameters (NCD).

The detailed results of classification by using different features with respect to different levels of traffic congestion are shown in Table 4, Table 5, and Table 6, respectively. From the results, we discover that traffic congestion $Cong_{day}$ may be mis-classified into only the neighboring traffic congestion levels. This means that $Cong_{day}$ is a reliable value to express the congestion situation because of no absurd mistakes. In the case of traffic speed Sp , it is relatively unreliable to show the correct congestion situation.

Table 4. Daytime traffic congestion classification results by using *Cong_{day}* (NCD).

		Results of classification					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (159)	<i>Low(56)</i>	52	4	0	0	0	0.93
	<i>Mild(35)</i>	4	30	1	0	0	0.86
	<i>Medium(26)</i>	0	2	24	0	0	0.92
	<i>Heavy(30)</i>	0	0	0	30	0	1.00
	<i>Jam(12)</i>	0	0	0	3	9	0.75

Table 5. Daytime traffic congestion classification results by using *Dens* (NCD).

		Results of classification					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (159)	<i>Low(56)</i>	52	4	0	0	0	0.93
	<i>Mild(35)</i>	5	27	3	0	0	0.77
	<i>Medium(26)</i>	0	1	23	2	0	0.88
	<i>Heavy(30)</i>	0	0	2	24	4	0.80
	<i>Jam(12)</i>	0	0	0	0	12	1.00

Table 6. Daytime traffic congestion classification results by using *Sp* (NCD).

		Results of classification					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (159)	<i>Low(56)</i>	32	12	12	0	0	0.57
	<i>Mild(35)</i>	7	9	19	0	0	0.26
	<i>Medium(26)</i>	3	1	16	6	0	0.62
	<i>Heavy(30)</i>	0	0	0	23	7	0.77
	<i>Jam(12)</i>	0	0	0	3	9	0.75

(2) Support Vector Machine (SVM)

The classification accuracy of congestion classification is 86.4% by using feature *Cong_{day}*, which is the average of *accuracy_{level}* of five levels.

The performance of traffic congestion classification by using SVM is similar to that by using NCD. Table 7 shows that traffic parameter *Cong_{day}* is the best traffic parameters to stand for the traffic congestion situation. The classification accuracies

by using different features in five levels are shown in Figure 24. Compared to NCD, the accuracy of classification is a little lower due to the different methods for threshold determination. For NCD, the threshold between two congestion levels is determined by the training data only in the two levels. However, for SVM, the threshold between two congestion levels is calculated based on the training data in all levels. For the characteristics of that the traffic congestion degree is linear (i.e. from low to jam), each threshold between two levels is decided by only the two level training data is better than by the training data in all the five levels.

Table 7. Comparison of classification performance with different features (SVM).

<i>Traffic Parameter</i>	Traffic congestion (<i>Cong_{day}</i>)	Traffic density (<i>Dens</i>)	Traffic speed (<i>Sp</i>)
<i>Accuracy</i>	86.4%	84.8%	49.6%

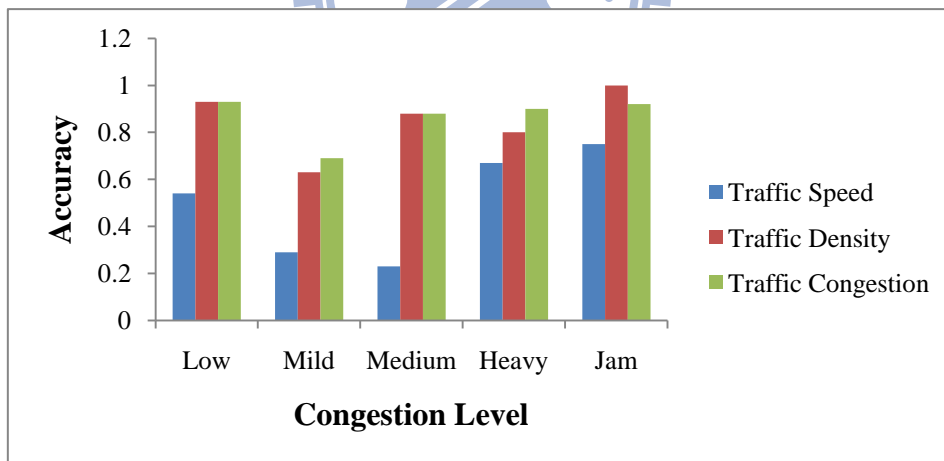


Figure 24. Classification accuracy in five levels by different parameters (SVM).

The detailed results of classification by using different features with respect to different levels of traffic congestion are shown in Table 8, Table 9, and Table 10, respectively.

Table 8. Daytime traffic congestion classification results by *Cong_{day}* (SVM).

		<i>Results of classification</i>					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (159)	<i>Low(56)</i>	52	4	0	0	0	0.93
	<i>Mild(35)</i>	6	24	5	0	0	0.69
	<i>Medium(26)</i>	0	1	23	2	0	0.88
	<i>Heavy(30)</i>	0	0	0	27	3	0.90
	<i>Jam(12)</i>	0	0	0	1	11	0.92

Table 9. Daytime traffic congestion classification results by *Dens* (SVM).

		<i>Results of classification</i>					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (159)	<i>Low(56)</i>	52	4	0	0	0	0.93
	<i>Mild(35)</i>	7	22	6	0	0	0.63
	<i>Medium(26)</i>	0	1	23	2	0	0.88
	<i>Heavy(30)</i>	0	0	2	24	4	0.80
	<i>Jam(12)</i>	0	0	0	0	12	1.00

Table 10. Daytime traffic congestion classification results by *Sp* (SVM).

		<i>Results of classification</i>					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (159)	<i>Low(56)</i>	30	17	9	0	0	0.54
	<i>Mild(35)</i>	15	10	10	0	0	0.29
	<i>Medium(26)</i>	17	2	6	1	0	0.23
	<i>Heavy(30)</i>	0	0	0	20	10	0.67
	<i>Jam(12)</i>	0	0	0	3	9	0.75

6.1.5 Performance of Execution Time

As shown in Table 11, the time for estimating traffic parameters is around 0.1 second per frame regardless of different congestion levels. In particular, the time for processing a frame is increased just a little as the traffic flow becomes crowded. That means the usage of virtual detectors is successful in reducing the computation cost as

the number of moving vehicles on roadway increases. As a whole, the proposed framework works well for the traffic surveillance video that processes at least 4 frames per second. So, using the key frames of videos is enough to accomplish the traffic congestion classification.

Table 11. Time for processing one frame in five congestion levels in daytime video.

<i>Traffic congestion</i>	Low	Mild	Medium	Heavy	Jam
<i>Time (s)</i>	0.08	0.09	0.10	0.10	0.11

6.2 Nighttime Traffic Congestion Classification

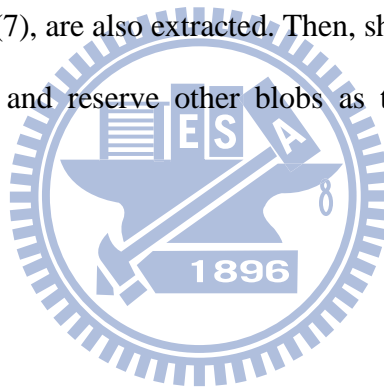
In this section, we are going to show the experimental results of the proposed nighttime traffic surveillance congestion classification module. The experimental results of headlight detection, vehicle detection, and traffic congestion classification are shown in the following sections. Moreover, the nighttime traffic surveillance videos that are captured on freeway by Taiwan Area National Freeway Bureau [44] are used as the experimental data. The resolution of video frames is 352×240 , and the sample varies from video to video. For the experiment, we use a computer with AMD 2.8 GHz dual-core CPU and 2.0 GB memory.

6.2.1 Headlight Detection

In this section, we present and discuss the results of headlight detection. The setting of the threshold: th_Y for luminance is 220 and th_{CV} for color variation is 20. These parameters are mentioned in Section 5.1. Figure 25(a) shows the original scene

of nighttime traffic surveillance and a horizontal green line denotes the virtual detection line, Figure 25(b) presents the detected bright region. Figure 25(c) illustrates the mask of detected bright blobs that touch the virtual detection line. The cyan bright blobs are the blobs which are filtered after headlight shape validation, but the red blobs are not.

Due to the use of virtual detection line, the headlight detection needs not to be applied to the whole bright pixels in the frame. Thus, only the bright pixels near the detection line are processed. Through the results, it is obvious that all headlights are extracted after bright region detection. Some bright blobs not belonging to headlights, such as reflection of light on the roadside fence in Figure 25(5) and neon lamps of a bus in Figure 25(7), are also extracted. Then, shape validation is executed to discard these bright blobs and reserve other blobs as the headlight candidates for further vehicle detection.



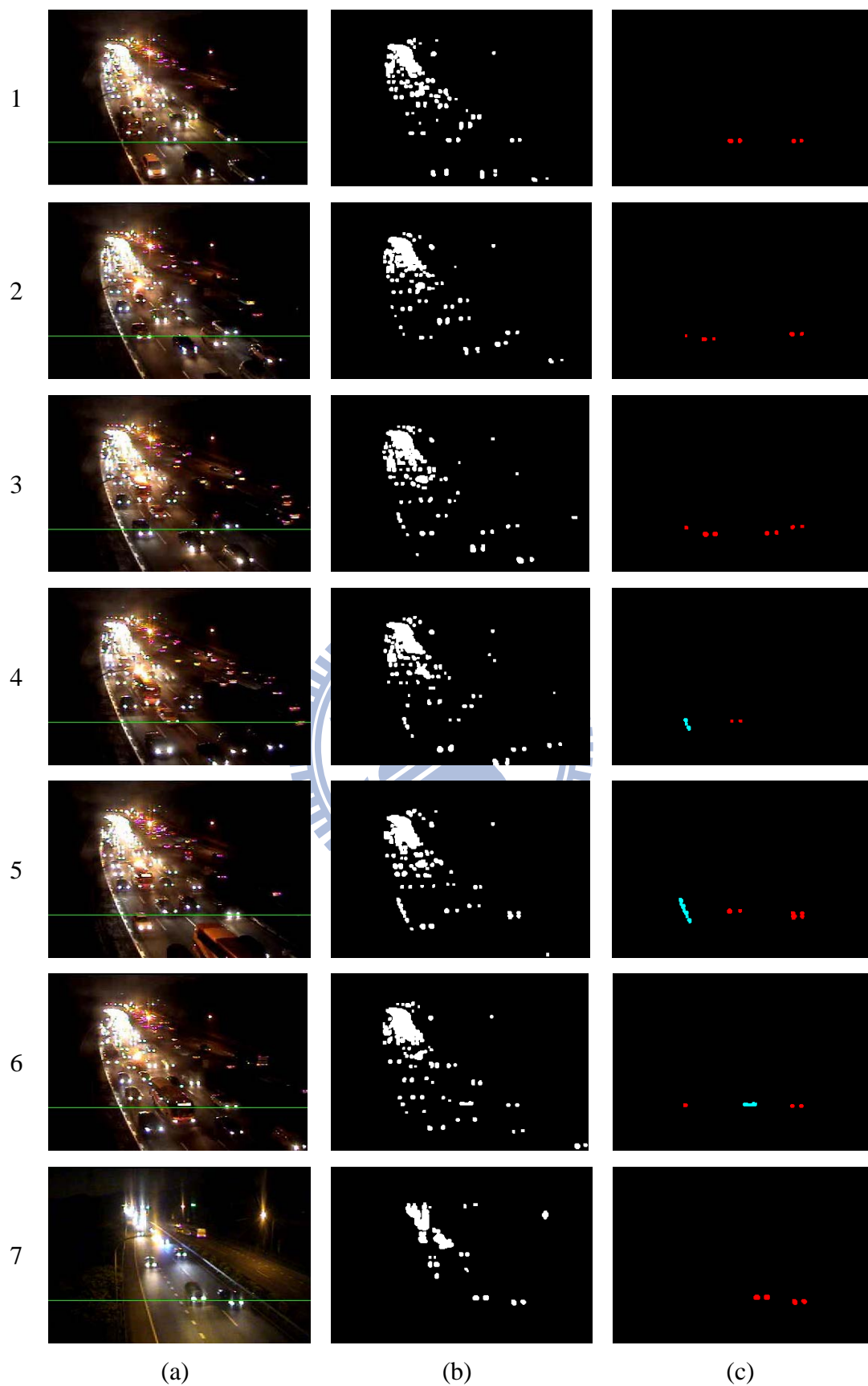


Figure 25. Experimental results of headlight detection. (a) Original frame captured at night. (b) Detected bright regions. (c) Bright regions after shape validation.

6.2.2 Vehicle Detection

In this section, we demonstrate some results of vehicle detection at night. The setting of the threshold: $W_{expected}$ for expected vehicle width is 20 and $E_{allowable}$ for error tolerance is 10 as mentioned in Section 5.2. Figure 26(a) shows the original scene of nighttime traffic surveillance and a horizontal green line as the virtual detection line. In Figure 26(b), the headlight candidates surrounded by a yellow rectangle are the actual headlights of vehicles and some solitary headlight candidates are removed. The horizontal line between two headlights indicates that the pair of headlights is a single vehicle. Through the results in Figure 26, we can find out that some headlight candidates that do not belong to the vehicle are removed by the vehicle detection as shown in Figure 26(2), Figure 26(3) and Figure 26(6).



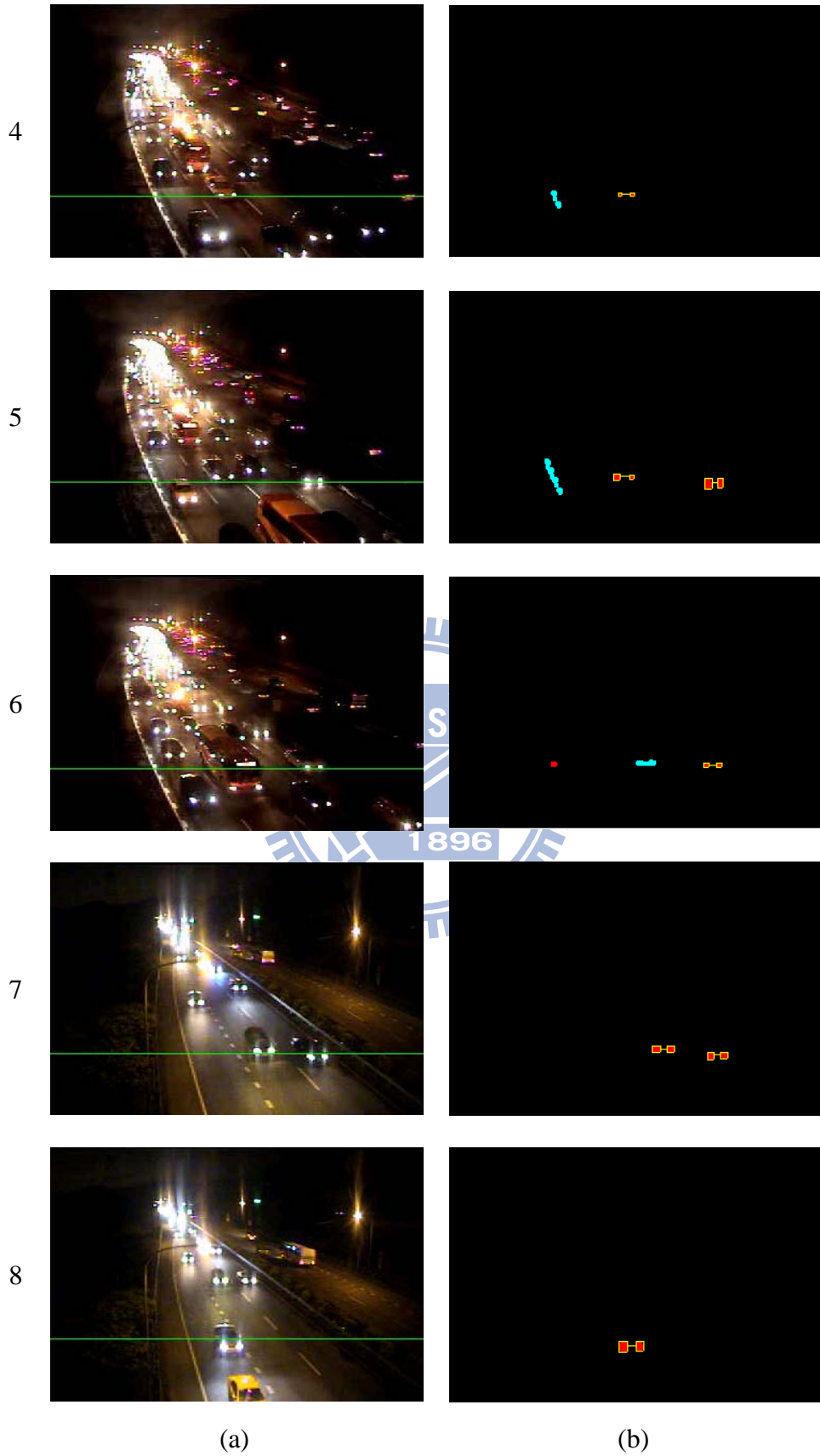


Figure 26. Experimental results of vehicle detection by headlight grouping. (a) Original frame captured at night. (b) Detected vehicles at virtual detection line.

6.2.3 Traffic Congestion Classification

We are going to present and discuss the experimental results of classification for nighttime traffic congestion levels in this section. Our approach is used to process the moving vehicles captured from vehicle front for headlight detection. Thus, only the incoming traffic flows can be analyzed in the nighttime surveillance videos. For the outgoing traffic flows, we can adopt another camera with opposite shooting direction to capture the traffic flows from vehicle front. In the experiment, there are 165 video clips, which monitor the unidirectional incoming traffic flows, are captured on the freeways from surveillance cameras during night. We use 49 video clips to train the thresholds for different congestion levels, and others are used as the testing video. The length of each video clip is 60 seconds.

The distributions of training data and testing data are shown in Table 12. The ground truths of the experimental data are determined manually according to the principles in Chapter 1. The traffic congestion $Cong_{night}$ of each training video clip in five congestion levels is estimated by our proposed framework, and all the results are used to determine the thresholds between the different levels by using the approaches mentioned in Section 5.3.

Table 12. Distribution of experimental nighttime surveillance video data.

<i>Congestion Levels</i>	Low	Mild	Medium	Heavy	Jam
<i>No. of Testing data</i>	21	21	47	21	6
<i>No. of Training data</i>	9	9	20	9	2

(1) Neighboring Class Distinguishing (NCD)

Based on the traffic congestion $Cong_{night}$ estimated by using virtual detecting line, accuracy of traffic congestion classification is 95.4%. The accuracy is calculated by the Eq.(41). The confusion matrix of the classification is shown in Table 13. From the results, we discover that traffic congestion $Cong_{night}$ may be mis-classified into the neighboring traffic congestion levels. This means that $Cong_{night}$ is a reliable value to express the congestion situation because of no absurd mistakes. For instance, the low traffic flow is mis-classified into jam level.

Table 13. Nighttime traffic congestion classification results by $Cong_{night}$ (NCD).

		Results of classification					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (134)	<i>Low(21)</i>	21	0	0	0	0	1.00
	<i>Mild(21)</i>	0	18	3	0	0	0.86
	<i>Medium(47)</i>	0	3	43	1	0	0.91
	<i>Heavy(21)</i>	0	0	0	21	0	1.00
	<i>Jam(6)</i>	0	0	0	0	6	1.00

(2) Support Vector Machine (SVM)

The accuracy of congestion classification by using SVM classifier is 88.2%. The result is lower than that by using NCD due to the different thresholds determination methods. For the properties of that the traffic congestion degree is linear (i.e. from low to jam), each threshold between two levels is decided by only the two level training data is better than by the training data in all the five levels.

Table 14. Nighttime traffic congestion classification results by $Cong_{night}$ (SVM).

		<i>Results of classification</i>					
		<i>Low</i>	<i>Mild</i>	<i>Medium</i>	<i>Heavy</i>	<i>Jam</i>	<i>Accuracy</i>
Ground Truth (134)	<i>Low(21)</i>	20	1	0	0	0	0.95
	<i>Mild(21)</i>	0	11	10	0	0	0.52
	<i>Medium(47)</i>	0	1	44	2	0	0.94
	<i>Heavy(21)</i>	0	0	0	21	0	1.00
	<i>Jam(6)</i>	0	0	0	0	6	1.00

6.2.4 Performance of Execution Time

The use of virtual detection line is successful in decreasing the time for processing frames in all congestion levels. In the experiments, the time for estimating traffic parameters is around 0.1 second per frame regardless of congestion levels. As a whole, the proposed framework works well for the traffic surveillance video that processes at least 4 frames per second. The result obviously shows the real-time response of our proposed framework. In addition, the size of headlight becomes a key factor to influence the execution time. It takes more time to process the larger headlights. As shown in Table 15, the jam level spends less time than that in the medium level due to the headlight size. That is the size of headlights has larger effect than congestion degree for the nighttime performance.

Table 15. Time for processing one frame in five congestion levels in nighttime video.

<i>Traffic congestion</i>	Low	Mild	Medium	Heavy	Jam
<i>Time (s)</i>	0.096	0.104	0.105	0.095	0.102

Chapter 7. Conclusions

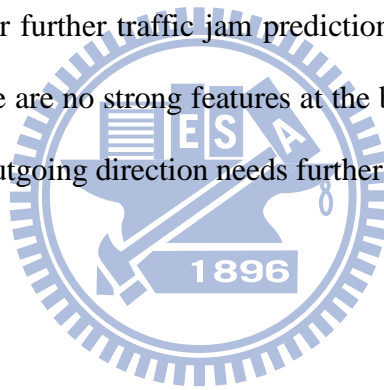
The traffic congestion classification framework which contains daytime and nighttime modules is proposed in this thesis. Through analyzing the traffic surveillance videos, our frameworks are able to recognize the traffic congestion level. The information of traffic congestion is useful for drivers to avoid traffic jam and for intelligent mobile devices to plan other alternate routes immediately.

In daytime module, an initialization procedure is used to obtain the consistent information of roadway, and the traffic congestion classification based on three traffic parameters are estimated from traffic surveillance video. During initialization procedure, automatic roadway detection, bidirectional roadway analysis and virtual detector installation methods are proposed to overcome the unbalanced traffic flow and roadway-type limitations in previous works. In addition, due to the use of virtual detectors, simplified procedure of vehicle tracking for traffic parameters estimation not only significantly reduces the cost caused by other complex algorithm, but also solves the difficulty of vehicle tracking in the complicated environment.

In nighttime module, we propose a vehicle detection method based on headlights detection and grouping, and the virtual detection line is employed to evaluate the traffic congestion. Headlights are extracted by using three strong features including luminance, color variation, and shape rather than just using luminance in earlier researches. Then not only the distance but also the edge feature between two headlight candidates are utilized to reveal the correlation of a pair of headlights. Consequently, vehicles are detected from the headlights according to the correlations. By calculating the frequency that vehicles touch the virtual detection line, the traffic congestion can be classified in real time.

As the approaches described above, we can enhance the ability of the traffic congestion evaluation in more complicated environment, such as a large number of vehicles and vehicle occlusions resulting from too low installation of a camera. Besides, the computational complexity is reduced to achieve the requirement of real-time response. The performance of the proposed framework is examined in videos with different surveillance scene and traffic congestion levels. We obtain the 89.2% and 95.4% accuracies of traffic congestion classification in daytime and nighttime surveillance videos, respectively.

In the future, once the traffic congestion level is classified instantly from surveillance camera, our frameworks can provide a sequence of patterns that represents the congestion condition for further traffic jam prediction with advanced data mining knowledge. Moreover, there are no strong features at the back of vehicle, so the traffic congestion evaluation for outgoing direction needs further investigations.



Bibliography

- [1] G. Wang, D. Xiao, and J. Gu, "Review on vehicle detection based on video for traffic surveillance," in *IEEE International Conference on Automation and Logistics*, Qingdao, China, 2008, pp. 2961-2966.
- [2] B. Li and Q.-mei Chen, "Framework for Freeway Auto-Surveillance from Traffic Video," in *WRI World Congress on Computer Science and Information Engineering*, Los Angeles, CA, USA, 2009, vol. 6, pp. 360-365.
- [3] B. Yu and A. K. Jain, "Lane boundary detection using a multiresolution Hough transform," in *International Conference on Image Processing*, Santa Barbara, CA, USA, 1997, vol. 2, pp. 748-751.
- [4] A. H. S. Lai and N. H. C. Yung, "Lane detection by orientation and length discrimination," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 30, no. 4, pp. 539-548, Aug. 2000.
- [5] B. D. Stewart, I. Reading, M. S. Thomson, T. D. Binnie, K. W. Dickinson, and C. L. Wan, "Adaptive lane finding in road traffic image analysis," in *International Conference on Road Traffic Monitoring and Control*, London, UK, 1994, pp. 133-136.
- [6] S. Pumrin and D. J. Dailey, "Roadside camera motion detection for automated speed measurement," in *IEEE 5th International Conference on Intelligent Transportation Systems*, 2002, pp. 147-151.
- [7] W. Lee and B. Ran, "Bidirectional roadway detection for traffic surveillance using online CCTV videos," in *IEEE International Conference on Intelligent Transportation Systems Conference*, Toronto, Ont., 2006, pp. 1556-1561.
- [8] A. Yilmaz, O. Javed, and M. Shah, "Object Tracking: A Survey," *Journal of ACM Computing Surveys*, vol. 38, no. 4, 2006.
- [9] M. Piccardi, "Background subtraction techniques: a review," in *IEEE International Conference on Systems, Man and Cybernetics*, 2004, vol. 4, pp. 3099-3104.
- [10] B. P. L. Lo and S. A. Velastin, "Automatic congestion detection system for underground platforms," in *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing*, Hong Kong, China, 2001, pp. 158-161.
- [11] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting Moving Objects, Ghosts, and Shadows in Video Streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1337-1342, 2003.
- [12] B. Chen, Y. Lei, and W. Li, "A novel background model for real-time vehicle detection," in *7th International Conference on Signal Processing*, 2004, vol. 2,

pp. 1276- 1279.

- [13] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Real-time tracking of the human body," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 1996, pp. 51-56.
- [14] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO , USA, 1999, vol. 2.
- [15] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proceedings of the 6th European Conference on Computer Vision-Part II*, 2000, pp. 751-767.
- [16] S. Zhang, H. Yao, and S. Liu, "Spatial-temporal nonparametric background subtraction in dynamic scenes," in *IEEE International Conference on Multimedia and Expo*, New York, NY, USA, 2009, pp. 518-521.
- [17] O. Javed and M. Shah, "Tracking and Object Classification for Automated Surveillance," in *Proceedings of the 7th European Conference on Computer Vision-Part IV*, London, UK, 2002, pp. 343-357.
- [18] J. M. Wang, Y. C. Chung, C. L. Chang, and S. W. Chen, "Shadow detection and removal for traffic images," in *IEEE International Conference on Networking, Sensing and Control*, 2004, vol. 1, pp. 649-654.
- [19] X. Song, Y. Ding, J. Gen, and Y. Chen, "Shadow Removal of Vehicles in a Video System Based on RGB Chroma Model," in *International Conference on Computer Science and Software Engineering*, Wuhan, China, 2008, pp. 977-980.
- [20] H. Liu, J. Li, Q. Liu, and Y. Qian, "Shadow Elimination in Traffic Video Segmentation," in *IAPR Conference on Machine Vision Applications*, Tokyo, 2007.
- [21] A. Yoneyama, C.-H. Yeh, and C. C. J. Kuo, "Robust Vehicle and Traffic Information Extraction for Highway Surveillance," *EURASIP Journal on Advances in Signal Processing*, no. 14, pp. 2305-2321, 2005.
- [22] S.-Y. Chien, Y.-W. Huang, B.-Y. Hsieh, S.-Y. Ma, and L.-G. Chen, "Fast video segmentation algorithm with shadow cancellation, global motion compensation, and adaptive threshold techniques," *IEEE Transactions on Multimedia*, vol. 6, no. 5, pp. 732-748, Oct. 2004.
- [23] J.-W. Hsieh, S.-H. Yu, Y.-S. Chen, and W.-F. Hu, "A Shadow Elimination Method for Vehicle Analysis," in *International Conference on Pattern Recognition*, Washington, DC, USA, 2004, pp. 372-375.
- [24] L. Xie, G. Zhu, Y. Wang, H. Xu, and Z. Zhang, "Robust vehicle extraction in video-based intelligent transportation systems," in *Visual Communications and Image Processing*, 2005.

- [25] T. J. Broida and R. Chellappa, "Estimation of object motion parameters from noisy images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 90–99, Jan. 1986.
- [26] G. Kitagawa, "Non-Gaussian State-Space Modeling of Nonstationary Time Series," *Journal of the American Statistical Association*, vol. 82, no. 400, pp. 1032-1041, Dec. 1987.
- [27] X. Chen and C. Zhang, "An Interactive Semantic Video Mining and Retrieval Platform—Application in Transportation Surveillance Video for Incident Detection," in *Proceedings of the Sixth International Conference on Data Mining*, Washington, DC, USA, 2006, pp. 129–138.
- [28] Y. Zou, G. Shi, H. Shi, and Y. Wang, "Image Sequences Based Traffic Incident Detection for Signaled Intersections Using HMM," in *International Conference on Hybrid Intelligent Systems*, Shenyang, China, 2009, vol. 1, pp. 257-261.
- [29] Y. Yue, "A Traffic-Flow Parameters Evaluation Approach Based on Urban Road Video," *International Journal of Intelligent Engineering and System*, vol. 2, 2009.
- [30] L. Li, L. Chen, X. Huang, and J. Huang, "A Traffic Congestion Estimation Approach from Video Using Time-Spatial Imagery," in *International Conference on Intelligent Networks and Intelligent Systems*, Wuhan, China, 2008, pp. 465-469.
- [31] T. B. L., C.-Y. Lin, and S. J.R., "Real-time video surveillance for traffic monitoring using virtual line analysis," *IEEE International Conference on Multimedia and Expo*, vol. 2, pp. 541- 544, 2002.
- [32] Liu Anan, "Video Vehicle Detection Algorithm based on Virtual-Line Group," in *IEEE Asia Pacific Conference on Circuits and Systems*, Singapore, 2006, pp. 1148-1151.
- [33] J. Wu, Z. Yang, J. Wu, and A. Liu, "Virtual Line Group Based Video Vehicle Detection Algorithm Utilizing Both Luminance and Chrominance," in *IEEE Conference on Industrial Electronics and Applications*, Harbin, China, 2007, pp. 2854-2858.
- [34] M. S. Sayed and J. Delva, "Low Complexity Contrast Enhancement Algorithm for Nighttime Visual Surveillance," in *International Conference on Intelligent Systems Design and Applications*, Cairo, 2010.
- [35] Y. Cai, K. Huang, T. Tan, and Y. Wang, "Context Enhancement of Nighttime Surveillance by Image Fusion," in *International Conference on Pattern Recognition*, Hong Kong, 2006, vol. 1, pp. 980-983.
- [36] A. Yamasaki, T. Hidenori, S. Kaneko, T. Kanade, and H. Ohki, "Denighting: Enhancement of nighttime images for a surveillance camera," in *International*

- Conference on Pattern Recognition*, Tampa, FL, 2008, pp. 1-4.
- [37] K. Huang, L. Wang, T. Tan, and S. Maybank, "A real-time object detecting and tracking system for outdoor night surveillance," *Pattern Recognition*, vol. 41, no. 1, pp. 432-444, Jan. 2008.
- [38] L. Wang, K. Huang, Y. Huang, and T. Tan, "Object detection and tracking for night surveillance based on salient contrast analysis," in *IEEE International Conference on Image Processing*, Cairo, Egypt, 2009, pp. 1113-1116.
- [39] K. Robert, "Night-Time Traffic Surveillance: A Robust Framework for Multi-vehicle Detection, Classification and Tracking," in *IEEE International Conference on Advanced Video and Signal Based Surveillance*, Genova, Italy, 2009, pp. 1-6.
- [40] Y.-L. Chen, B.-F. Wu, and C.-J. Fan, "Real-time vision-based multiple vehicle detection and tracking for nighttime traffic surveillance," in *IEEE International Conference on Systems, Man and Cybernetics*, San Antonio, TX, 2009, pp. 3352-3358.
- [41] Z.-N. and D., Mark S. Li, *Fundamentals of Multimedia*. Prentice-Hall of India, 2005.
- [42] S. Theodoridis and K. Koutroumbas, *Pattern Recognition & Matlab Intro: Pattern Recognition, Fourth Edition*, 4th ed. Academic Press, 2008.
- [43] Z.-H. Lee, "Human Behavior Analysis with Shadow Cancellation and Occlusion Detection Technique," in *National Chiao Tung University Master Thesis*, 2006.
- [44] "Taiwan Area National Freeway Bureau," <http://www.freeway.gov.tw/>, 04-Jul-2008.
- [45] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, Sep. 1995.