國立交通大學

電機與控制工程學系

碩 士 論 文

基於電腦視覺之即時穩健的
泛型障礙物與車道偵測行車系統

A Real-Time Robust On-Vehicle
Generic Obstacle and Lane Detection System
Based on Computer Vision Technique

研 究 生：賴則全

指導教授：吳炳飛　教授

中 華 民 國 九 十 四 年 七 月

基於電腦視覺之即時穩健的泛型障礙物與車道偵測行車系統
A Real-Time Robust On-Vehicle Generic Obstacle and Lane Detection System Based on Computer Vision Technique

研 究 生：賴則全   Student：Tze-Chiuan Lai

指導教授：吳炳飛   Advisor：Bing-Fei Wu

國 立 交 通 大 學

電 機 與 控 制 工 程 學 系

碩 士 論 文

A Thesis

Submitted to Department of Electronic and Control Engineering

College of Electrical Engineering and Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Electronic and Control Engineering

July 2005

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 四 年 七 月

基於電腦視覺之即時穩健的泛型障礙物與車道偵測行車系統

學生：賴則全　　　　　　　　　　　　　指導教授：吳炳飛　博士

國立交通大學電機與控制工程學系　碩士班

摘　　　　要

　　近年來隨著交通問題日益嚴重,智慧型運輸系統(Intelligent transportation system, ITS)的相關研究愈來愈受到重視,其中智慧型車輛又是最有發展潛力的研究之一。而泛型障礙物與車道偵測系統是智慧車所需配備的最基本功能,能夠偵測出路面障礙物的位置與車道資訊,用以預警駕駛人注意或者提供車輛自動行駛所必需的道路資訊。

　　本文主要是利用影像處理與電腦視覺的技術去偵測路上障礙物與車道的位置。將兩支單色 CCD 攝影機分別上下地架設在車上,利用 histogram-based 的方法將上方攝影機所擷取出的道路影像做分類,以偵測出不同類別的交界所組成之近乎水平的邊線。所偵測出的邊線可能位於地面或者障礙物上,這兩種情況判斷的依據是藉由立體視覺的技術分別預估此邊線在下方影像中可能是地面的位置以及可能是障礙物的位置,然後量測與上方影像中之邊線的相關係數何者比較大來做判斷,因此可以鑑別出影像中障礙物與路面的部分。

　　而在車道偵測方面,使用一支單色 CCD 攝影機擷取道路影像,以偵測車道標線的位置。本文所發展出的車道偵測演算法是基於車道幾何模型的標線偵測方式,能夠提供一個穩健的偵測結果,並且適當地預估與縮小搜尋範圍,以降低搜尋時間而提高車道偵測的效率。最後並重建車道 3-D 幾何模型以修正道路傾斜度與寬度,因此本文所提出的演算法亦適用於非平坦的路面。

本文所發展出的車道偵測系統已經在快速道路與高速公路成功地實車驗證過，在 2.6 GHz 的 PC 平台上平均每張影像所需的偵測時間小於 1 ms。此外，車道偵測系統亦結合方向盤控制器，做為無人駕駛智慧車的視覺系統，完成台灣第一台可以 hand-free 自動駕駛的智慧車 TAIWAN *i*TS-1。TAIWAN *i*TS-1 以時速 90 km/hr 與 110 km/hr 分別在東西向快速道路與國道 3 號高速公路順利地自動駕駛實車測試，並經過國外卓越計畫評審委員的評鑑與肯定，驗證了本文所提出的車道偵測系統的實用性與穩定性。

A Real-Time Robust On-Vehicle Generic Obstacle and Lane Detection System Based on Computer Vision Technique

student：Tze-Chiuan Lai   Advisor：Dr. Bing-Fei Wu

Department of Electronic and Control Engineering
National Chiao Tung University

ABSTRACT

As the traffic is becoming more and more serious in most developed countries, a lot of researches about the intelligent transportation system (ITS) have been paid attention in recent years. Above all, one of the most promoting topics for the ITS applications is concerning the smart vehicles. The fundamental function of the smart vehicle is the generic obstacle and lane detection system, which can warn the driver or provide the road information for the unmanned vehicle.

In this thesis the techniques of image processing and computer vision are applied to the detection system. Two monochromatic CCD cameras are mounted top and bottom on the vehicle, and the road image captured by the top camera is segmented by thresholding the histogram. After that, the quasi-horizontal boundaries formed by the interconnection of two different segments are detected in order, and each detected boundary could belong to either the ground or the obstacle. The criterion to distinguish between them is to predict the corresponding ground and obstacle boundaries in the bottom image by the stereo vision, and to compute the normalized correlation coefficients of the detected boundary in the top image with respect to the ground and obstacle boundaries in the bottom image respectively. The detected boundary in the top image belongs to the obstacle if the normalized correlation coefficient associated with the obstacle is larger than that associated with the ground. Thus the road image can be divided into the ground and obstacle parts.

On the other hand, a single monochromatic CCD camera is used in the lane detection system to detect the lane markings. Based on the geometric lane model, the algorithm of lane detection proposed in this thesis can generate a robust result. Besides, the detection region of interest can be estimated to narrow the searching area and to reduce the computational load. Eventually, the 3-D lane geometry is reconstructed to update the road inclination and lane width. Therefore the proposed algorithm is available in the case of non-flat roads.

The lane detection system proposed in this thesis has been successfully verified on the expressway and freeway. On the PC platform of 2.6-GHz CPU and 512-MB RAM, the average time of lane detection is less than 1 ms per frame. In addition, the lane detection system can be treated as the vision system of the automatic vehicle by integrating the controller of the steering wheel. This work has been implemented on the experimental car, TAIWAN iTS-1, running on the expressway and freeway with velocities of 90 km/hr and 110 km/hr respectively. TAIWAN iTS-1 is the first smart car in Taiwan capable of hand-free driving on the real road, which verifies the practicability and robustness of the proposed lane detection system.

Contents

CONTENTS

List of Figures

LIST OF FIGURES

List of Tables

Chapter 1 Introduction

1.1 Motivation

As the conveyances are getting growth with years, the traffic is becoming more and more serious in most developed countries. A lot of researches about the intelligent transportation systems (ITS), including the smart vehicles, the driving safety, and the traffic mobility, have been proposed in recent years. In fact, many problems are still expected to be overcome. Above all, one of the most interesting and important issues for the ITS applications is concerning the smart vehicles.

It is necessary to acquire the information about the on-road obstacles and the lane tendency while driving on the way. Thanks to the driver's careless attitude, his/her moving vehicle may hit the obstacles on the road, or may deviate from the correct lane orientation, which induces the traffic accidents. Hence the on-vehicle obstacle and lane detection system plays a fundamental and essential role in moving vehicles. Such a system can either be the driver assistance function to warn the drivers of occurrences of which they may not be aware, or be the vision system of unmanned vehicles to supply the car controller with the road information for the goal of the automatic driving.

In general, the vision-based obstacle and lane detection system is a good choice for ITS applications. Cameras are mounted on the vehicle, and then the road images are captured and processed. The systems based on the vision have advantages of the high spatial resolution and the fast image scansion. Many approaches using the image processing have been developed [1], and different techniques will be reviewed in the next section.

CHAPTER 1 INTRODUCTION

1.2 Background

1.2.1 Related work of obstacle detection

The definition of obstacles induces the development of detection algorithms. Since the vehicles are most of obstacles on the road, some approaches to detect obstacles are limited to search for particular features and then to match them with specific patterns, such as the symmetry, textures, shapes, an approximate contour, and so on. In this case the processing can be focused on the analysis of a single still image. Broggi et al. perform a function of vehicle detection to locate and track the vehicle by exploiting the symmetry of the rear parts of a typical car and a bounding box satisfying specific aspect ratio constraints [2]. However, such a pattern-based approach may fail when characteristics of obstacles do not match the pre-defined model.

As we know, vehicles are not the only obstacles on the road. A generic obstacle is defined as an object rising out significantly from the road surface. Following this definition, the pattern-based approach does not work owing to the lack of a prior knowledge about generic obstacles on the road. More complex techniques must be imported to handle such a problem, and two and more images may need to be taken into account.

The optical flow-based approach utilizes a sequence of two or more images to obtain reliable and dense optical flows. In the assumption of the small difference between two successive images due to the short time interval, the two-dimensional motion between two images approximates the single direction. And therefore, the optical flow field can be computed and the ego-motion can be estimated. Giachetti et al. use a correlation technique to compute the flow field, and the obstacles moving with different speeds can be segmented by analyzing the velocity fields [3]. However, the optical flow-based approach may fail deriving from the lack of textures on the road, or from large displacements between two consecutive

CHAPTER 1 INTRODUCTION

frames due to the higher speed or vibrations of the vehicle.

Another technique similar to the optical flow-based approach is known as the motion-based method by estimating the motion of the ground plane and then detecting the obstacles whose motions differ from that of the ground [4-7]. In this method, it is necessary to make a tracking about the motion among images for large displacements, and as a consequence the assumption of rectilinear motions in optical flow-based methods is invalid. Since the scenes vary very much among images, it is difficult to identify the pixel correspondence. If the size of searching area is too small, the correct matching for the corresponding pixels may be missed. On the other hand, if the size is too large, too many possibilities may exist. Notice that both optical flow-based and motion-based approaches need expensive computational costs.

The stereo vision-based technique is also used to detect the generic obstacles. The GOLD system transforms both left and right stereo images into top views in order to remove the perspective effect. The ideal square obstacle is transformed into two triangles in the difference image of both remapped views. The polar histogram is constructed from the difference image and then the two peaks in the polar histogram are joined to identify the obstacle [2, 8].

Labayrade et al. also use both left and right stereo images to construct v-disparity image to detect potential obstacles whose disparities differ from that on the road surface. The angles between the cameras and the road are then estimated [9-10]. In conclusion the stereo vision-based method is a better framework than others, and is adopted in this thesis.

1.2.2 Related work of lane detection

It is the objective for lane detection to detect the relative position between the vehicle and the road, and to determine the lane information, such as the offset, the orientation, the curvature, and so forth. Since the structured roads are met in the practical applications, most researches focus on the analysis of marking roads where lane markings are painted on the road surface. Several features of the lane markings, including the constant lane width, the higher brightness on markings, the structured lane shape, etc.

The GOLD system removes the perspective effect by mapping the road image into the top view, and determines the lane markings by relying on the feature of the constant lane width, which may fail when the assumption of a flat road is not valid [8]. Based on the GOLD system, Jiang et al. model the lane as two straight lines to estimate the inclined angle on the condition of non-flat roads [11].

However, the road shape usually is not straight in real cases. Polynomials or splines may be better lane fittings than the straight line. Such a geometric model-based lane detection technique is more robust against the interferences such as shadows, textures, or other vehicles. Based on the lane geometry, the coefficients of lane model can be found out by several methods. LOIS, LANA, and RVP-I systems decide the coefficients with the maximum likelihood by completely searching the parameter spaces where all possibilities produced in the training phase are built [12-14].

Instead of searching throughout the databases, some road features are detected in the overall image in order to determine the coefficients. Yue Wang et al. [15] and Goldbeck et al. [16] use the edge-oriented methods to measure the matching degree between the model and the edge map in order to determine the parameters, respectively. Gonzalez et al. classify the objects in the image as the road surface, markings, or obstacles by a histogram-based segmentation method and then pixels belonging to the markings are taken into the fitting of

the lane model [17].

Different from the lane geometry, the statistical model can be used to specify the detection region of interest (ROI) in order to narrow the searching area [7, 18]. On the other hand, the ROI can also be determined according to the features of markings used in the TFALDA [19]. However, the statistical parameters and the weights of marking features have to be trained in advance.

Since the model-based approaches have more robust results and the use of the detection ROI can reduce the computational cost, both ideas are adopted in this thesis. The details will be proposed later.

1.3 Organization

This thesis is organized as follows. A review of algorithms about the obstacle and lane detections is given in this chapter. The preliminary knowledge of the computer vision is introduced in Chapter 2. In Chapter 3, the algorithm of the generic obstacle detection based on two top and bottom stereo cameras is developed. The approach to detect the lane is proposed in Chapter 4. And afterward the experimental results of both obstacle and lane detections are demonstrated in Chapter 5. Finally, a conclusion is presented in Chapter 6.
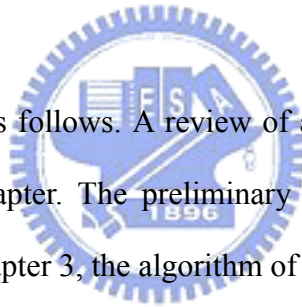
Chapter 2 Stereo Vision System

In this chapter the preliminary knowledge of the computer vision will be introduced. In the beginning, the relationship of image, camera, and world coordinate systems is discussed. And then the surface of the non-flat road is modeled. The architecture of two top and bottom stereo cameras and the calibration principle are presented finally.

2.1 Geometric Camera Model

2.1.1 Perspective projection

The scene points (X_c, Y_c, Z_c) in the camera coordinate system can be captured by a camera and be projected onto the image pixels (u, v) in the image coordinate system, as illustrated in Fig. 2.1. This phenomenon can be described as the perspective projection, and the camera can be modeled quite well by the so-called ideal pinhole camera, which induces the projection equations as follows [20]:

$$u = e_u \frac{X_c}{Y_c} \tag{2-1}$$

and
$$v = e_v \frac{Z_c}{Y_c} \tag{2-2}$$

where (u, v) and (X_c, Y_c, Z_c) are the image and camera coordinates, respectively. Note that e_u and e_v are the intrinsic parameters of the camera, and are represented by:

$$e_u = \frac{f}{du} \quad \text{and} \quad e_v = \frac{f}{dv} \tag{2-3}$$

where du and dv are the physical width and height, respectively, of an image pixel. And f is the focal length of the camera.

(2-1) and (2-2) are the non-linear equations transforming the scene points of $\mathbf{R}^3$ into the

image pixels of $\mathbf{R}^2$, and therefore, the homogeneous coordinate system is suitable for a general simple treatment of the perspective projection.
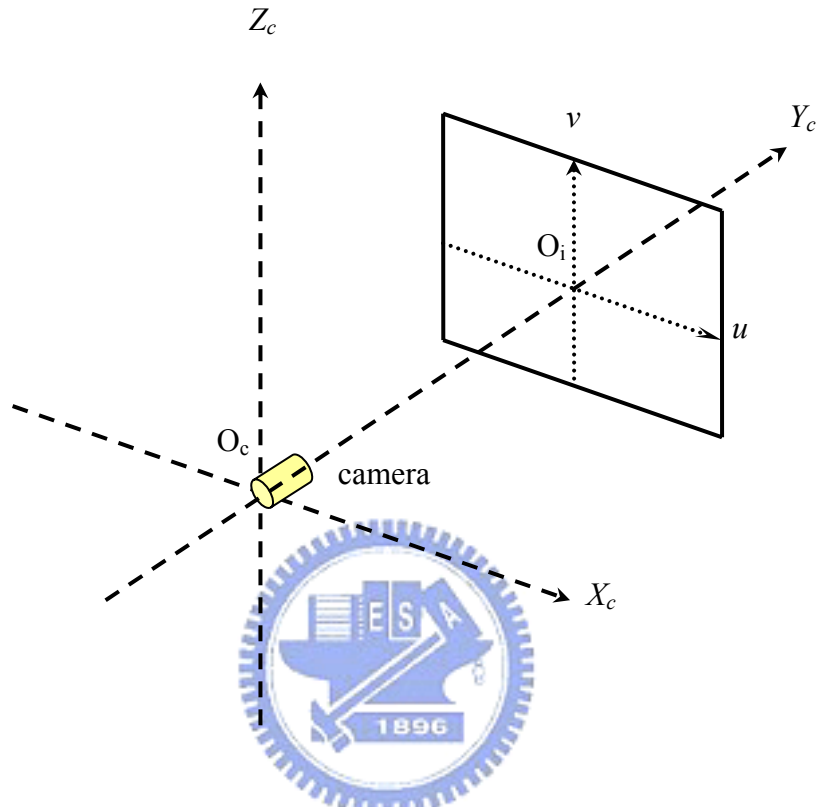
Fig. 2.1   The relationship of the camera and image coordinate systems.

Let $\mathbf{P}_{proj}$ is the 4×4 perspective transform matrix, expressed as follows:

$$\mathbf{P}_{proj} = \begin{bmatrix} e_u & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e_v & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2-4}$$

transforming $\bar{C}_h = \begin{bmatrix} X_c & Y_c & Z_c & 1 \end{bmatrix}'$ into $\bar{I}_h = \begin{bmatrix} x_i & y_i & z_i & 1 \end{bmatrix}'$, i.e.

$$\bar{I}_h = \mathbf{P}_{proj} \bar{C}_h \tag{2-5}$$

where $\bar{C}_h$ and $\bar{I}_h$ are the homogeneous camera and image coordinates, respectively. Notice that the prime denotes the transpose.

The non-homogeneous image coordinates (u, v) can then be obtained by

$$u = \frac{x_i}{y_i} \tag{2-6}$$

and

$$v = \frac{z_i}{y_i}. \tag{2-7}$$

It is obvious that there exists the invertible matrix $\mathbf{P}_{proj}^{-1}$ satisfying $\bar{C}_h = \mathbf{P}_{proj}^{-1} \bar{I}_h$, and however, there is no sufficient information from the image coordinates (u, v) of $\mathbf{R}^2$ to get the camera coordinates (X_c, Y_c, Z_c) of $\mathbf{R}^3$. The solutions about this issue will be presented later.

2.1.2 Point relationship of camera and world coordinates

In the following, it is necessary to perform positioning in three coordinate systems, $\bar{I}_h$, $\bar{C}_h$, and $\bar{W}_h$, shown in Fig. 2.1 and Fig. 2.2. The relationship of $\bar{I}_h$ and $\bar{C}_h$, homogeneous image and camera coordinates, respectively, have been explained in Section 2.1.1. In this section the point relationship of $\bar{C}_h$ and $\bar{W}_h$ will be discussed, and the transformation from $\bar{W}_h$ to $\bar{I}_h$ will be briefed in Section 2.1.3.
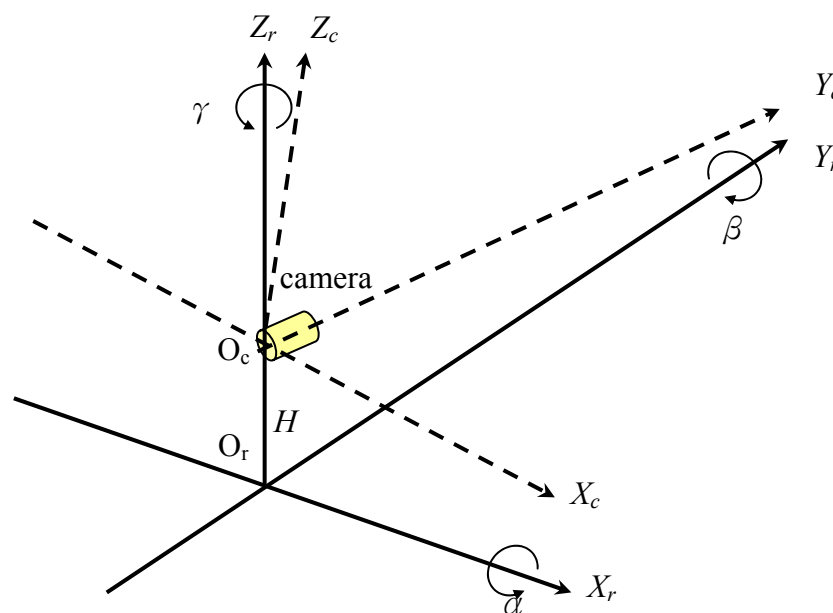
Fig. 2.2   The relationship of the camera and world coordinate systems.

The 4×1 vector $\vec{W}_h = \begin{bmatrix} X_r & Y_r & Z_r & 1 \end{bmatrix}'$ is the homogeneous world coordinates, associated with $\vec{C}_h$ by

$$\vec{C}_h = \mathbf{R}_r \vec{W}_h - \vec{T} \qquad (2\text{-}8)$$

where $\mathbf{R}_r$ is the 4×4 rotation matrix between the camera and the road, and $\vec{T}$ is the 4×1 translation vector from O_r to O_c, the origins of world and camera coordinate systems, respectively.

Often $\vec{T}$ is the 1-D translation from O_r to O_c, expressed by:

$$\vec{T} = \begin{bmatrix} 0 & 0 & H & 0 \end{bmatrix}' \qquad (2\text{-}9)$$

and H is the distance between O_r and O_c.

In general, $\mathbf{R}_r$ is however, composed of three 4×4 rotation matrices, $\mathbf{R}_\alpha$, $\mathbf{R}_\beta$, and $\mathbf{R}_\gamma$, i.e.
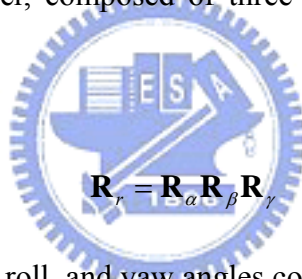
$$\mathbf{R}_r = \mathbf{R}_\alpha \mathbf{R}_\beta \mathbf{R}_\gamma \qquad (2\text{-}10)$$

where α, β, and γ are the pitch, roll, and yaw angles counterclockwise looking at the origin O_r from $+X_r$, $+Y_r$, and $+Z_r$ axes, respectively, and

$$\mathbf{R}_\alpha = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2\text{-}11)$$

$$\mathbf{R}_\beta = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2\text{-}12)$$

$$\mathbf{R}_\gamma = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2\text{-}13)$$

Usually the yaw angle, γ, can be taken no account without relating to the lane orientation on the road and can be withdrawn. Hence (2-10) can be replaced by:

$$\mathbf{R}_r = \mathbf{R}_\alpha \mathbf{R}_\beta = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ \sin\alpha \cdot \sin\beta & \cos\alpha & -\sin\alpha \cdot \cos\beta & 0 \\ -\cos\alpha \cdot \sin\beta & \sin\alpha & \cos\alpha \cdot \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2-14}$$

and its inverse matrix is

$$\mathbf{R}_r^{-1} = \mathbf{R}_\beta \mathbf{R}_\alpha = \begin{bmatrix} \cos\beta & \sin\alpha \cdot \sin\beta & -\cos\alpha \cdot \sin\beta & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ \sin\beta & -\sin\alpha \cdot \cos\beta & \cos\alpha \cdot \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2-15}$$

(2-8) can be rewritten as:

$$\vec{W}_h = \mathbf{R}_r^{-1}\left(\vec{C}_h + \vec{T}\right) \tag{2-16}$$

Substituting each term into (2-16) yields

$$X_r = \cos\beta \cdot X_c + \sin\alpha \cdot \sin\beta \cdot Y_c - \cos\alpha \cdot \sin\beta \cdot (Z_c + H), \tag{2-17}$$

$$Y_r = \cos\alpha \cdot Y_c + \sin\alpha \cdot (Z_c + H), \tag{2-18}$$

and $\qquad Z_r = \sin\beta \cdot X_c - \sin\alpha \cdot \cos\beta \cdot Y_c + \cos\alpha \cdot \cos\beta \cdot (Z_c + H), \qquad$ (2-19)

which transform a point from (X_c, Y_c, Z_c) in the camera coordinates to (X_r, Y_r, Z_r) in the world coordinates.

2.1.3 Point relationship of image and world coordinates

By the combination of (2-5) and (2-8), it yields

$$\bar{I}_h = \mathbf{P}_{proj}\bar{C}_h$$
$$= \mathbf{P}_{proj}\left(\mathbf{R}_r\bar{W}_h - \bar{T}\right) \tag{2-20}$$
$$= \begin{bmatrix} e_u & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e_v & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta \cdot X_r + \sin\beta \cdot Z_r \\ \sin\alpha \cdot \sin\beta \cdot X_r + \cos\alpha \cdot Y_r - \sin\alpha \cdot \cos\beta \cdot Z_r \\ -\cos\alpha \cdot \sin\beta \cdot X_r + \sin\alpha \cdot Y_r + \cos\alpha \cdot \cos\beta \cdot Z_r - H \\ 1 \end{bmatrix}$$

The transformation from the point (X_r, Y_r, Z_r) in the world coordinates to the pixel (u, v) in the image coordinates can be described by

$$u = \frac{x_i}{y_i} = e_u \frac{\cos\beta \cdot X_r + \sin\beta \cdot Z_r}{\sin\alpha \cdot \sin\beta \cdot X_r + \cos\alpha \cdot Y_r - \sin\alpha \cdot \cos\beta \cdot Z_r} \tag{2-21}$$

and
$$v = \frac{z_i}{y_i} = e_v \frac{-\cos\alpha \cdot \sin\beta \cdot X_r + \sin\alpha \cdot Y_r + \cos\alpha \cdot \cos\beta \cdot Z_r - H}{\sin\alpha \cdot \sin\beta \cdot X_r + \cos\alpha \cdot Y_r - \sin\alpha \cdot \cos\beta \cdot Z_r}. \tag{2-22}$$

If the roll angle, β, of the camera, approximates to zero, then

$$u = e_u \frac{X_r}{\cos\alpha \cdot Y_r - \sin\alpha \cdot Z_r} \tag{2-23}$$

and
$$v = e_v \frac{\sin\alpha \cdot Y_r + \cos\alpha \cdot Z_r - H}{\cos\alpha \cdot Y_r - \sin\alpha \cdot Z_r}. \tag{2-24}$$

If $\alpha = 0$ and $\beta = 0$, i.e. no rotation occurs between both coordinate systems, then (2-21) and (2-22) reduce to

$$u = e_u \frac{X_r}{Y_r} \tag{2-25}$$

and
$$v = e_v \frac{Z_r - H}{Y_r}. \tag{2-26}$$

2.2 Modeling the Road Surface

2.2.1 Consideration for the angle of inclination on the non-flat road

Usually the surface on the real road is not flat, and it may be modeled as a succession composed of piecewise planes. For the simplification and the practicality, the road surface in this thesis is modeled as the plane with the inclined angle θ, formed by the road ground and the plane where the vehicle mounted the camera is standing, see Fig. 2.3.
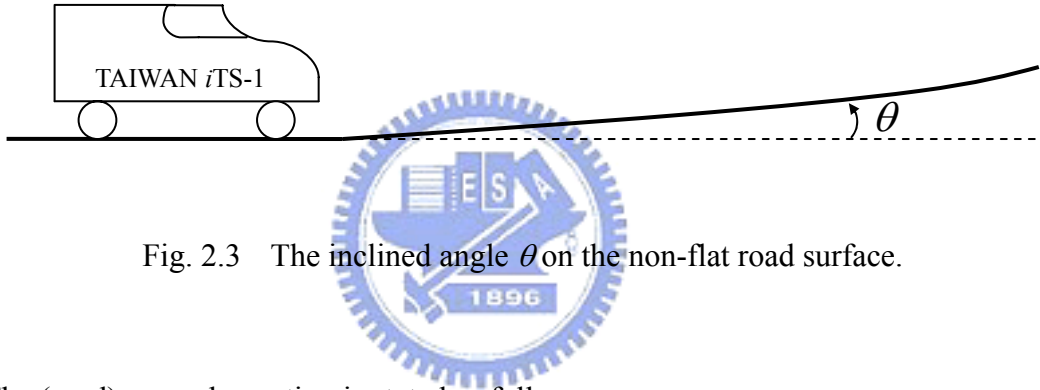
Fig. 2.3 The inclined angle θ on the non-flat road surface.

The (road) ground equation is stated as follows:

$$Z_r = \tan\theta \cdot Y_r = m_\theta \cdot Y_r, \tag{2-27}$$

where $m_\theta = \tan\theta$ is the road inclination. Assume that both α and β approximate to zero, and combine (2-25), (2-26), and (2-27) to produce

$$X_r = \frac{u \cdot H}{e_v \cdot m_\theta - v} \cdot \frac{e_v}{e_u} \tag{2-28}$$

$$Y_r = e_v \frac{H}{e_v \cdot m_\theta - v} \tag{2-29}$$

$$Z_r = e_v \frac{m_\theta \cdot H}{e_v \cdot m_\theta - v}. \tag{2-30}$$

If the road inclination m_θ and the ground coordinates (u, v) in the image are given, the physical ground coordinates (X_r, Y_r, Z_r) can be estimated by (2-28), (2-29), and (2-30).

Notice that the intrinsic difference of the discussions between Section 2.1.3 and Section 2.2.1. Section 2.1.3 has proposed that a point in the world coordinate system can be projected onto the image plane, which is affected by the camera angles. It is never said that $Z_r = 0$ means the road surface. Ideally, $Z_r = 0$ may happen on the flat road. If the camera angles, i.e. α and β, are thought as what are included by the inclinations of the camera and the road, it can be true that $Z_r = 0$ for certain α and β in every local zone is exactly the road plane, and the angles for $Z_r = 0$ may be different zone by zone.

On the contrary, the inclined angle θ discussed in this section is a global consideration for the model of the non-flat road. It can be exactly said that (2-27) is representative of the road ground. Similarly, if the inclined angle θ is treated as what is included by the inclinations of the camera and the road, then the pitch angle α can never be considered. Furthermore, if β is small enough to be ignored, as assumed earlier, then three axes in the camera coordinates are coincided with those in the world coordinates.
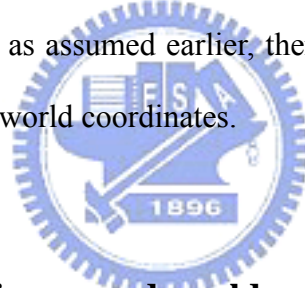
2.2.2 Width mapping of image and world coordinates

In this section we focus on the mapping of the width on the road ground from the world coordinates to the image coordinates. From (2-28), it is easy to get

$$\Delta X_r = \Delta u \cdot \frac{H}{e_v \cdot m_\theta - v} \cdot \frac{e_v}{e_u} \tag{2-31}$$

and

$$\Delta u = \Delta X_r \cdot \frac{e_v \cdot m_\theta - v}{H} \cdot \frac{e_u}{e_v} \tag{2-32}$$

If the road inclination m_θ and ΔX_r, such as the lane width, are given, the pixel distance, say Δu, of the abscissa in the image can be determined by (2-32).

2.2.3 Effects on distance accuracy associated with the inclined angle of the

road and the camera height

The distance of objects in front of the camera is a desire for the application of smart vehicles. As indicated in (2-29), the distance, Y_r, is associated with the camera height H and the inclined angle θ. However, H or θ may change due to the oscillation in motion or the non-flat road surface, which results in an inaccurate measure of distance. The effects on the distance by the camera height H and by the inclined angle θ will be discussed, respectively, and an ideal case is assumed that $H = 135$ cm and $\theta = 0°$.

➢   A: *the variation in the camera height H*

In this case θ is fixed and equals to zero. However, the change of $H \rightarrow H + \Delta H$ results in the change of $Y_r \rightarrow Y_r + \Delta Y_r$. From (2-29), a simple analysis can be to obtain the factor of variation $\left(\Delta Y_r / Y_r \right)_{\Delta H}$ as follows:

$$\left(\frac{\Delta Y_r}{Y_r} \right)_{\Delta H} = \frac{\Delta H}{H} \tag{2-33}$$

It is obvious that $\left(\Delta Y_r / Y_r \right)_{\Delta H}$ is only related to ΔH and is not affected by the distance Y_r. The change, ΔH, of the camera height due to the vibration in motion can be assumed to bound in ±20 cm. Table 2-1 shows some cases of different ΔH. For example, the maximum error of the distance on the condition of $Y_r = 50$ m and $\Delta H = 20$ cm is 7.4 m.

Table 2-1 The effect on the distance by the variation in the camera height H.

ΔH (cm)	± 5	± 10	± 15	± 20
$\left(\dfrac{\Delta Y_r}{Y_r} \right)_{\Delta H}$	± 0.037	± 0.074	± 0.111	± 0.148

➢ B: *the variation in the inclined angle θ*

In this case H is fixed and equals to 135 cm. The error ΔY_r derived from the change of $\theta \rightarrow \theta + \Delta\theta$ can also be computed by (2-29).

$$\Delta Y_r = e_v \frac{H}{e_v \cdot m_{\theta+\Delta\theta} - v} - Y_r = e_v \frac{H}{e_v \cdot m_{\theta+\Delta\theta} - e_v \cdot (m_\theta - H/Y_r)} - Y_r$$

$$= \frac{-m_{\theta+\Delta\theta} + m_\theta}{m_{\theta+\Delta\theta} - m_\theta + H/Y_r} \cdot Y_r = \frac{-m_{\Delta\theta}}{m_{\Delta\theta} + H/Y_r} \cdot Y_r \quad (\because \theta = 0°)$$

$$= \frac{1}{\dfrac{H}{m_{\Delta\theta} \cdot Y_r} - 1} \cdot Y_r \tag{2-34}$$

Furthermore, the factor of variation in θ is indicated by

$$\left(\frac{\Delta Y_r}{Y_r}\right)_{\Delta\theta} = \frac{1}{\dfrac{H}{m_{\Delta\theta} \cdot Y_r} - 1} \tag{2-35}$$

It is clear that $\left(\Delta Y_r/Y_r\right)_{\Delta\theta}$ is associated with not only $\Delta\theta$ but also the distance Y_r. An observation reveals that the farther the distance is, the larger the absolute value of $\left(\Delta Y_r/Y_r\right)_{\Delta\theta}$ is. Some cases of different $\Delta\theta$ are listed in Table 2-2. It is obvious that small changes in θ still affect the distance very much.

Table 2-2 The effect on the distance by the variation in the inclined angle θ.

$\left(\Delta Y_r/Y_r\right)_{\Delta\theta}$ $\Delta\theta$ / Y_r (m)	1°	2°	3°	-1°	-2°	-3°
10	0.148	0.349	0.635	-0.114	-0.205	-0.280
20	0.349	1.072	3.472	-0.205	-0.341	-0.437
30	0.634	3.465	-7.075	-0.280	-0.437	-0.538
40	1.071	-29.827	-2.809	-0.341	-0.509	-0.608
50	1.829	-4.409	-2.063	-0.392	-0.564	-0.659

According to the above discussions, a conclusion is given that ΔH can usually be ignored because the variation in θ has the dominant effect on the distance. Such a concept is used throughout this thesis.

2.3 Stereo Cameras

2.3.1 Relationship of main and sub stereo cameras

This thesis will propose the framework of both top and bottom stereo cameras, namely main and sub cameras, respectively, as illustrated in Fig. 2.4. The top camera is the main image sensor used for the obstacle and lane detection. The lower sub camera is the auxiliary utilized only to detect the obstacle.
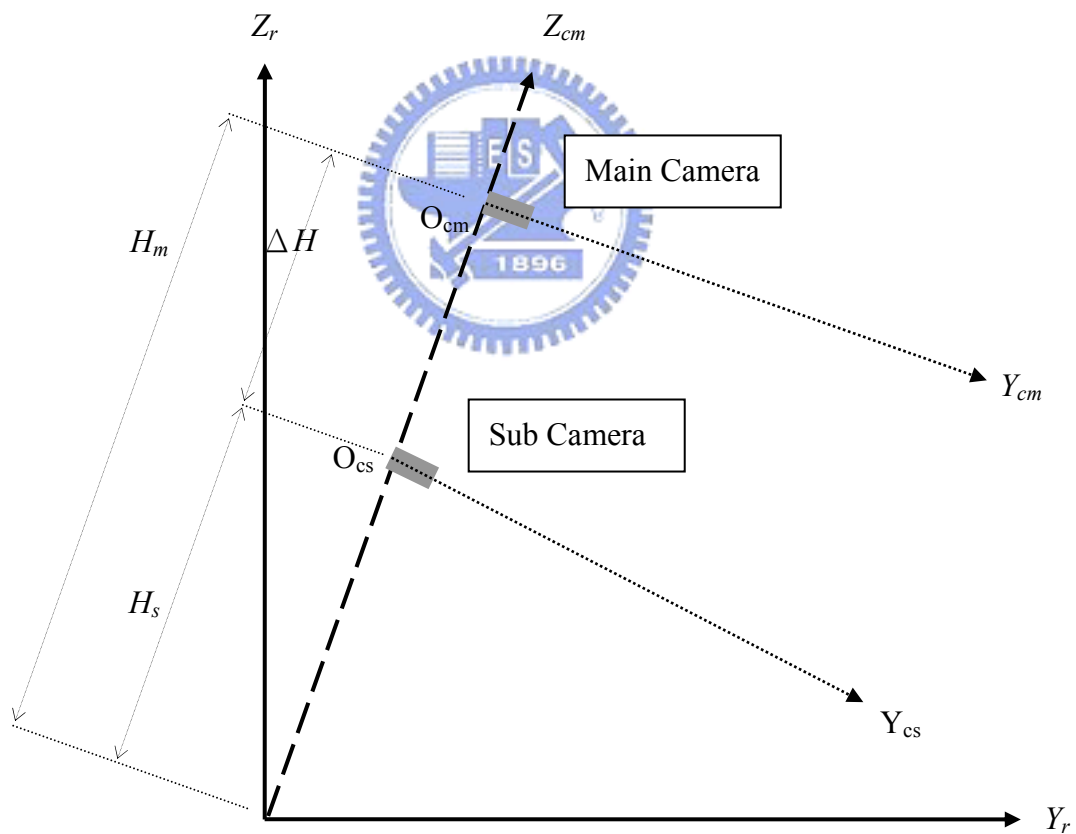
Fig. 2.4   The relationship of main and sub stereo cameras.

CHAPTER 2 STEREO VISION SYSTEM

The relationship of homogeneous image, camera, and world coordinate systems, namely $\vec{I}_{hm}$, $\vec{C}_{hm}$, and $\vec{W}_{hm}$, respectively, of the main image sensor is the same as what is described in (2-5) and (2-8), that is,

$$\vec{I}_{hm} = \mathbf{P}_{proj}\vec{C}_{hm} \tag{2-36}$$

and

$$\vec{C}_{hm} = \mathbf{R}_r\vec{W}_{hm} - \vec{T}_m \tag{2-37}$$

where $\vec{T}_m = \begin{bmatrix} 0 & 0 & H_m & 0 \end{bmatrix}'$ is the 4×1 translation vector from O_r to O_{cm}, the origin of the main camera coordinate system, and H_m is the distance between O_r and O_{cm}.

The relationship of homogeneous main and sub coordinate systems is the combination of one translation and one rotation, which is expressed as follows:

$$\vec{C}_{hs} = \mathbf{R}_c\left[\vec{C}_{hm} + \vec{T}_m - \vec{T}_s\right] \tag{2-38}$$

where $\vec{C}_{hs} = \begin{bmatrix} X_{cs} & Y_{cs} & Z_{cs} & 1 \end{bmatrix}'$ is the 4×1 homogeneous sub camera coordinates, $\mathbf{R}_c$ is the 4×4 homogeneous rotation matrix between main and sub cameras, and $\vec{T}_s = \begin{bmatrix} 0 & 0 & H_s & 0 \end{bmatrix}'$ is the 4×1 translation vector from O_r to O_{cs}, the origin of the sub camera coordinate system. Notice that

$$\Delta H = H_m - H_S \tag{2-39}$$

is the distance between O_{cm} and O_{cs}.

Applying (2-36) through (2-38), the following equations are easily obtained

$$\vec{C}_{hs} = \mathbf{R}_c\left[\mathbf{R}_r\vec{W}_h - \vec{T}_s\right] \tag{2-40}$$

$$\vec{I}_{hs} = \mathbf{P}_{proj}\vec{C}_{hs} = \mathbf{P}_{proj}\mathbf{R}_c\left[\mathbf{R}_r\vec{W}_h - \vec{T}_s\right] \tag{2-41}$$

and

$$\mathbf{R}_c^{-1}\mathbf{P}_{proj}^{-1}\vec{I}_{hs} - \mathbf{P}_{proj}^{-1}\vec{I}_{hm} = \vec{T}_m - \vec{T}_s \tag{2-42}$$

2.3.2 Main camera coordinates from pixel correspondence of stereo images

Remember that $\mathbf{R}_c$ is the 4×4 homogeneous rotation matrix between main and sub camera, as introduced in Section 2.3.1. Due to the pitch , roll, and yaw angles between main and sub cameras, $\mathbf{R}_c$ is similar to (2-10). For the simplification and the convenience, $\mathbf{R}_c$ is denoted by:

$$\mathbf{R}_c = \begin{bmatrix} r_{00} & r_{01} & r_{02} & 0 \\ r_{10} & r_{11} & r_{12} & 0 \\ r_{20} & r_{21} & r_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2-43}$$

and we know $\mathbf{R}_c^{-1} = \mathbf{R}_c'$. Thus (2-38) can be expanded as:

$$\begin{bmatrix} X_{cs} \\ Y_{cs} \\ Z_{cs} \\ 1 \end{bmatrix} = \mathbf{R}_c \begin{bmatrix} X_{cm} \\ Y_{cm} \\ Z_{cm} + \Delta H \\ 1 \end{bmatrix} = \begin{bmatrix} r_{00} \cdot X_{cm} + r_{01} \cdot Y_{cm} + r_{02} \cdot (Z_{cm} + \Delta H) \\ r_{10} \cdot X_{cm} + r_{11} \cdot Y_{cm} + r_{12} \cdot (Z_{cm} + \Delta H) \\ r_{20} \cdot X_{cm} + r_{21} \cdot Y_{cm} + r_{22} \cdot (Z_{cm} + \Delta H) \\ 1 \end{bmatrix} \tag{2-44}$$

Furthermore, the following equations also hold true:

$$v_s = e_v \frac{Z_{cs}}{Y_{cs}} = e_v \frac{r_{20} \cdot X_{cm} + r_{21} \cdot Y_{cm} + r_{22} \cdot (Z_{cm} + \Delta H)}{r_{10} \cdot X_{cm} + r_{11} \cdot Y_{cm} + r_{12} \cdot (Z_{cm} + \Delta H)}, \tag{2-45}$$

$$v_m = e_v \frac{Z_{cm}}{Y_{cm}}, \tag{2-46}$$

and $$u_m = e_u \frac{X_{cm}}{Y_{cm}}. \tag{2-47}$$

The above three equations can be represented by the following linear algebraic system

$$\begin{bmatrix} r_{10} \cdot v_s - r_{20} \cdot e_v & r_{11} \cdot v_s - r_{21} \cdot e_v & r_{12} \cdot v_s - r_{22} \cdot e_v \\ 0 & v_m & -e_v \\ -e_u & u_m & 0 \end{bmatrix} \begin{bmatrix} X_{cm} \\ Y_{cm} \\ Z_{cm} \end{bmatrix} = \begin{bmatrix} -(r_{12} \cdot v_s - r_{22} \cdot e_v)\Delta H \\ 0 \\ 0 \end{bmatrix} \tag{2-48}$$

By Cramer's rule, the solution is given:

$$X_{cm} = \frac{\det X_{cm}}{\det A_{cm}} \tag{2-49}$$

$$Y_{cm} = \frac{\det Y_{cm}}{\det A_{cm}} \tag{2-50}$$

$$Z_{cm} = \frac{\det Z_{cm}}{\det A_{cm}} \tag{2-51}$$

where $\det A_{cm} = e_u \left\{ v_m \left(r_{12} \cdot v_s - r_{22} \cdot e_v \right) + u_m \left(r_{10} \cdot v_s - r_{20} \cdot e_v \right) \frac{e_v}{e_u} + \left(r_{11} \cdot v_s - r_{21} \cdot e_v \right) e_v \right\}$ (2-52)

$$\det X_{cm} = -u_m \cdot \left(r_{12} \cdot v_s - r_{22} \cdot e_v \right) \cdot \Delta H \cdot e_v \tag{2-53}$$

$$\det Y_{cm} = - \quad \left(r_{12} \cdot v_s - r_{22} \cdot e_v \right) \cdot \Delta H \cdot e_v \cdot e_u \tag{2-54}$$

$$\det Z_{cm} = -v_m \cdot \left(r_{12} \cdot v_s - r_{22} \cdot e_v \right) \cdot \Delta H \cdot e_u \tag{2-55}$$

In summary it can be to obtain the main camera coordinates, X_{cm}, Y_{cm}, and Z_{cm}, on condition that the pixel correspondence of stereo image coordinates, u_m, v_m, and v_s, is given.

2.3.3 Pixel correspondence of stereo images

Multiplying both numerator and denominator of (2-45) by e_v / Y_{cm} together, another form is indicated as:

$$
\begin{aligned}
v_s &= e_v \frac{Z_{cs}}{Y_{cs}} = e_v \frac{r_{20} \cdot X_{cm} + r_{21} \cdot Y_{cm} + r_{22} \cdot (Z_{cm} + \Delta H)}{r_{10} \cdot X_{cm} + r_{11} \cdot Y_{cm} + r_{12} \cdot (Z_{cm} + \Delta H)} \\[2mm]
&= e_v \frac{r_{20} \cdot e_u \dfrac{X_{cm}}{Y_{cm}} \cdot \dfrac{e_v}{e_u} + r_{21} \cdot e_v + r_{22} \cdot \left(e_v \dfrac{Z_{cm}}{Y_{cm}} + e_v \dfrac{\Delta H}{Y_{cm}} \right)}{r_{10} \cdot e_u \dfrac{X_{cm}}{Y_{cm}} \cdot \dfrac{e_v}{e_u} + r_{11} \cdot e_v + r_{12} \cdot \left(e_v \dfrac{Z_{cm}}{Y_{cm}} + e_v \dfrac{\Delta H}{Y_{cm}} \right)} \\[2mm]
&= e_v \frac{r_{20} \cdot u_m \dfrac{e_v}{e_u} + r_{21} \cdot e_v + r_{22} \cdot \left(v_m + e_v \dfrac{\Delta H}{Y_{cm}} \right)}{r_{10} \cdot u_m \dfrac{e_v}{e_u} + r_{11} \cdot e_v + r_{12} \cdot \left(v_m + e_v \dfrac{\Delta H}{Y_{cm}} \right)}
\end{aligned}
\tag{2-56}
$$

In the same way, u_s is given

$$
u_s = e_u \frac{r_{00} \cdot u_m \dfrac{e_v}{e_u} + r_{01} \cdot e_v + r_{02} \cdot \left(v_m + e_v \dfrac{\Delta H}{Y_{cm}} \right)}{r_{10} \cdot u_m \dfrac{e_v}{e_u} + r_{11} \cdot e_v + r_{12} \cdot \left(v_m + e_v \dfrac{\Delta H}{Y_{cm}} \right)}
\tag{2-57}
$$

Define two functions of three variables as follows:

$$\text{PreDot}(m, n, o_i) \equiv o_{i0} \cdot m \frac{e_v}{e_u} + o_{i1} \cdot e_v + o_{i2} \cdot n \tag{2-58}$$

$$\text{PostDot}(m, n, o_i) \equiv o_{0i} \cdot m \frac{e_v}{e_u} + o_{1i} \cdot e_v + o_{2i} \cdot n \tag{2-59}$$

where $\text{PreDot}(m, n, o_i)$ is the inner product of $\begin{bmatrix} o_{i0} & o_{i1} & o_{i2} \end{bmatrix}'$ and $\begin{bmatrix} m \dfrac{e_v}{e_u} & e_v & n \end{bmatrix}'$.

Let
$$\Delta v_i \equiv e_v \frac{\Delta H}{Y_{ci}} \tag{2-60}$$

where the suffix i denotes the main or sub camera.

Hence (2-57) and (2-56) can be, respectively, described by:

$$u_s = e_u \frac{\text{PreDot}(u_m, v_m, r_0) + r_{02} \cdot \Delta v_m}{\text{PreDot}(u_m, v_m, r_1) + r_{12} \cdot \Delta v_m} \tag{2-61}$$

$$v_s = e_v \frac{\text{PreDot}(u_m, v_m, r_2) + r_{22} \cdot \Delta v_m}{\text{PreDot}(u_m, v_m, r_1) + r_{12} \cdot \Delta v_m}. \tag{2-62}$$

From the similar process, u_m and v_m can be obtained as follows:

$$u_m = e_u \frac{\text{PostDot}(v_s, u_s, r_0)}{\text{PostDot}(v_s, u_s, r_1)} \tag{2-63}$$

$$v_m = e_v \frac{\text{PostDot}(v_s, u_s, r_2) - \Delta v_s}{\text{PostDot}(v_s, u_s, r_1)} \tag{2-64}$$

where
$$\Delta v_s = \{ e_v \cdot \Delta v_m \} / \{ \text{PreDot}(u_m, v_m, r_1) + r_{12} \cdot \Delta v_m \} \tag{2-65}$$

➢ Part A: *the condition of $Z_r = 0$*

The relationship of Z_r and the main camera coordinates can be found in (2-19):

$$Z_r = \sin \beta \cdot X_{cm} - \sin \alpha \cdot \cos \beta \cdot Y_{cm} + \cos \alpha \cdot \cos \beta \cdot (Z_{cm} + H_m) \tag{2-66}$$

Instead of (2-66), (2-67) results from the fact of $Z_r = 0$ on the ground.

$$0 = \sin \beta \cdot X_{cmg} - \sin \alpha \cdot \cos \beta \cdot Y_{cmg} + \cos \alpha \cdot \cos \beta \cdot (Z_{cmg} + H_m) \tag{2-67}$$

where the suffix g indicates the ground of $Z_r = 0$.

By algebraic manipulations, Δv_{mg} can be determined, that is,

$$\Delta v_{mg} = e_v \frac{\Delta H}{Y_{cmg}} = -\frac{\Delta H}{H_m} \left\{ \left[\frac{\tan \beta}{\cos \alpha} \right] \cdot u_{mg} \frac{e_v}{e_u} + \left[-\tan \alpha \right] \cdot e_v + v_{mg} \right\} \tag{2-68}$$

To summarize, it is able to determine the sub image coordinates, u_{sg}, and v_{sg}, of the ground from (2-61) and (2-62) on condition that u_{mg}, v_{mg}, and Δv_{mg} are known.

CHAPTER 2 STEREO VISION SYSTEM

➢ Part B: *the condition of the same Y_r*

Given the pixel correspondence between both stereo images, i.e. P_{m1} (u_{m1}, v_{m1}) in the main image associated with P_{s1} (u_{s1}, v_{s1}) in the sub image, as illustrated in Fig. 2.5, if there exists P_{m2} (u_{m2}, v_{m2}) with Y_r as same as that of P_{m1}, the goal is to find out its corresponding P_{s2} (u_{s2}, v_{s2}) in the sub image.

Applying (2-18) to the equality of Y_r at P_{m1} and P_{m2} yields

$$\cos\alpha \cdot Y_{cm1} + \sin\alpha \cdot (Z_{cm1} + H_m) = \cos\alpha \cdot Y_{cm2} + \sin\alpha \cdot (Z_{cm2} + H_m) \tag{2-69}$$

and (2-69) implies

$$\frac{1}{Y_{cm2}} = \frac{e_v + \tan\alpha \cdot v_{m2}}{e_v + \tan\alpha \cdot v_{m1}} \cdot \frac{1}{Y_{cm1}} \tag{2-70}$$

By the definition of Δv_m in (2-60), it yields

$$\Delta v_{m2} = \frac{e_v + \tan\alpha \cdot v_{m2}}{e_v + \tan\alpha \cdot v_{m1}} \cdot \Delta v_{m1} \tag{2-71}$$

Since Δv_{m1} is known from the pixel correspondence of P_{m1} and P_{s1}, P_{s2} (u_{s2}, v_{s2}) can be derived from (2-61) and (2-62) by using (2-71) to get Δv_{m2}.
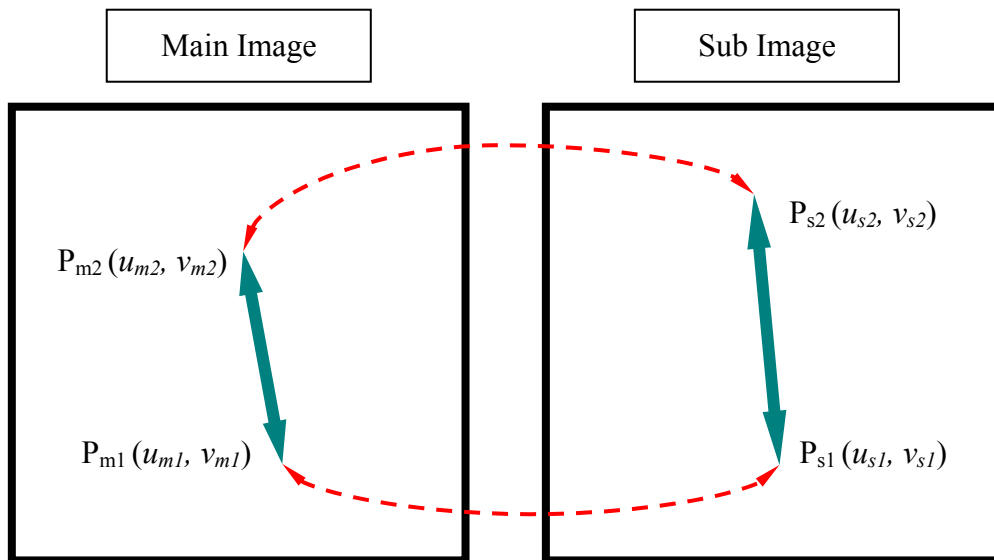
Fig. 2.5   The corresponding pixels between main and sub images.

CHAPTER 2 STEREO VISION SYSTEM

➢ Part C: *the condition of the same X_r*

In this case, the target is to determine $P_{s2} (u_{s2}, v_{s2})$ associated with $P_{m2} (u_{m2}, v_{m2})$ whose X_r is the same as that of $P_{m1} (u_{m1}, v_{m1})$.

In the beginning, $\mathbf{R}_r$ in (2-14) is re-presented by

$$\mathbf{R}_r = \begin{bmatrix} R_{00} & R_{01} & R_{02} & 0 \\ R_{10} & R_{11} & R_{12} & 0 \\ R_{20} & R_{21} & R_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2-72}$$

and it is obvious that $\mathbf{R}_r^{-1} = \mathbf{R}_r'$. And then (2-17) becomes

$$X_r = R_{00} \cdot X_c + R_{10} \cdot Y_c + R_{20} \cdot (Z_c + H) \tag{2-73}$$

Applying (2-73) to the equality of X_r yields

$$R_{00} \cdot X_{cm1} + R_{10} \cdot Y_{cm1} + R_{20} \cdot Z_{cm1} = R_{00} \cdot X_{cm2} + R_{10} \cdot Y_{cm2} + R_{20} \cdot Z_{cm2} \tag{2-74}$$

Let

$$
\begin{aligned}
E_i &= R_{00} \cdot X_{cmi} + R_{10} \cdot Y_{cmi} + R_{20} \cdot Z_{cmi} \\[1mm]
&= \frac{Y_{cmi}}{e_v} \left\{ R_{00} \cdot e_u \frac{X_{cmi}}{Y_{cmi}} \cdot \frac{e_v}{e_u} + R_{10} \cdot e_v + R_{20} \cdot e_v \frac{Z_{cmi}}{Y_{cmi}} \right\} \\[1mm]
&= \frac{\Delta H}{\Delta v_{mi}} \left\{ R_{00} \cdot u_{mi} \cdot \frac{e_v}{e_u} + R_{10} \cdot e_v + R_{20} \cdot v_{mi} \right\} \\[1mm]
&= \frac{\Delta H}{\Delta v_{mi}} \cdot \mathrm{PostDot}(u_{mi}, v_{mi}, R_0) \tag{2-75}
\end{aligned}
$$

Substituting (2-75) into (2-74) yields

$$\Delta v_{m2} = \frac{\mathrm{PostDot}(u_{m2}, v_{m2}, R_0)}{\mathrm{PostDot}(u_{m1}, v_{m1}, R_0)} \Delta v_{m1} \tag{2-76}$$

Since Δv_{m1} is given from the pixel correspondence of P_{m1} and P_{s1}, $P_{s2} (u_{s2}, v_{s2})$ can be determined from (2-61) and (2-62) by using (2-76) to get Δv_{m2}.

2.4 Calibration Principles

2.4.1 Calibration on both stereo cameras

The calibration on stereo cameras is a very important issue. The goal is to determine the relationship between both stereo cameras, and the result influences the accuracy of 3-D reconstruction. Once the calibration is finished, it is reasonable to suppose that the relative position between both cameras is invariant in use.

For the application in this thesis, we wish that no rotation occurs between both cameras to avoid the matrix calculation for the real-time consideration. However, it is difficult to reduce to zero in whole for the pitch, roll, and yaw angles between both cameras. But it is sure that we do our best to minimize the angles as small as possible.

The idea of the calibration is to transform both images into the same coordinate domain, and then to match the same objects in both images with each other. The ground on the road is an ideal choice for the pattern matching. As mentioned in Section 2.3.3, the transformation from the sub image coordinate system into the main image coordinate system is called "Vision Transform." The road image is transformed from the sub image domain into the main image domain, and the ground in both images is coincided with each other after the calibration.

Steps for the calibration on both stereo cameras are as follows:

(1) Set up the main camera to satisfy that its optical axis is paralleling the ground, i.e. $\alpha = 0$ and $\beta = 0$.

(2) Set up and regulate the sub camera by using Vision Transform to match the texture on the ground as possible, as displayed in Fig. 2.6.

(3) After setting up both cameras, estimate the angles by completely matching the ground texture, and then the relationship between both cameras is determined.
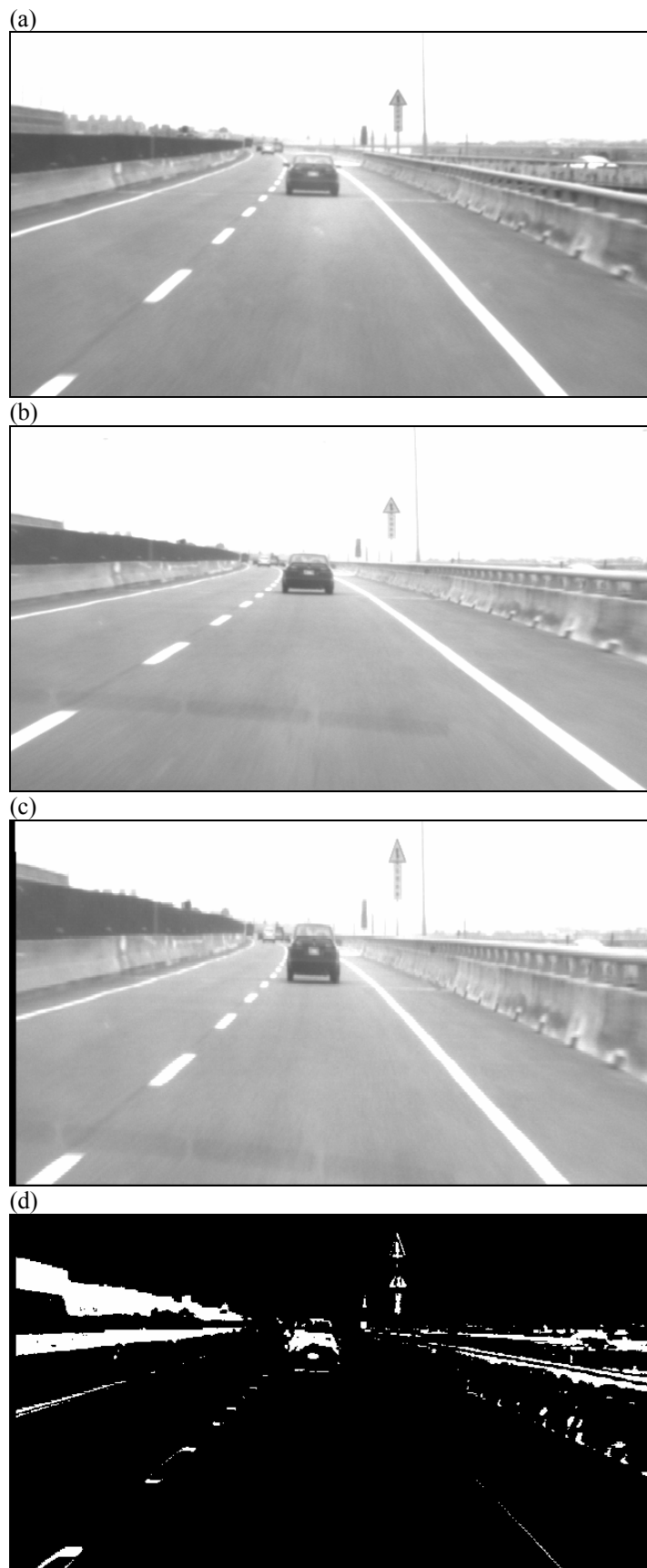
Fig. 2.6 Calibration on both stereo cameras. (a) The main (top) image. (b) The sub (bottom) image. (c) The sub image in (b) is mapped into the main image domain by Vision Transform. (d) The difference between (a) and (c).

2.4.2 Calibration on the main camera and the road

Due to the vibration in motion or the non-flat road surface, the pitch and roll angles, α and β, between the main camera and the road are always different. As discussed in Section 2.2.1, there exist certain α and β in every local zone so that $Z_r = 0$ is exactly the road plane. Here α and β for every local zone of $Z_r = 0$ are interesting.

Arrange (2-68) in another form:

$$-v_{mg} = \left(H_m\right) \cdot \left\{ \frac{e_v}{Y_{cmg}} \right\} + \left(\frac{\tan \beta}{\cos \alpha} \right) \cdot \left\{ u_{mg} \frac{e_v}{e_u} \right\} + \left(-\tan \alpha\right) \cdot \left\{ e_v \right\} \qquad (2\text{-}77)$$

If some data of u_{mg}, v_{mg}, and Y_{cmg} are given, then the unknowns, $\left(H_m\right)$, $\left(\tan \beta / \cos \alpha\right)$, and $\left(-\tan \alpha\right)$, of (2-77) can be solved by the least-squares method. However, as proposed in Section 2.2.3, the camera height has a smaller effect on the distance accuracy than the angles, and accordingly, H_m can be taken as a constant and be moved to the left side of (2-77). Again, (2-77) becomes

$$-\left\{ v_{mg} + e_v \frac{H_m}{Y_{cmg}} \right\} = \left(\frac{\tan \beta}{\cos \alpha} \right) \cdot \left\{ u_{mg} \frac{e_v}{e_u} \right\} + \left(-\tan \alpha\right) \cdot \left\{ e_v \right\} \qquad (2\text{-}78)$$

and two unknowns, $\left(\tan \beta / \cos \alpha\right)$ and $\left(-\tan \alpha\right)$ can be solved according to the same way. Finally, both α and β can be determined by the coefficients of (2-78).

Chapter 3 Generic Obstacle Detection

Since the preliminary knowledge of the computer vision has been proposed in Chapter 2, the algorithm of generic obstacle detection based on both stereo cameras will be introduced in this chapter. Two cameras are mounted top and bottom on the vehicle, and the top and bottom images captured by them are called the main and sub images respectively. Most manipulations are performed in the main image, and the sub image is used in the pattern matching. The details will be presented in the following.

3.1 Overview

Fig. 3.1 shows the flowchart of generic obstacle detection. The dashed blocks mean the successive detection mode and are not performed in the initial frame. In the beginning, the procedure of the preprocessing is executed in order to simplify the following detection. In the preprocessing, the road image captured by the main camera is segmented according to the gray levels and the so-called *Minimum Ground* in the main image is defined, which will be introduced in Section 3.4.

Proceeding to the next process, the flow enters the principal detection loop. In this detection loop, the boundaries are determined one by one, and then they are discriminated between the *ground* and *obstacle boundaries*. The so-called obstacle boundaries are the interconnecting boundaries between the ground and obstacles, and the others are called the ground boundaries. The discrimination method will be presented in Section 3.6.

The similar detection process is repeated until the *Row Leader* arrives at its ending. After that, all obstacle boundaries have been determined and then are updated for the detection in the next frame. The obstacle boundaries are divided into the *motion* and *roadside boundaries* according to their slopes. The boundaries with the sharp slopes are classified into the roadside

ones, and the others are the motion ones. And then the roadside boundaries are fitted to only the left and right roadside boundaries which can roughly bound the roadsides on the left and right sides of the lane.

If the obstacles have been detected in the last frame, they will be tracked in the current frame before the regular detection loop. However, only the motion boundaries can be tracked. The obstacle tracking can stabilize the detection result and reduce the detection time. The details will be proposed in Section 3.7.
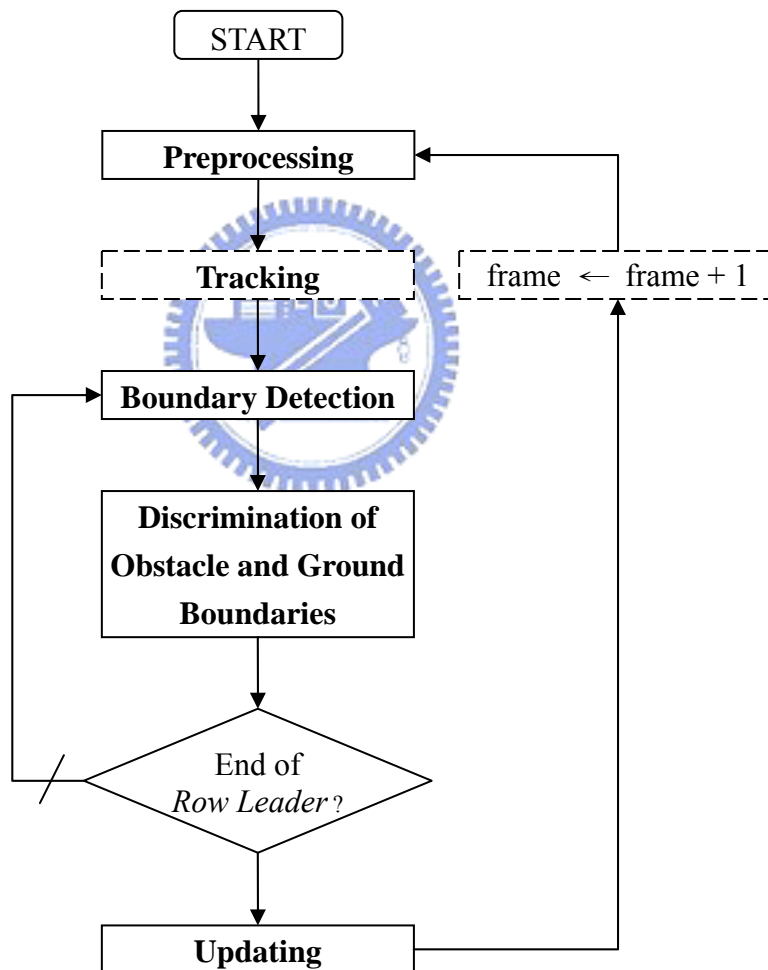
Fig. 3.1   The flowchart of generic obstacle detection.

3.2 Image Segmentation by Thresholding

In this section the goal is to segment the road image into several groups according to the gray levels. Given a grayscale image, the mean and variance within the region of interest (ROI) in the grayscale histogram h_l, $l \in [0, 255]$, are respectively computed as:

$$\mu = \frac{\sum\limits_{l \in [L,H]} h_l \cdot l}{M_0} = \frac{M_1}{M_0} \tag{3-1}$$

$$\sigma^2 = \frac{\sum\limits_{l \in [L,H]} h_l \cdot (l - \mu)^2}{M_0} = \frac{M_2}{M_0} - \left(\frac{M_1}{M_0} \right)^2 \tag{3-2}$$

where l is the gray level, L and H are, respectively, the low and high bounds of the ROI, and M_0, M_1, and M_2 are, respectively, the zero, first, and second moments of the histogram to the origin, represented as follows:
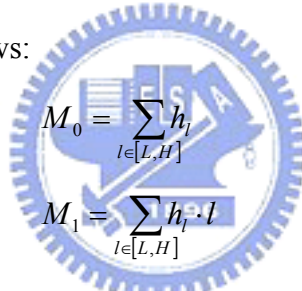
$$M_0 = \sum_{l \in [L,H]} h_l \tag{3-3}$$

$$M_1 = \sum_{l \in [L,H]} h_l \cdot l \tag{3-4}$$

$$M_2 = \sum_{l \in [L,H]} h_l \cdot l^2 \tag{3-5}$$

The number of clusters in the ROI is not unique if $\sigma^2 > \sigma_{th}^2$ and there exists a threshold l_{th} partitioning the region into two clusters so as to maximize the between-class variance, say $\sigma_{B1,2}^2$, where

$$\sigma_{B1,2}^2 = \omega_1 \omega_2 (\mu_1 - \mu_2)^2 \tag{3-6}$$

$$\omega_1 = M_0^1 / M_0 \tag{3-7}$$

$$\omega_2 = 1 - \omega_1 \tag{3-8}$$

$$\mu_1 = \frac{M_1^1}{M_0^1} \tag{3-9}$$

$$\mu_2 = \frac{(M_1 - M_1^1)}{(M_0 - M_0^1)} \tag{3-10}$$

$$M_0^1 = \sum_{l \in [L, l_{th})} h_l \quad \text{(the zero moment of histogram within class 1 to the origin)} \quad (3\text{-}11)$$

and $\quad M_1^1 = \sum_{l \in [L, l_{th})} h_l \cdot l$ (the first moment of histogram within class 1 to the origin) $\quad (3\text{-}12)$

The above clustering process can be applied to each cluster iteratively until $\sigma^2 \leq \sigma_{th}^2$ or

$M_0 / \sum_{l \in [0,255]} h_l \leq P_{th}$. The reason to bound the population within the ROI in the grayscale

histogram is that the small class may be useless and be referred to the noise as detecting the

obstacle in the road image. Therefore it is a benefit for the following processing to avoid the

small cluster.

On condition of a blaze of daylight or the illimitable highway the gray levels on the

farther surface of the road are similar to those in the heavens, and hence both may be the same

class after the clustering process. As the histogram in Fig. 3.2 (b), the farther surface and the

sky are classified the same by the gray level, say l_{max}. However, there exists a threshold,

namely l_{bright}, meeting that $\sum_{l=l_{bright}}^{255} h_l \geq P_{bright}$, used to replace l_{max} for the separation

between the road and the sky if $l_{bright} > l_{max}$. An example is demonstrated in Fig. 3.3.

(a)

(b)

$I_{max}I_{bright}$
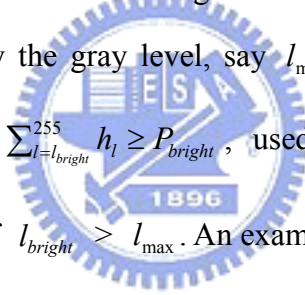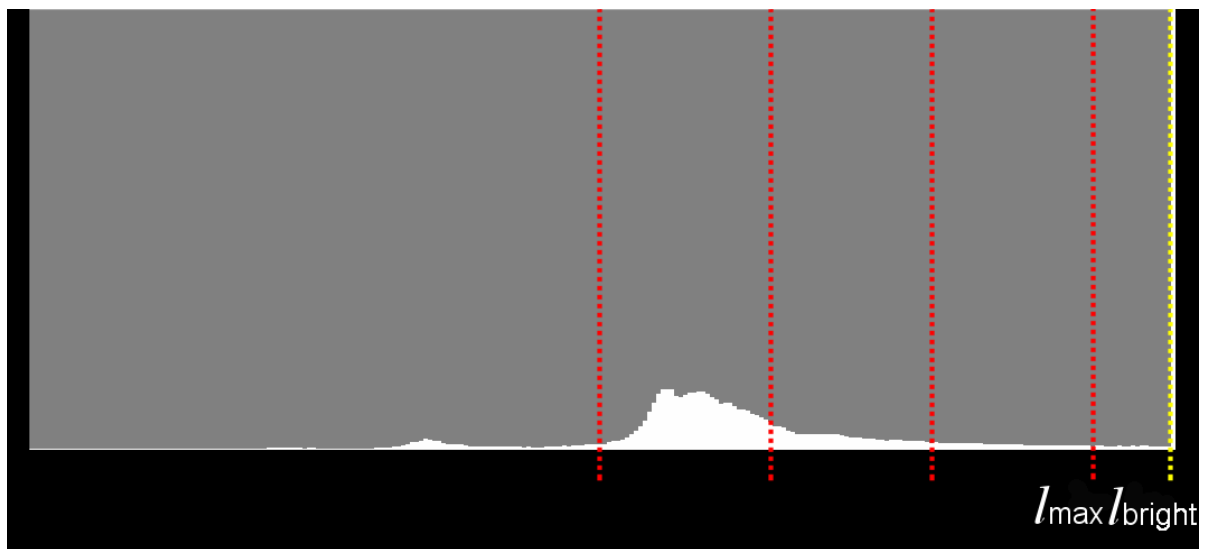
Fig. 3.2　The grayscale histogram of a road image with a blaze on the farther road surface. (a) A road image with a blaze on the farther road surface. (b) The grayscale histogram of (a).

(a)

(b)
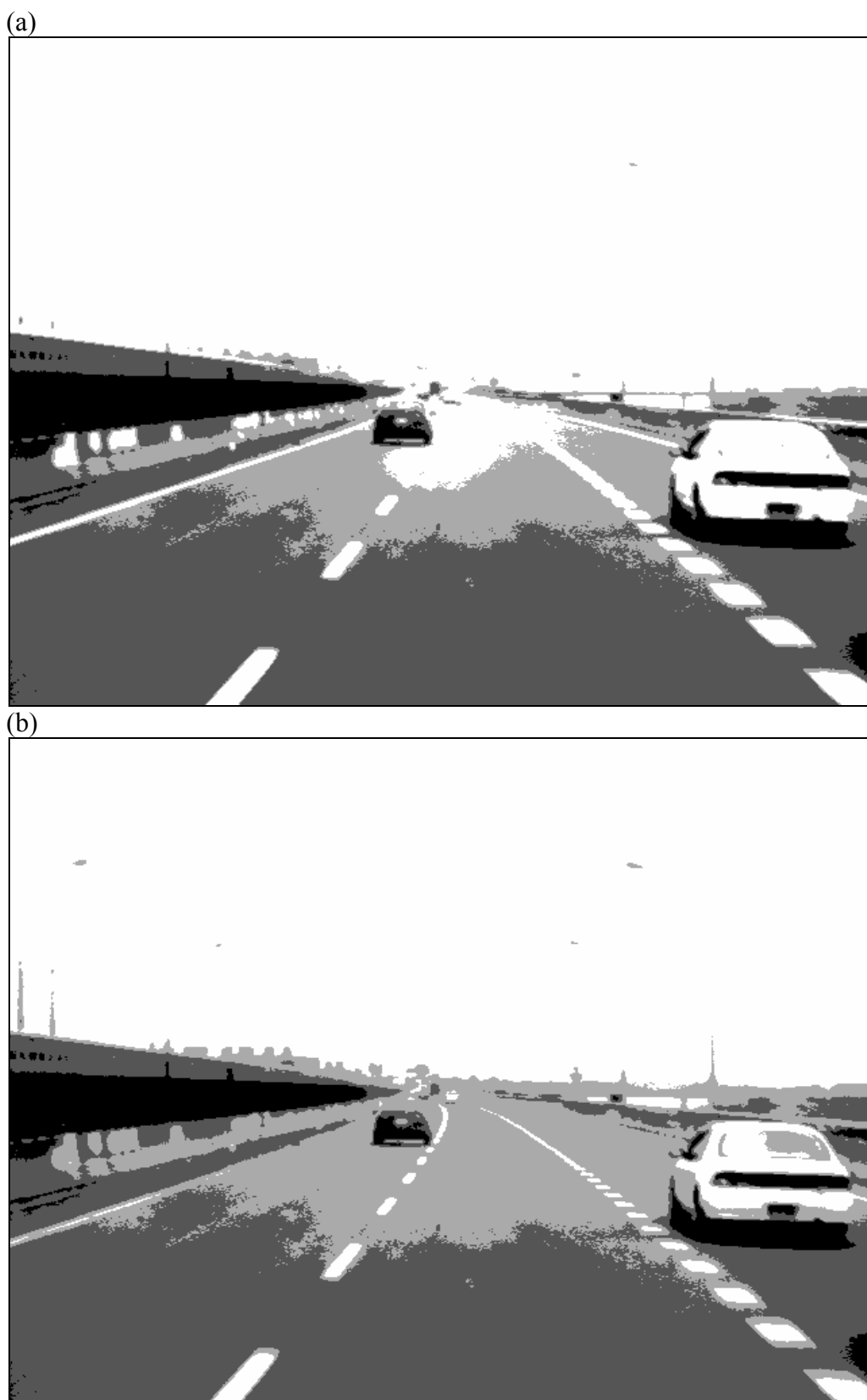
Fig. 3.3 An example of the road image segmentation by different thresholds. (a) The segmented image of Fig. 3.2 (a) by the threshold l_{max}. (b) The segmented image of Fig. 3.2 (a) by the threshold l_{bright}.

3.3 Boundary Detection

The goal of boundary detection is to determine the significant boundaries which could be the textures on the ground, the interconnection between the ground and obstacles, or the edges on the obstacles. Since the road image has been segmented as described in Section 3.2, the boundary is the connection of edge pixels between two different clusters. An example is displayed in Fig. 3.4. The detection process will be proposed later.

3.3.1 Overview

Since the boundary is composed of edge pixels between two different clusters, the idea of boundary detection is to determine an edge pixel at first, and then to expand it into a boundary, as illustrated in Fig. 3.4.

In practice, a boundary is represented as the set of rows in the road image, and only one row per boundary column has to be recorded. Therefore, the size of each boundary can be simply regarded as the number of boundary columns. The target here is to determine the corresponding rows for each boundary column.

In order to detect the edge pixel for the expansion, pixels are scanned row by row for every column. As shown in Fig. 3.4 (a), the row bound, namely the *Row Leader*, is to limit what the current row can not exceed while scanning the edge, which guarantees that the edge pixel with the lowest row among all columns is found out first. And the lowest boundary is then produced by expanding the lowest edge pixel. Consequently, all boundaries will be detected in order from bottom to top in the segmented image.

The flowchart of boundary detection is shown in Fig. 3.5. If an edge pixel is found out at a certain column, the boundary is confirmed after the expansion process. A boundary is valid if its size is large enough, and it will be partitioned into several ones according to its disjunctive points, which will be introduced later.

(a)

(b)

(c)

(d)
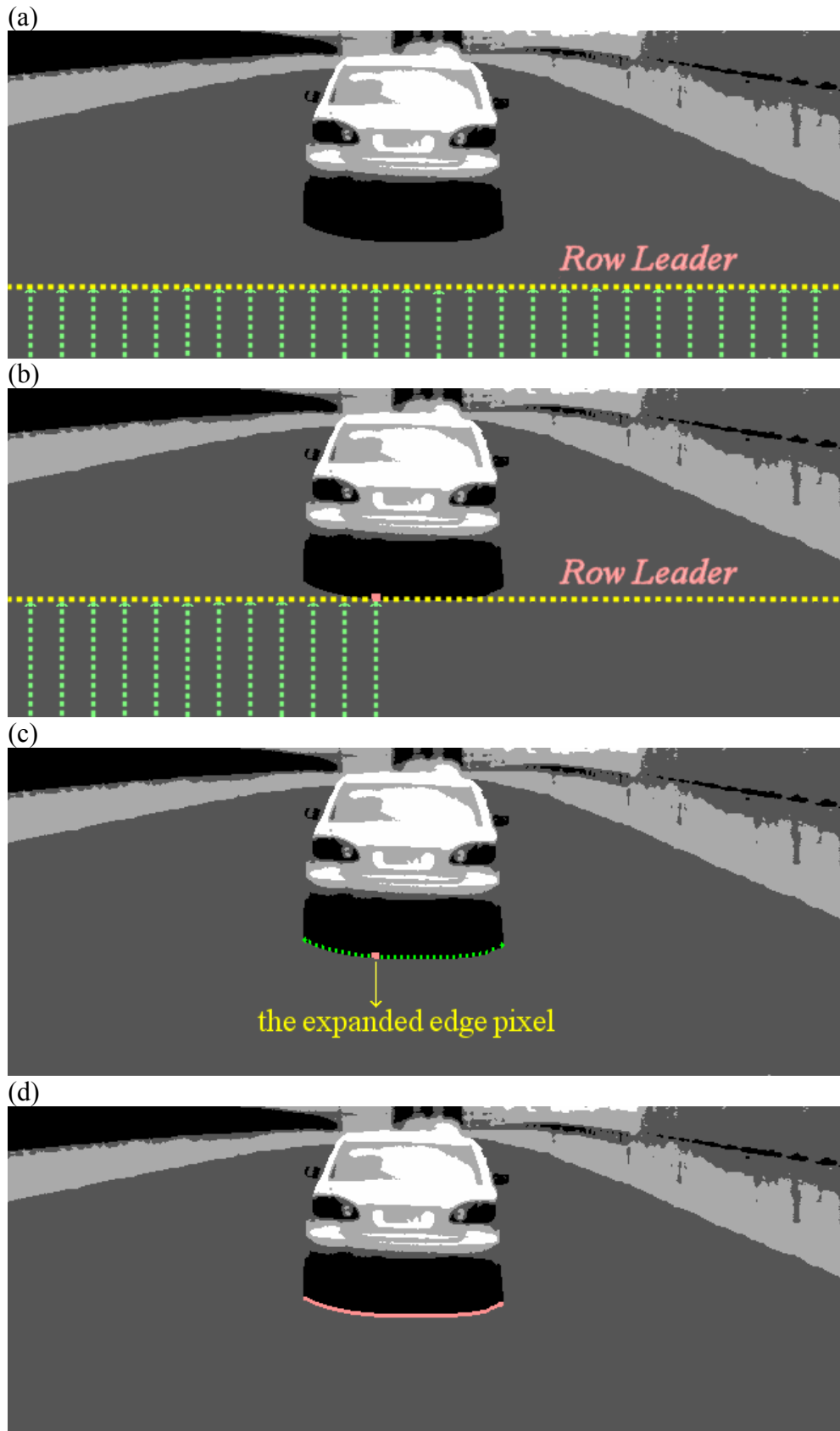
Fig. 3.4 An example of boundary detection. (a) The current row can not exceed the *Row Leader* while scanning the edge. (b) The edge pixel of the lowest row is detected first. (c) The boundary is determined by expanding the edge pixel. (d) The final extended boundary.

If no boundary is determined for a specific *Row Leader*, the *Row Leader* moves up, and then a new iteration is run. This process is repeated until any valid boundary is found out or the *Row Leader* arrives at its ending.
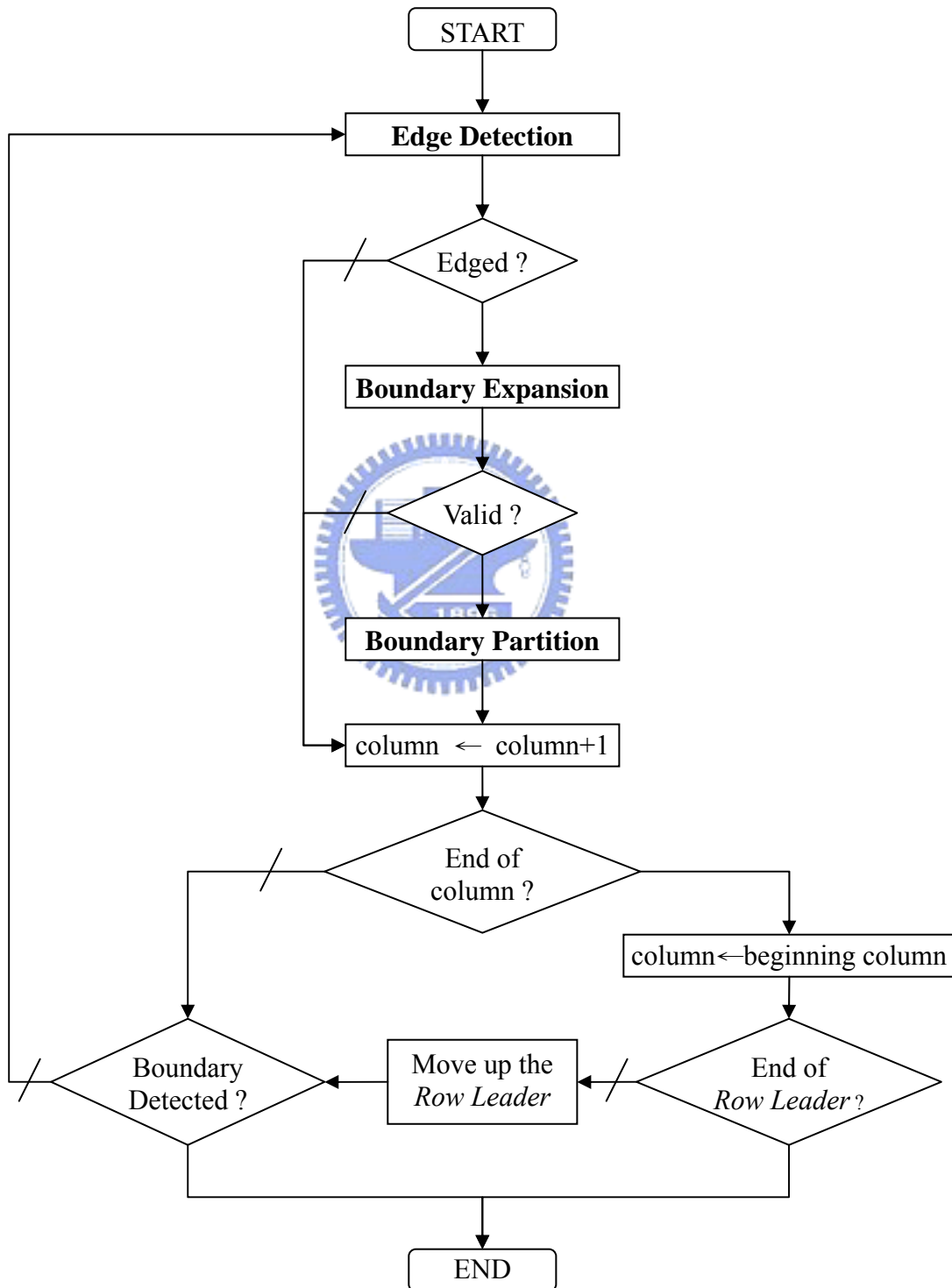
Fig. 3.5   The flowchart of boundary detection.

3.3.2 Edge Detection

As mentioned above, an edge is mode up of two different clusters in the segmented image. And an edge pixel is defined as the upper one of two pixels with different clusters. The edge detection is to determine an edge row for a given column on condition of $row \leq Row\ Leader$. The edge row corresponding to the given column is searched from the row of the last iteration to the *Row Leader*, and this process is terminated when an edge pixel is found out. Fig. 3.6 shows the flowchart of edge detection.

Fig. 3.6   The flowchart of edge detection.
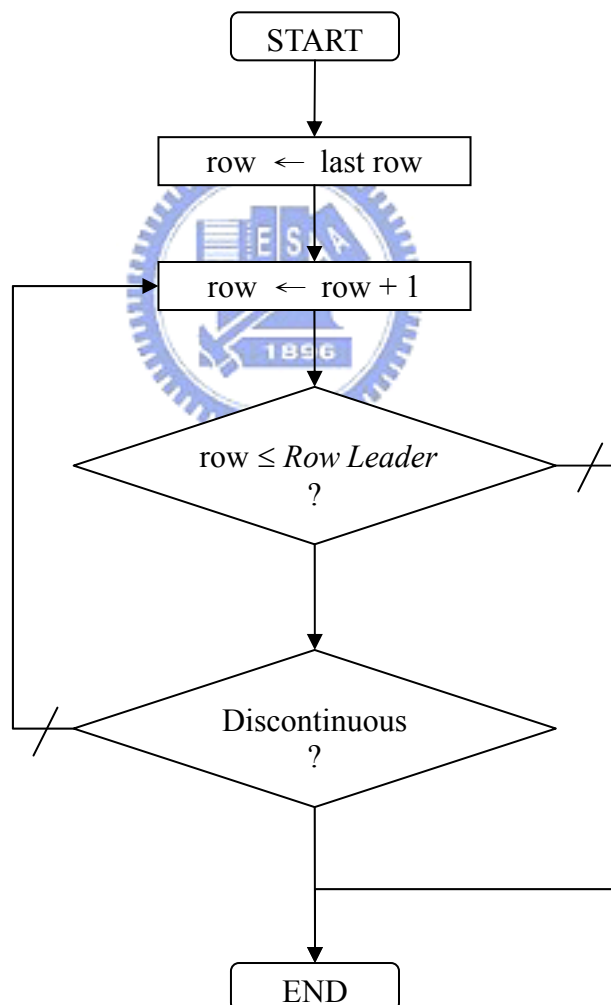
CHAPTER 3 GENERIC OBSTACLE DETECTION

3.3.3 Boundary Expansion

Given an edge pixel, the boundary can be leftward and rightward extended according to the same cluster. An 8-directional connecting process is used to extend the boundary. Fig. 3.7 indicates the direction numbers. Since the expanded edge pixel is the upper one whose cluster differs from that of the lower one, its direction number can be initialized to 0. And then the boundary is extended leftward and rightward by searching for the ways clockwise and counterclockwise, respectively.
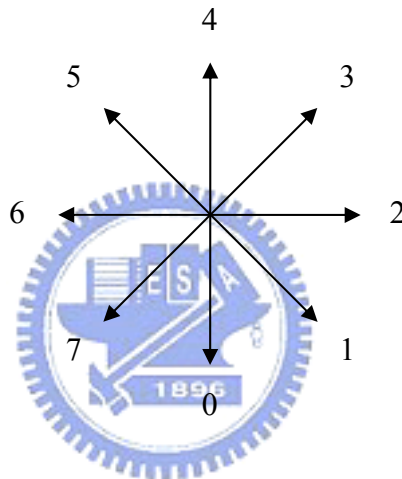
Fig. 3.7   The direction numbers for the 8-directional connecting process.

The expansion process stops on some conditions stated as follows:

(1)   The current searching pixel comes back to the beginning entry.

(2)   A U-turn is too deeper because two similar boundaries are closer very much. Fig. 3.8 (a) displays such an example.

(3)   There are too many steps of the vertical motion at a time. It could be the case of the vertical boundary on the obstacles, which is not desired, as shown in Fig. 3.8 (b).

Finally, it is necessary to notice that only the lowest rows for every columns of the boundary have to be recorded.

(a) A deeper U-turn (b) Too many vertical steps
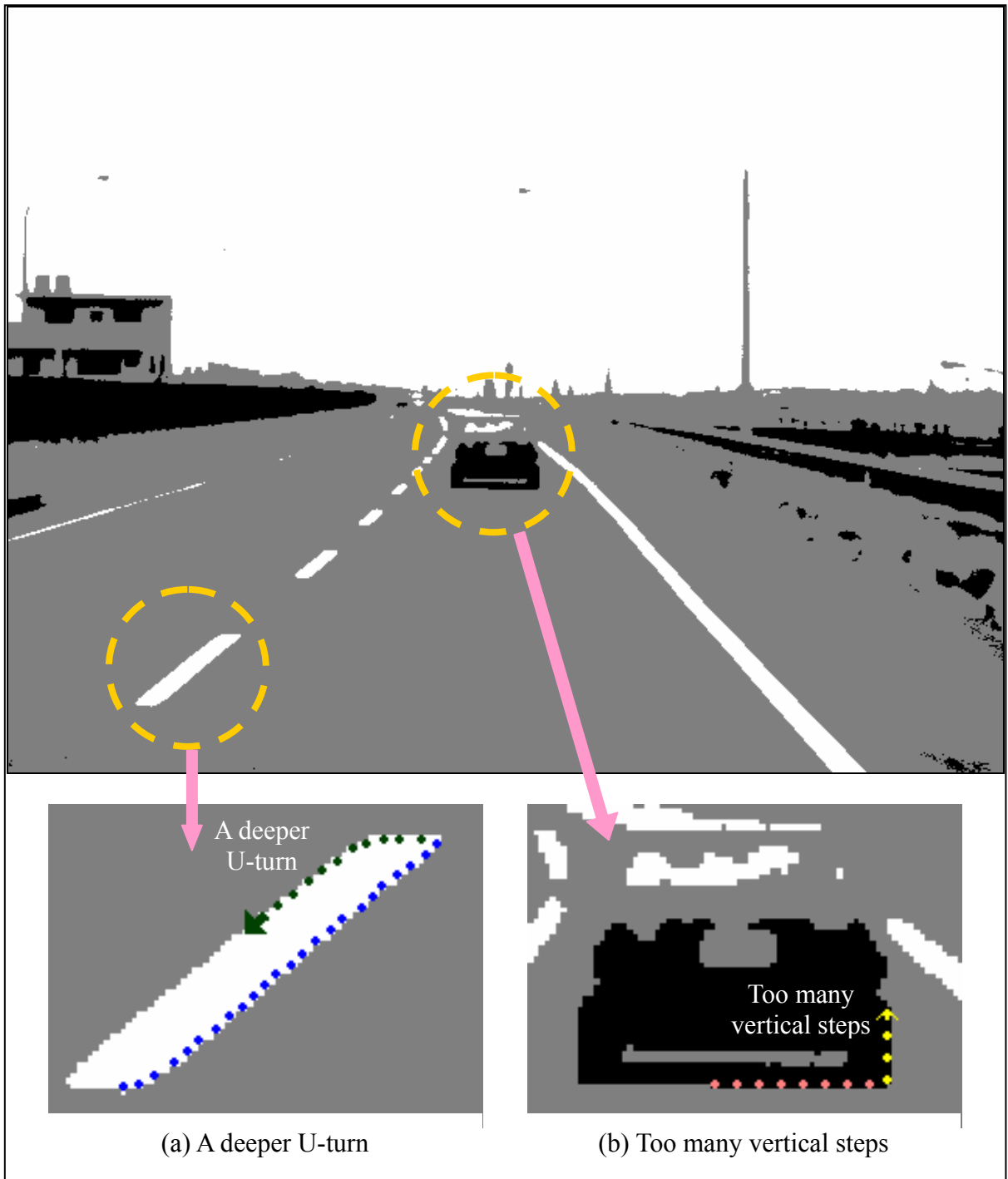
Fig. 3.8   Some restrictions on the boundary expansion. (a) A deeper U-turn. (b) Too many vertical steps.

3.3.4 Boundary Partition

After the expansion process, it can obtain a set of connective pixels, which could be regarded as what is composed of several quasi-straight lines. In this procedure, the extended boundary is partitioned into several quasi-straight ones according to their interconnecting points.

A quasi-straight boundary is smooth, and a disjunctive pixel is defined as the interconnecting point of two adjoining boundaries, which form an angle ϕ satisfying $\phi < \phi_{th}$, as illustrated in Fig. 3.9. After partitioning a divisible boundary into several ones, some of them are erased if their sizes are invalid, and the others are recorded as new boundaries, as shown in the flowchart of boundary partition in Fig. 3.10.
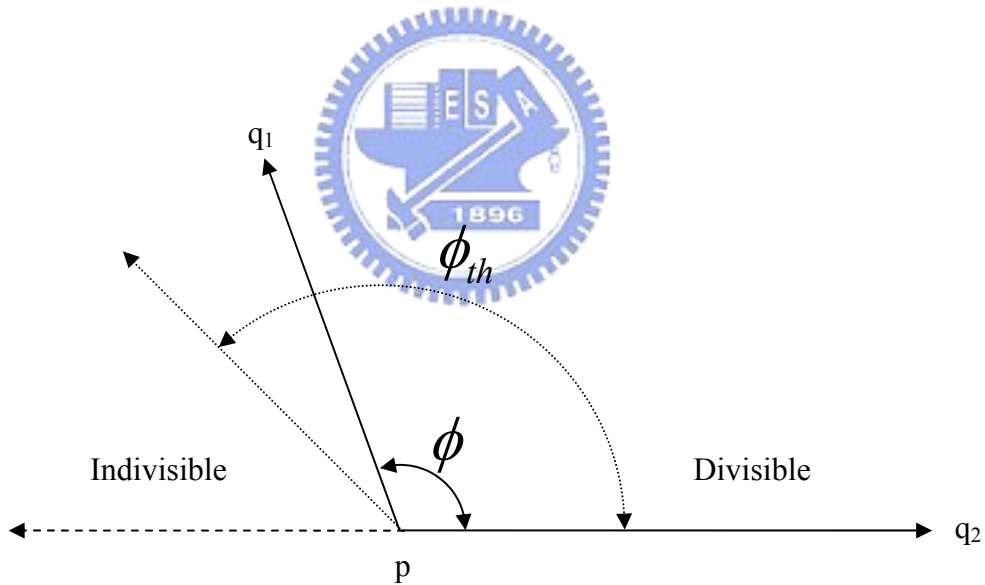
Fig. 3.9　The disjunctive point, p, satisfies $\phi < \phi_{th}$.

A well-known formula is used to calculate the angle ϕ

$$\phi = \cos^{-1}\left\{ \frac{\overrightarrow{pq_1} \bullet \overrightarrow{pq_2}}{\left\| \overrightarrow{pq_1} \right\| \left\| \overrightarrow{pq_2} \right\|} \right\}$$

(3-13)

However, instead of (3-13), another form is utilized to avoid the computation of square roots :

$$sign(\cos\phi)\cdot\left(\cos^2\phi\right)= sign\left(\overrightarrow{pq_1}\bullet\overrightarrow{pq_2}\right)\cdot\frac{\left(\overrightarrow{pq_1}\bullet\overrightarrow{pq_2}\right)^2}{\left\|\overrightarrow{pq_1}\right\|^2\left\|\overrightarrow{pq_2}\right\|^2} \tag{3-14}$$

where $sign(x)=1$ if $x\geq0$, -1 if $x<0$. The point p is said disjunctive on condition of

$$sign(\cos\phi)\cdot\left(\cos^2\phi\right)>-\cos^2\phi_{th}.$$
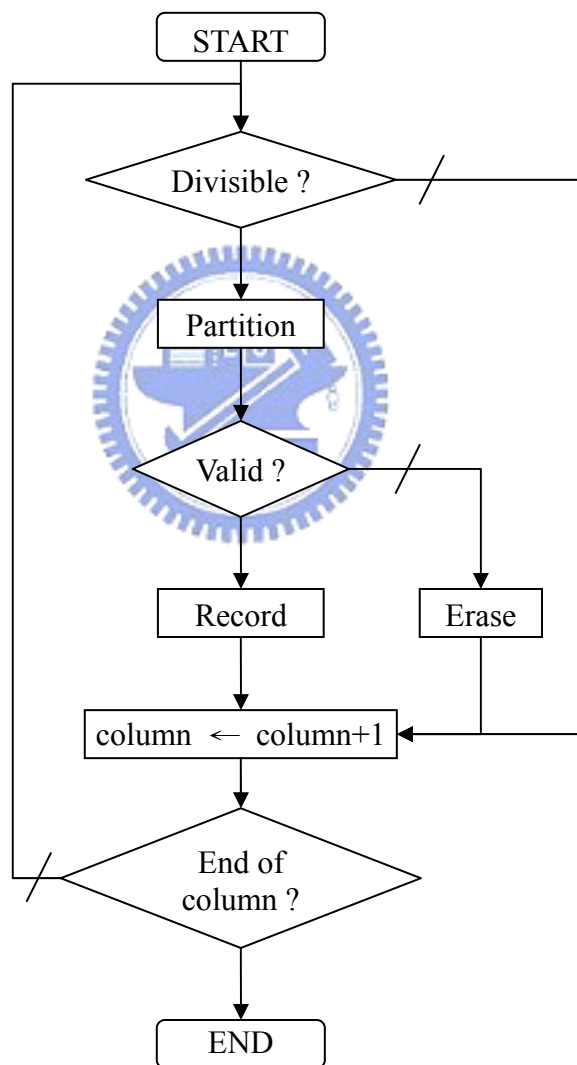
Fig. 3.10 The flowchart of boundary partition.

3.4 Preprocessing

3.4.1 Overview

Since the objects in the road image can be divided into two parts, i.e. obstacles and the ground, the topic of obstacle detection is reduced to the issue of removing all the ground in the road image. However, it is difficult to do so, but the problem of obstacle detection can be simplified by removing the likely-known ground, namely the *Minimum Ground*, which will be presented in Section 3.4.2.

The flowchart of the preprocessing is shown in Fig. 3.11. In the beginning, the road image is segmented by thresholding the histogram, as described in Section 3.2. In this step, a smaller threshold, σ_{th}^2, of the between-class variance is chosen to generate much more clusters so as to ensure no loss of the interconnecting boundaries between the ground and obstacles.
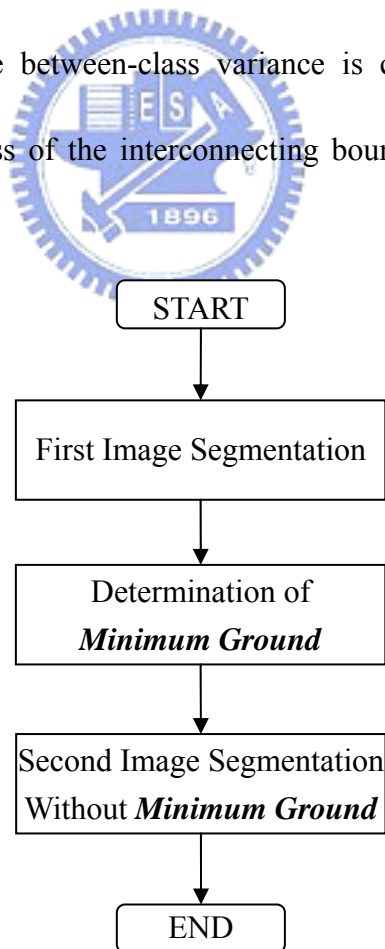
Fig. 3.11 The flowchart of the preprocessing.
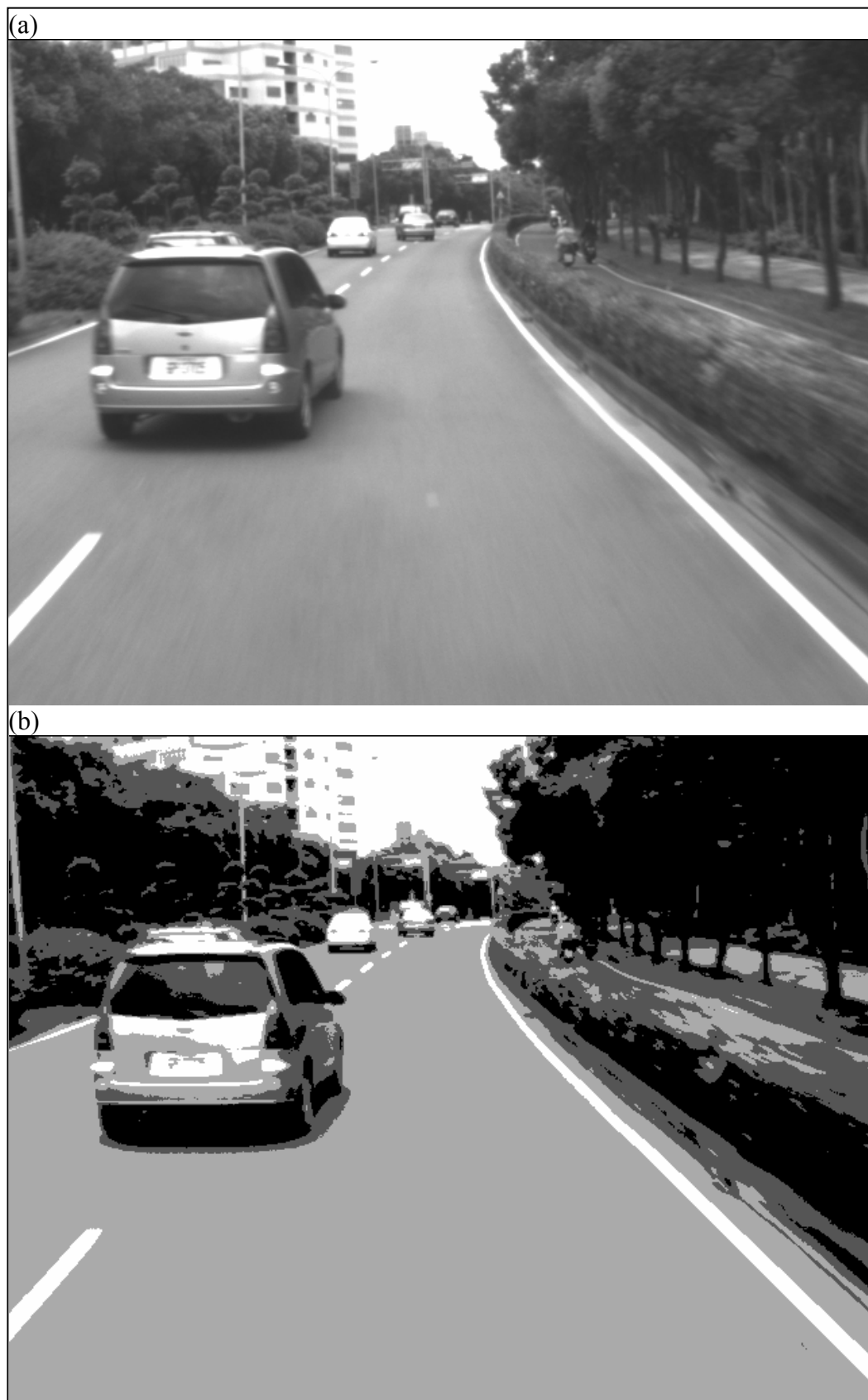
Fig. 3.12 An example of the preprocessing. (a) The original road image. (b) After the first image segmentation.

Fig. 3.12 An example of the preprocessing. (c) The road image with the *Minimum Ground*. (d) After the second image segmentation without the *Minimum Ground*.

To proceed, the *Minimum Ground* is determined, and thus the location of the *minimum rows* for every column is known. Since the *Minimum Ground* is defined, its effect on the grayscale histogram can be removed, and thus the obstacles occupy most of the remainder. Again, a new segmentation by thresholding the reshaped histogram is performed, and a greater σ_{th}^2 is chosen to create fewer clusters without classifying the obstacles too much.

An example is demonstrated in Fig. 3.12. It is obvious that much less noise on the obstacles after the second clustering process, and it is a benefit for the following obstacle detection.

3.4.2 Minimum Ground

In this section the approach to determine the *Minimum Ground* is proposed. At first, three assumptions are given as follows:

(1) The ground and obstacles on the road are interconnected.

(2) The interconnection between obstacles and the ground is visible in the road image.

(3) The artifacts in the road image, such as other vehicles and textures on the ground, contain some quasi-horizontal lines [5].

By upwardly searching the edge pixels for every column in the road image, the first-detected pixels with a grater gradient are called the *minimum rows*, which could belong to the textures on the road surface or the interconnecting boundaries between the ground and obstacles. In consequence, there exists only one *minimum row* per column, and all *minimum rows* in every column enclose a region of the ground, which is exactly the *Minimum Ground*, see Fig. 3.12 (c).

In order to determine the *Minimum Ground*, the technique of boundary detection is used to determine the ground boundaries. The overall image is scanned, and boundaries are generated one by one. A new detected boundary is preserved if its gradient is greater enough, or it is erased if not. While detecting the boundaries, the columns inside the existed boundaries are never scanned. Finally, only one row per column is recorded, and these rows enclose the region of the *Minimum Ground*.

As shown in Fig. 3.13 (a), the boundary gradient is computed by

$$Grad(\text{boundary}) = \frac{\left| \sum\limits_{\text{column} \in \text{boundary}} \left\{ \sum\limits_{i=1}^{N} \left[G(column, row+i) - G(column, row-i) \right] \right\} \right|}{N \cdot \text{boundary size}} \qquad (3\text{-}15)$$

where $G(c,r)$ is the gray level of the pixel at (c, r) and N is the neighbor size of the boundary. The neighbor size N is adaptive and is determined from three steps:

(1) Divide the boundary into four parts of the same spacing by three points; see Fig. 3.13 (b).

(2) Scan upwardly the pixels row by row in the segmented image from these three points, respectively, until the different clusters are met.

(3) Choose the maximum size of three scanning lines as the neighbor size N.

(a) The illustration of (3-15).

(b) Divide the boundary into 4 parts by 3 points. The size of the longest scanning line at the starting point A is chosen as the neighbor size N.
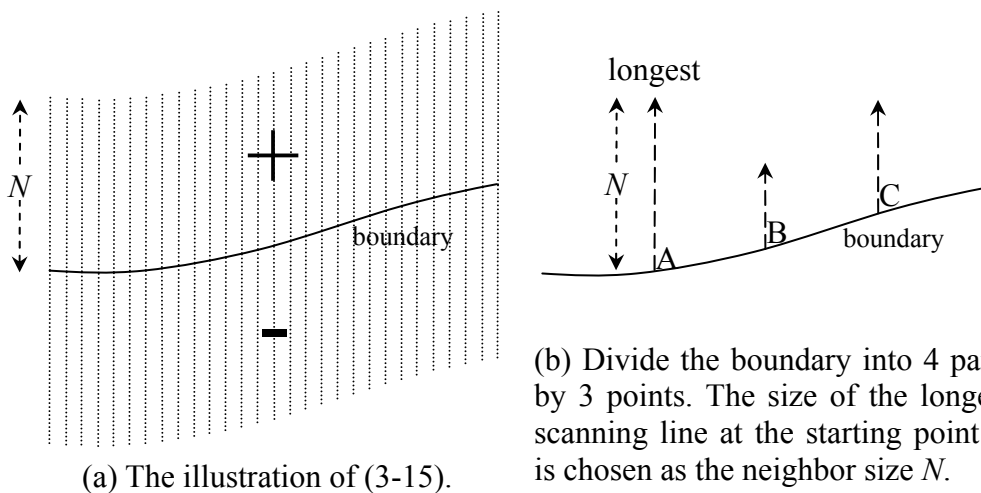
Fig. 3.13 The boundary gradient. (a) The illustration of (3-15). (b) Divide the boundary into 4 parts by 3 points. The size of the longest scanning line at the starting point A is chosen as the neighbor size N.

3.5 Estimation of Pith and Roll Angles between the Main Camera

and the Ground

3.5.1 Overview

Due to the vibration in motion or the non-flat road surface, the pitch and roll angles, i.e. α and β, between the main camera and the ground are time-variant. As mentioned in Section 2.4.2, (2-78) can be utilized to estimate α and β if given some data of u_{mg}, v_{mg}, and Y_{cmg}. And Y_{cmg} can be determined from (2-50) if the pixel correspondence of stereo image coordinates, u_{mg}, v_{mg}, and u_{sg}, is known. Therefore, the pixel correspondence of stereo images must be determined first for the purpose of estimating α and β.

Given a ground boundary, $\left\{ \left(u_{mg}^{i}, v_{mg}^{i} \right) \middle| i \in \text{boundary pixels} \right\}$, of the main image, its corresponding ground boundary, $\left\{ \left(u_{sg}^{i}, v_{sg}^{i} \right) \middle| i \in \text{boundary pixels} \right\}$, of the sub image can be determined by searching the sub image and matching $\left(u_{sg}^{i}, v_{sg}^{i} \right)$ with $\left(u_{mg}^{i}, v_{mg}^{i} \right)$. After that, the pixel pairs of stereo images are confirmed and then α and β can be estimated from (2-78).

3.5.2 Similarity measure based on normalized correlation coefficient

A pixel pair of stereo images is said the pixel correspondence if their corresponding blocks of neighbor pixels are similar. Let the neighbor blocks in the main and sub images be, respectively, the $(2M+1)\times(2N+1)$ image arrays, $m(u_m+i,v_m+j)$ and $s(u_s+i,v_s+j)$, $-M \le i \le M, -N \le j \le N$. The goal in this section is to develop a measure of the similarity between $m(u_m+i,v_m+j)$ and $s(u_s+i,v_s+j)$.

The so-called *normalized correlation coefficient*, defined as

$$NCC(u_m,v_m,u_s,v_s)=\frac{\displaystyle\sum_{i=-M}^{M}\sum_{j=-N}^{N}[m(u_m+i,v_m+j)-\overline{m}][s(u_s+i,v_s+j)-\overline{s}]}{\sqrt{\displaystyle\sum_{i=-M}^{M}\sum_{j=-N}^{N}[m(u_m+i,v_m+j)-\overline{m}]^2\cdot\sum_{i=-M}^{M}\sum_{j=-N}^{N}[s(u_s+i,v_s+j)-\overline{s}]^2}} \tag{3-16}$$

is an appropriate similarity measure which is not too sensitive to illumination changes [21], where $\overline{m}$ and $\overline{s}$ are the means of $m(u_m+i,v_m+j)$ and $s(u_s+i,v_s+j)$, respectively. Two neighbor blocks $m(u_m+i,v_m+j)$ and $s(u_s+i,v_s+j)$ are highly correlated if $NCC(u_m,v_m,u_s,v_s)$ approximates to 1, and in this case (u_m,v_m) and (u_s,v_s) are the corresponding pair.

(3-16) can be exploited to generate more accurate and reliable results but would lead to a high computational load. Some steps are taken into account for the simplification. Consider the following.

$$\sum_{i=-M}^{M}\sum_{j=-N}^{N}(m_{ij}-\overline{m})(s_{ij}-\overline{s})$$

$$=\sum_{i=-M}^{M}\sum_{j=-N}^{N}m_{ij}s_{ij}-\overline{s}\cdot\sum_{i=-M}^{M}\sum_{j=-N}^{N}m_{ij}-\overline{m}\cdot\sum_{i=-M}^{M}\sum_{j=-N}^{N}s_{ij}+\sum_{i=-M}^{M}\sum_{j=-N}^{N}\overline{m}\cdot\overline{s}$$

$$=\sum_{i=-M}^{M}\sum_{j=-N}^{N}m_{ij}s_{ij}-(2M+1)(2N+1)\cdot\overline{m}\cdot\overline{s}-(2M+1)(2N+1)\cdot\overline{m}\cdot\overline{s}+(2M+1)(2N+1)\cdot\overline{m}\cdot\overline{s}$$

$$=\sum_{i=-M}^{M}\sum_{j=-N}^{N}m_{ij}s_{ij}-(2M+1)(2N+1)\cdot\overline{m}\cdot\overline{s} \tag{3-17}$$

$$\sum_{i=-M}^{M} \sum_{j=-N}^{N} \left(m_{ij} - \overline{m}\right)^2$$

$$= \sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}^2 - 2 \sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}\overline{m} + \sum_{i=-M}^{M} \sum_{j=-N}^{N} \overline{m}^2$$

$$= \sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}^2 - 2(2M+1)(2N+1)\overline{m}^2 + (2M+1)(2N+1)\overline{m}^2$$

$$= \sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}^2 - (2M+1)(2N+1)\overline{m}^2 \qquad (3\text{-}18)$$

Hence (3-16) is reduced to

$$NCC(u_m, v_m, u_s, v_s) = \frac{\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}s_{ij} - (2M+1)(2N+1) \cdot \overline{m} \cdot \overline{s}}{\sqrt{\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}^2 - (2M+1)(2N+1)\overline{m}^2} \sqrt{\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} s_{ij}^2 - (2M+1)(2N+1)\overline{s}^2}} \qquad (3\text{-}19)$$

In implementation, squaring (3-19) and taking its sign into account yield the *signed square normalized correlation coefficient*, i.e.

$$SSNCC(u_m, v_m, u_s, v_s) =$$

$$\frac{sign\left(\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}s_{ij} - (2M+1)(2N+1) \cdot \overline{m} \cdot \overline{s}\right) \cdot \left(\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}s_{ij} - (2M+1)(2N+1) \cdot \overline{m} \cdot \overline{s}\right)^2}{\left(\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} m_{ij}^2 - (2M+1)(2N+1)\overline{m}^2\right)\left(\displaystyle\sum_{i=-M}^{M} \sum_{j=-N}^{N} s_{ij}^2 - (2M+1)(2N+1)\overline{s}^2\right)} \qquad (3\text{-}20)$$

where $sign(x)=1$ if $x \geq 0$, -1 if $x < 0$. Paying attention to (3-20), it is clear that

(1) The computation of square roots is eliminated.

(2) Both m_{ij}^2 and s_{ij}^2 can be determined from a look-up table of size 256 since the gray level ranges from 0 to 255.

(3) All terms of (3-20) can be confirmed simultaneously in only one loop of traveling throughout the neighbor blocks.

(4) (3-20) is still scaled in the -1 to 1 range.

As a consequence, (3-20) is used for the similarity measure in practice.

3.5.3 Logarithmic search for pattern matching

In order to determine the pixel correspondence, the neighbor block $m(u_m + i, v_m + j)$ in the main image is superimposed on the sub image and it is translated to all possible positions (u_s, v_s) within it in order to locate the maximum correlation. Due to the fact that two cameras are mounted top and bottom, respectively, and that the angles between them are made small as possible during the calibration process, the pattern search can be restricted to the one-dimensional vertical direction.

The initial searching center ■   ① $(r, k) = (4, 2)$

② $(r, k) = (2, 1)$

The matched pixel   ③ $(r, k) = (1, 0)$
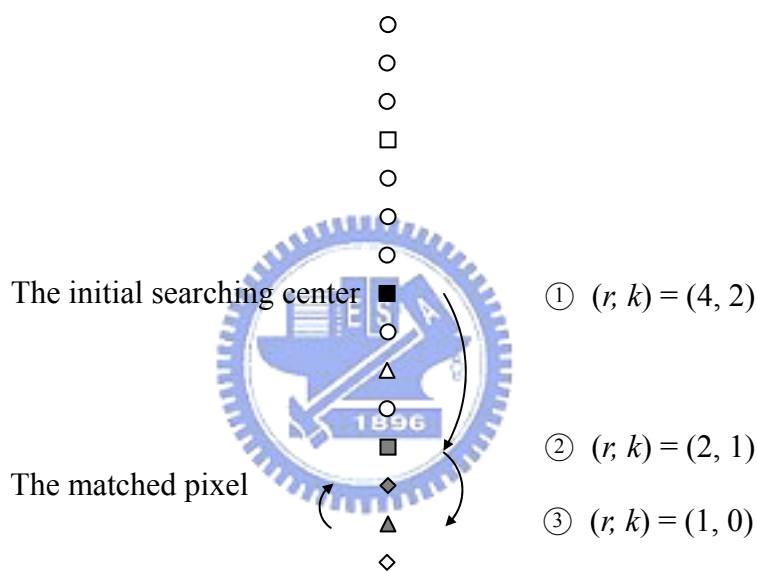
Fig. 3.14 The 1-D logarithmic search in the case of $r = 4$ ($k = 2$).

However, the 1-D full search still traverses too many possibilities to save the computational time. It is better to use the 1-D logarithmic search to reduce the possibilities [22]. The searching location is initialized to the center of the vertical path of size $4r - 1$, where $r = 2^k$, k is a non-negative integer. Fig. 3.14 demonstrates the case of $r = 4$ ($k = 2$). The computation of (3-20) is first performed at the center as well as the two points located at a radius of r pixels. After that, the center is moved to the point of the maximum correlation and $r = r / 2$ ($k = k - 1$). The process is repeated and finally the computation is performed at

the two points at a radius of $r = 1$. The matched pixel (u_s, v_s) is located where its correlation is the maximum among these three points. It is obvious that the number of computations is $4 \cdot 2^k - 1$ operations for the 1-D full search and is reduced to $2(k+1)+1$ for the 1-D logarithmic search.

As we know, the number of operations for the 1-D logarithmic search is associated with the initial radius r. If the probable location of the matched pixel is able to be predicted, a smaller r will be initialized so as to decrease the searching cost.

Since the corresponding pixels on the ground on the condition of $Z_r = 0$ are desired, in the ideal case of $\alpha = 0$ and $\beta = 0$, (2-68) becomes

$$\Delta v_{mg} = -\frac{\Delta H}{H_m} \cdot v_{mg} \tag{3-21}$$

Hence, after substituting (3-21) into both (2-61) and (2-62), the corresponding pixels (u_{sg}, v_{sg}) in the sub image can be obtained. In our case that $H_m = 172\,cm$, $H_s = 117\,cm$, $\Delta H = 55\,cm$, $\alpha_c = -0.7°$, $\beta_c = 0°$, and $\gamma_c = 0.2°$, where the subscript c denotes the angles between both stereo cameras, the predicted offset of $u_{sg} - u_{mg}$ approximates a constant of -5 and the 1-D v_{sg}-predicting table for each v_{mg} is listed in Table 3-1. By analyzing several images, it is concluded that the maximum absolute difference of u_{sg} between the real and predictive cases is about 6 pixels, and therefore the initial radius of $r = 4$ is used in the logarithmic search.

In conclusion, the 1-D look-up table of predicting the v_{sg} corresponding to the v_{mg}, and the constant offset. i.e. -5 pixels, of $u_{sg} - u_{mg}$ are both exploited to locate the initial center of the 1-D logarithmic search with the initial radius of $r = 4$, which achieves the low-cost and highly-accurate matching process.

Table 3-1 The look-up table of predicting the v_{sg} for every v_{mg}.

v_{mg}	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14
v_{sg}	-20	-20	-21	-22	-22	-23	-24	-25	-25	-26	-27	-27	-28	-29	-29
v_{mg}	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29
v_{sg}	-30	-31	-32	-32	-33	-34	-34	-35	-36	-36	-37	-38	-39	-39	-40
v_{mg}	-30	-31	-32	-33	-34	-35	-36	-37	-38	-39	-40	-41	-42	-43	-44
v_{sg}	-41	-41	-42	-43	-43	-44	-45	-46	-46	-47	-48	-48	-49	-50	-50
v_{mg}	-45	-46	-47	-48	-49	-50	-51	-52	-53	-54	-55	-56	-57	-58	-59
v_{sg}	-51	-52	-53	-53	-54	-55	-55	-56	-57	-57	-58	-59	-60	-60	-61
v_{mg}	-60	-61	-62	-63	-64	-65	-66	-67	-68	-69	-70	-71	-72	-73	-74
v_{sg}	-62	-62	-63	-64	-64	-65	-66	-67	-67	-68	-69	-69	-70	-71	-71
v_{mg}	-75	-76	-77	-78	-79	-80	-81	-82	-83	-84	-85	-86	-87	-88	-89
v_{sg}	-72	-73	-74	-74	-75	-76	-76	-77	-78	-78	-79	-80	-81	-81	-82
v_{mg}	-90	-91	-92	-93	-94	-95	-96	-97	-98	-99	-100	-101	-102	-103	-104
v_{sg}	-83	-83	-84	-85	-85	-86	-87	-88	-88	-89	-90	-90	-91	-92	-92
v_{mg}	-105	-106	-107	-108	-109	-110	-111	-112	-113	-114	-115	-116	-117	-118	-119
v_{sg}	-93	-94	-95	-95	-96	-97	-97	-98	-99	-99	-100	-101	-102	-102	-103
v_{mg}	-120	-121	-122	-123	-124	-125	-126	-127	-128	-129	-130	-131	-132	-133	-134
v_{sg}	-104	-104	-105	-106	-106	-107	-108	-109	-109	-110	-111	-111	-112	-113	-113
v_{mg}	-135	-136	-137	-138	-139	-140	-141	-142	-143	-144	-145	-146	-147	-148	-149
v_{sg}	-114	-115	-116	-116	-117	-118	-118	-119	-120	-120	-121	-122	-123	-123	-124
v_{mg}	-150	-151	-152	-153	-154	-155	-156	-157	-158	-159	-160	-161	-162	-163	-164
v_{sg}	-125	-125	-126	-127	-127	-128	-129	-130	-130	-131	-132	-132	-133	-134	-134
v_{mg}	-165	-166	-167	-168	-169	-170	-171	-172	-173	-174	-175	-176	-177	-178	-179
v_{sg}	-135	-136	-137	-137	-138	-139	-139	-140	-141	-141	-142	-143	-144	-144	-145
v_{mg}	-180	-181	-182	-183	-184	-185	-186	-187	-188	-189	-190	-191	-192	-193	-194
v_{sg}	-146	-146	-147	-148	-148	-149	-150	-151	-151	-152	-153	-153	-154	-155	-155
v_{mg}	-195	-196	-197	-198	-199	-200	-201	-202	-203	-204	-205	-206	-207	-208	-209
v_{sg}	-156	-157	-158	-158	-159	-160	-160	-161	-162	-162	-163	-164	-165	-165	-166
v_{mg}	-210	-211	-212	-213	-214	-215	-216	-217	-218	-219	-220	-221	-222	-223	-224
v_{sg}	-167	-167	-168	-169	-169	-170	-171	-172	-172	-173	-174	-174	-175	-176	-176
v_{mg}	-225	-226	-227	-228	-229	-230	-231	-232	-233	-234	-235	-236	-237	-238	-239
v_{sg}	-177	-178	-179	-179	-180	-181	-181	-182	-183	-183	-184	-185	-186	-186	-187
v_{mg}	-240	-241	-242	-243	-244	-245	-246								
v_{sg}	-188	-188	-189	-190	-190	-191	-192								

3.6 Discrimination of Obstacle and Ground Boundaries

After the process of boundary detection in Fig. 3.1 is performed, the boundaries are generated one by one, and the flow proceeds to the discrimination process. In this section it is ready to propose the method to distinguish if a given boundary is an obstacle boundary or not.

Since the given boundary in the road image is quasi-horizontal, there may be another boundary above the given boundary. Hence the given boundary and its corresponding top boundary in the main image are called the bottom and top boundaries, namely B_{mb} and B_{mt}, respectively, as illustrated in Fig. 3.15. A simple criterion to judge if the bottom boundary is the obstacle boundary is to judge if its corresponding top boundary is located above the vanishing line. If most of pixels within the top boundary are located above the vanishing line or even if there exists no top boundary with respect to the bottom boundary, then the bottom boundary is exactly the obstacle boundary.

Fig. 3.16 shows the flowchart of the discrimination process. If most of pixels within the top boundary are below the vanishing line, it is necessary to distinguish the bottom boundary by extra methods.
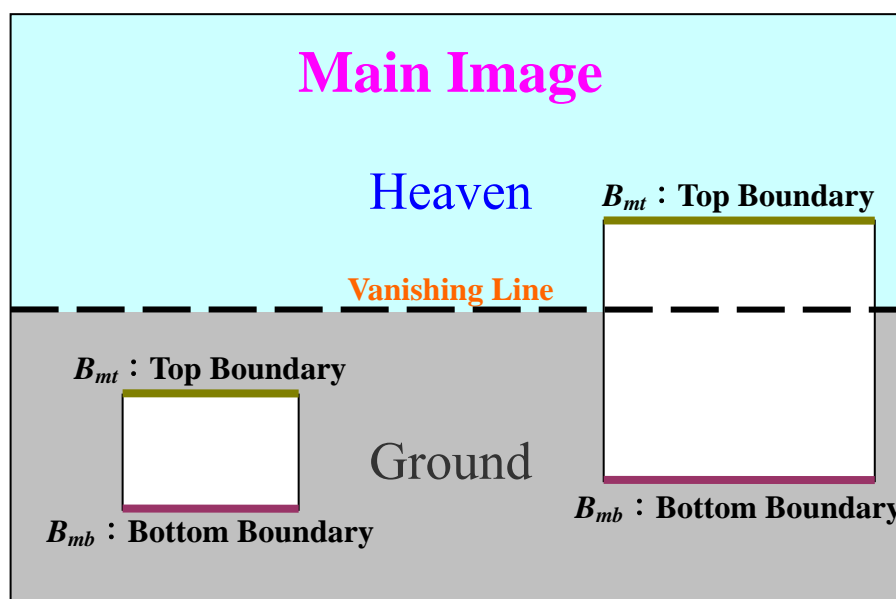
Fig. 3.15 The bottom boundary and its corresponding top boundary.

CHAPTER 3 GENERIC OBSTACLE DETECTION

At first, the calibration process is performed. Since the obstacle boundary is the interconnection between the ground and obstacles, it is believed that the bottom boundary is located on the ground no matter what it is. Therefore, its corresponding bottom boundary in the sub image can be found out by the logarithmic search, and then the pitch and roll angles can be determined, as described in Section 3.5.

```
                        ┌──────────┐
                        │  START   │
                        └──────────┘
                             │
                             ▼
              ┌───────────────────────────────┐
              │   Top Boundary Detection      │
              │           ( B_mt )            │
              └───────────────────────────────┘
                             │
                             ▼
                        ╱ Large Heaven ╲
                       ╱     Size ?     ╲
                       ╲                ╱
                        ╲              ╱
                             │
                             ▼
                     ┌───────────────┐
                     │  Calibration  │
                     └───────────────┘
                             │
                             ▼
              ┌───────────────────────────────┐
              │      Determination of         │
              │      Two Predicted            │
              │      Boundaries               │
              │    in the Sub Image,          │
              │   i.e.  B_st^g  and  B_st^o   │
              └───────────────────────────────┘
                             │
                             ▼
```

Decision: $Correlation(B_{mt}, B_{st}^o) > Correlation(B_{mt}, B_{st}^g)$?

Obstacle Boundary

Ground Boundary

END
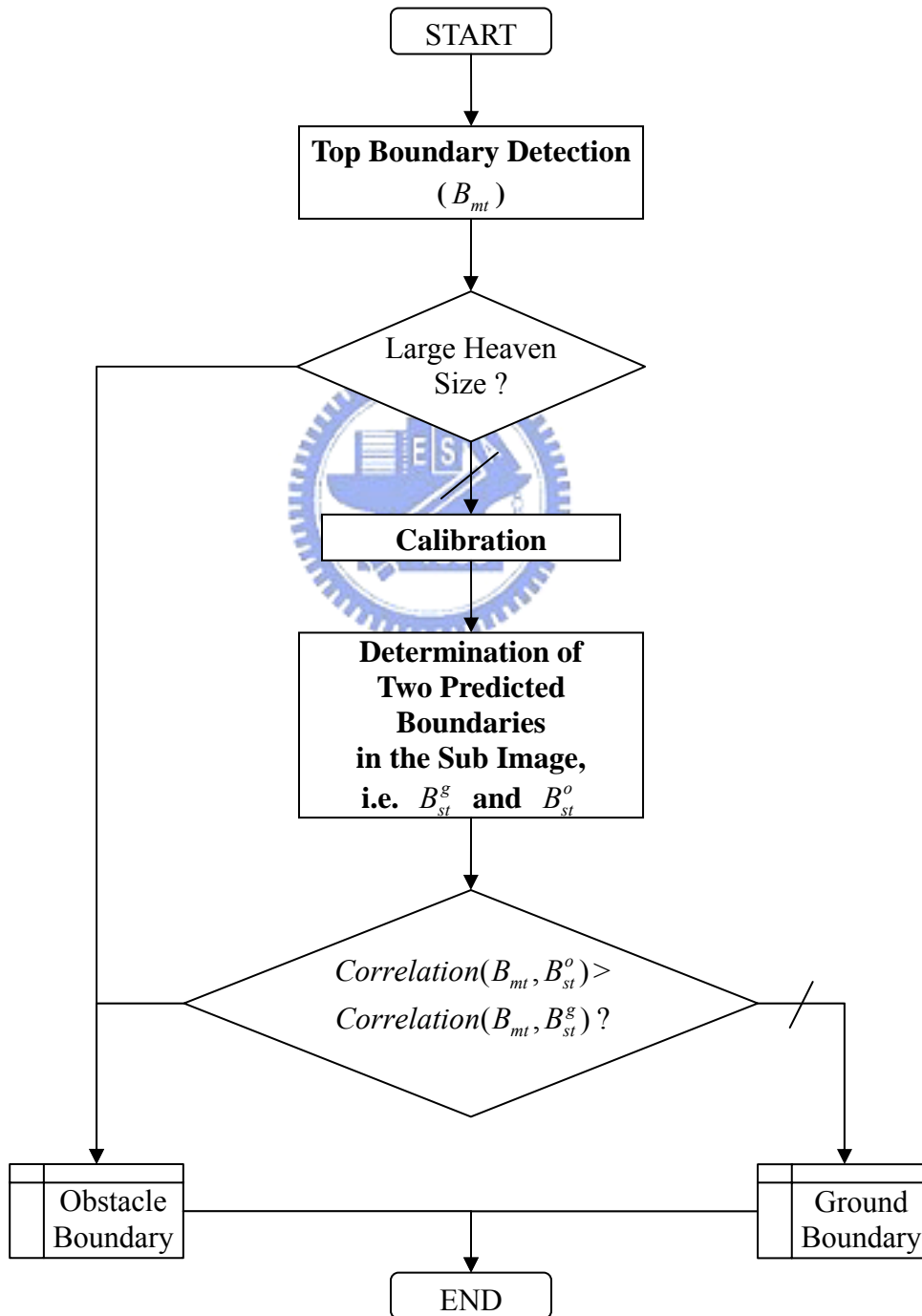
Fig. 3.16 The flowchart of the discrimination process.

CHAPTER 3 GENERIC OBSTACLE DETECTION

If the bottom boundary is the ground boundary, its corresponding top boundary is another bottom boundary; in words, the top boundary is located on the ground, which satisfies the condition of $Z_r = 0$. Thus the corresponding top boundary in the sub image, namely B_{st}^g, is on the ground and can be predicted by applying (2-68) to (2-61) and (2-62), as presented in Section 2.3.3 (Part A).

If the bottom boundary is the obstacle boundary, its corresponding top boundary must be not the ground boundary. In this case, the top and bottom boundaries could belong to the same obstacle, and the corresponding pixels within these two boundaries satisfy the condition of the same X_r. Hence the corresponding top boundary in the sub image, namely B_{st}^o, belongs to the obstacle and can be predicted by substituting (2-76) into (2-61) and (2-62), as presented in Section 2.3.3 (Part C).

The correlation between two boundaries of the same size, represented as $Correlation(B1, B2)$, is defined as the *signed square normalized correlation coefficient* of their corresponding blocks of neighbor pixels, as described in (3-20). The neighbor block of the boundary is the zone enclosed by two vertical lines and two parallel shifting boundaries, as illustrated in Fig. 3.17. As soon as B_{st}^g and B_{st}^o are determined, $Correlation(B_{mt}, B_{st}^g)$ and $Correlation(B_{mt}, B_{st}^o)$ can be computed. Therefore the bottom boundary in the main image is referred to the obstacle boundary if $Correlation(B_{mt}, B_{st}^o) > Correlation(B_{mt}, B_{st}^g)$, or else the ground boundary.
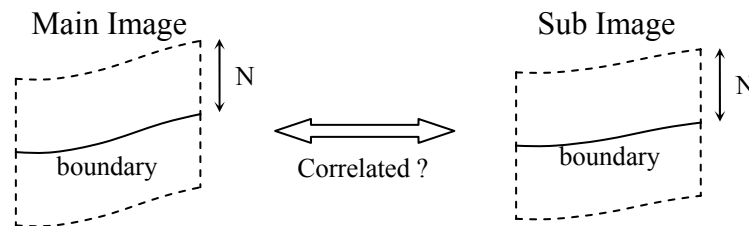
Fig. 3.17 The neighbor block of the boundary.

3.7 Motion Boundary Tracking

If the obstacles have been detected in the last frame, they will be tracked in the current frame. The detected obstacle boundaries can be classified into the *motion* and *roadside boundaries* according to their slopes. Since all roadside boundaries, i.e. the obstacle boundaries of sharp slopes, have been fitted to two left and right ones in the last frame, they will not be tracked in the current frame. Hence the tracking process is applied to only the motion boundaries.

The new boundaries in the current frame are sought out one by one in the neighborhood of a given motion boundary detected in the last frame, and the tracking succeeds if the new detected boundary and the given motion boundary are highly correlated. Since the size of the motion boundary in the last frame may be different from that of the new detected boundary in the current frame, the dimensions of their neighbor blocks may not be the same. However, due to the smooth variation of gray levels along the direction of the parallel boundary, each mean gray level along the parallel boundary of different vertical spacing can be evaluated, which forms a vector of mean gray levels, see Fig. 3.18. And the correlation between the mean vectors of two boundaries of different sizes can be computed from (3-20). In consequence, the tracking succeeds if the mean vectors of the new detected boundary and the given motion boundary are highly correlated.
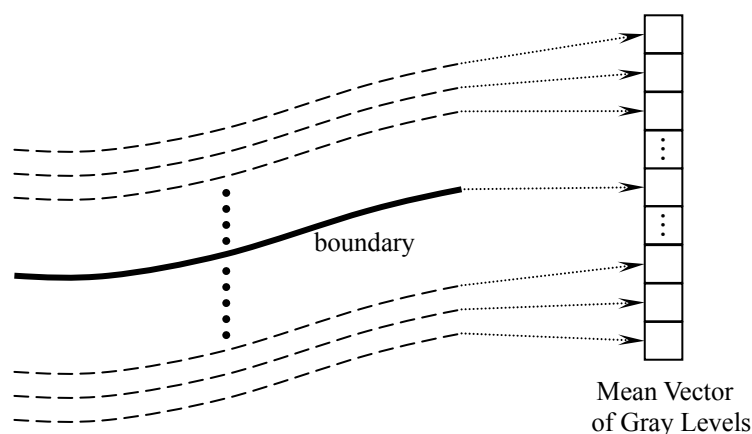
Fig. 3.18 The mean vector of the boundary.

Chapter 4 Lane Detection

In this chapter the algorithm of lane detection will be presented. Since the obstacles have been determined as mentioned in Chapter 3, the ground part of the top image will be utilized to detect the lane without the disturbance of obstacles. In addition, the lane detection system can also independently work without the aid of obstacle detection by using a single camera. The details will be proposed soon.

4.1 Overview

The algorithm of lane detection will be developed in this chapter. The target of lane detection is to determine information about the lane, such as the offset, the orientation, or the curvature. The lane detection system can immediately warn the drivers of the danger if their vehicles are straying from the correct way or can provide the controller of the automatic vehicle with the lane information.

Since the structured roads are met in most practical cases, this thesis focuses on the detection of marking roads where lane markings are painted on the road surface. Some significant features of such a structured marking road are listed as follows:

(1) The structured lane geometry.

(2) The constant lane width.

(3) The constant marking width.

(4) The higher gray levels on the markings.

(5) The continuity of the lane markings.

Based on these features, a robust lane model is fitted into the lane geometry and the detection regions of interest (ROIs) are predicted and specified to narrow the searching area of lane markings of a constant width. Due to the constant lane width, the 3-D lane geometry

CHAPTER 4 LANE DETECTION

can be reconstructed by the computer vision.

In addition, the lane parameters of the last frame can be exploited to assist in detecting the lane in the current frame since the variation of the lane between two successive frames is assumed slight according to the continuity. The flowchart of lane detection is displayed in Fig. 4.1. The process of lane detection is initially performed in the single mode without any lane information. After that, the lane parameters are updated, and the flow enters the successive mode to detect the lane by using the updated knowledge of the last frame. The process is repeated in the successive mode until the detection fails, and then the flow goes back to the single mode to try detecting once more. The details will be presented later.
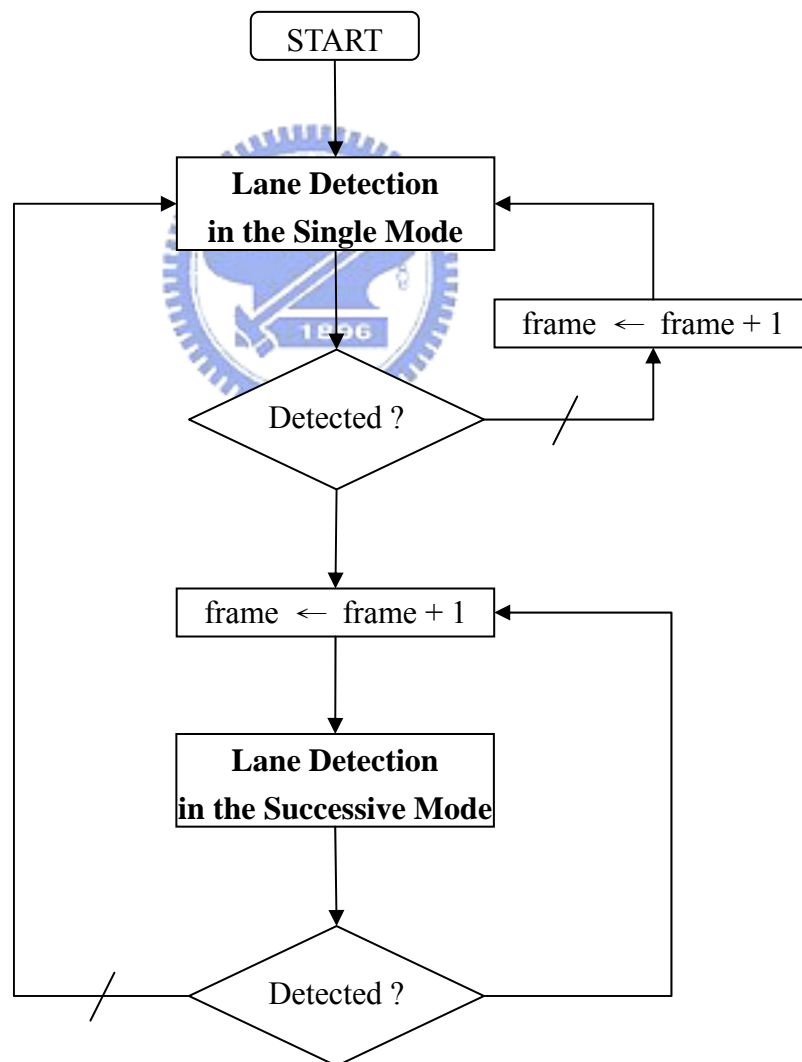
Fig. 4.1   The flowchart of lane detection.

4.2 Geometric Lane Model

Since the lane model can stabilize the detection results against the interferences such as shadows, textures, or other vehicles [12-14], a global geometric lane model is used in the lane detection system in this thesis. Such a lane model can also be utilized to predict where the lane markings are by computer vision, which thus increases the accuracy of lane detection.

4.2.1 Parabolic polynomial

Since the lane boundaries on the structured marking road can be represented by the arcs of small curvatures, a parabolic polynomial of the form

$$X_r = k \cdot Y_r^2 + m \cdot Y_r + b \tag{4-1}$$

is applied to the global geometric lane model. And consequently, the topic of lane detection is reduced to the issue of determining the coefficients (k, m, b).

Some assumptions are made here that β approximates to zero and that θ is the angle included by the inclinations of the camera and the road, as mentioned in Section 2.2.1. Therefore, the equations of (2-25) throughout (2-30) still hold true.

Substituting (2-28) and (2-29) into (4-1) yields

$$u = \frac{k \cdot e_u \cdot e_v \cdot H}{e_v \cdot m_\theta - v} + m \cdot e_u + \frac{b}{H} \cdot \frac{e_u}{e_v} \cdot (e_v \cdot m_\theta - v) \tag{4-2}$$

(4-2) represents the lane model in terms of the image coordinates (u, v) as well as the road inclination, m_θ, which is applied to the case of the non-flat road.

Let $P_L = (X_{Lr}, Y_{Lr}, Z_{Lr})$ and $P_R = (X_{Rr}, Y_{Rr}, Z_{Rr})$ be the world coordinates respectively located on the left and right sides of the lane separated by a distance equal to the lane width W. The corresponding middle point on the lane axis is $P_M = (P_L + P_R)/2$. Assume that $\overline{P_L P_R}$ is parallel to the X_r-axis so $X_{Rr} - X_{Lr} = W$ and $Y_{Rr} = Y_{Lr} = Y_{Mr}$ as well as that no torsion occurs on the road so $Z_{Rr} = Z_{Lr} = Z_{Mr}$. Let (u_L, v_L), (u_R, v_R), and (u_M, v_M) be the image

coordinates associated with P_L, P_R, and P_M, respectively. The following equations can be obtained [18].

$$\frac{1}{2}\left(u_R + u_L\right) = u_M = e_u \frac{X_{Mr}}{Y_{Mr}} \tag{4-3}$$

$$u_R - u_L = e_u \frac{X_{Rr} - X_{Lr}}{Y_{Mr}} = e_u \frac{W}{Y_{Mr}} \tag{4-4}$$

$$v_R = v_L = v_M = e_v \frac{Z_{Mr} - H}{Y_{Mr}} \tag{4-5}$$

Furthermore, (4-3) throughout (4-5) can also induce

$$X_{Mr} = \frac{u_M \cdot W}{u_R - u_L} \tag{4-6}$$

$$Y_{Mr} = e_u \frac{W}{u_R - u_L} \tag{4-7}$$

$$Z_{Mr} = H + \frac{v_M \cdot W}{u_R - u_L} \cdot \frac{e_u}{e_v} \tag{4-8}$$

Substituting (4-6) and (4-7) into (4-1) yields

$$u_M\left(u_R - u_L\right) = \left[k \cdot e_u^{\,2} \cdot W\right] + \left[m \cdot e_u\right] \cdot \left(u_R - u_L\right) + \left[\frac{b}{W}\right] \cdot \left(u_R - u_L\right)^2 \tag{4-9}$$

Both (4-2) and (4-9) represent the lane model in the image coordinate system. (4-2) is available if m_θ is given. (4-9) results from the assumption of the constant lane width, i.e. W. In fact, both m_θ and W can not be determined exactly in the various environments. Hence, a combination of (4-2) and (4-9) is made to predict the lane tendency and to result in the calibration on both m_θ and W. The details will be presented later.

4.2.2 Prediction of lane tendency

(4-9) can be rewritten as

$$U = C_{xy0} + C_{xy1} \cdot \Delta u + C_{xy2} \cdot \Delta u^2 \tag{4-10}$$

where

$$C_{xy0} = k \cdot e_u^{\,2} \cdot W \tag{4-11}$$

$$C_{xy1} = m \cdot e_u \tag{4-12}$$

$$C_{xy2} = \frac{b}{W} \tag{4-13}$$

$$U = u_M \left(u_R - u_L \right) \tag{4-14}$$

$$\Delta u = \left(u_R - u_L \right) \tag{4-15}$$

If some data of the pair (u_L, u_R) are given, the coefficients, C_{xy0}, C_{xy1}, and C_{xy2}, of

(4-10) can be determined by means of the weighted-least-squares approximation. Accordingly,

the unknowns (k, m, b) of (4-2) can be further confirmed so that the lane tendency in the

image domain can be obtained from (4-2).

On the other hand, since u is a function of v in (4-2), it is denoted as $u = f(v)$. The

first-order Taylor polynomial for f in powers of $(v - v_e)$ is represented as

$$\tilde{f}(v) = f(v_e) + f'(v_e) \cdot (v - v_e) \tag{4-16}$$

where

$$f'(v_e) = \frac{k \cdot e_u \cdot e_v \cdot H}{\left(e_v \cdot m_\theta - v_e \right)^2} - \frac{b}{H} \cdot \frac{e_u}{e_v} \tag{4-17}$$

is the first derivative of f at v_e.

If few pairs of (u_L, u_R) are given, the parameters (k, m, b) evaluated from (4-10) may

be inaccurate, so that there is a certain error in the lane tendency in the image domain

predicted from (4-2). Nevertheless, few data still contribute the information to the lane

tendency around them. If the lane tendency in a small specific region of the image is just

expected, (4-16) can be performed to obtain the approximation by expanding f about the

coordinate v_e close to the specific region of the image.

Eventually, note that both left and right sides of the lane can also be predicted from (4-16)

by replacing b with $b \mp W/2$, respectively, i.e.

$$b_i = \begin{cases} b - W/2 & ,if \ i = L \ \text{(left side)} \\ b + W/2 & ,if \ i = R \ \text{(right side)} \end{cases} \tag{4-18}$$

4.3 Marking Detection

The task of marking detection is to detect such marking pixels lying on both sides of the lane in the road image. The lane markings can be characterized by two intrinsic factors:

(1) The gray levels of the markings are greater than those of the road surface. There exist the sharper edges between the markings and the road surface, so as to produce the higher gradients located at the edges. Since only the vertical edges are interesting, the 3×3 mask shown in Fig. 4.2 is used to compute the gradients. Notice that the maximum gradients are located at the darker pixels.

(2) All widths of the markings are thought constant in the world coordinate system since they are artificially painted on the road surface. An example is demonstrated in Fig. 4.3 Of course the marking widths could slightly vary in different areas, but all of them in a certain zone can be considered the same. In general, the marking width ranges from 10 cm to 30 cm, and is referred to 20 cm in this thesis. Since the lane markings will be detected in the image coordinates, the constant marking width M_W in the world domain can be transformed into its corresponding width m_i in the image domain from (2-32), as described in Section 2.2.2. See Fig. 4.3.

1	-2	1
1	-2	1
1	-2	1

Fig. 4.2   The 3×3 mask for determining the gradients of the vertical edges. Note that the maximum gradients will be located at the darker pixels.

(a)   The original road image.

(b)   The marking in the image coordinate system.

m_i

(c)   The top view of (a).

(d)   The marking in the world coordinate system. Its width is *constant*.

M_W
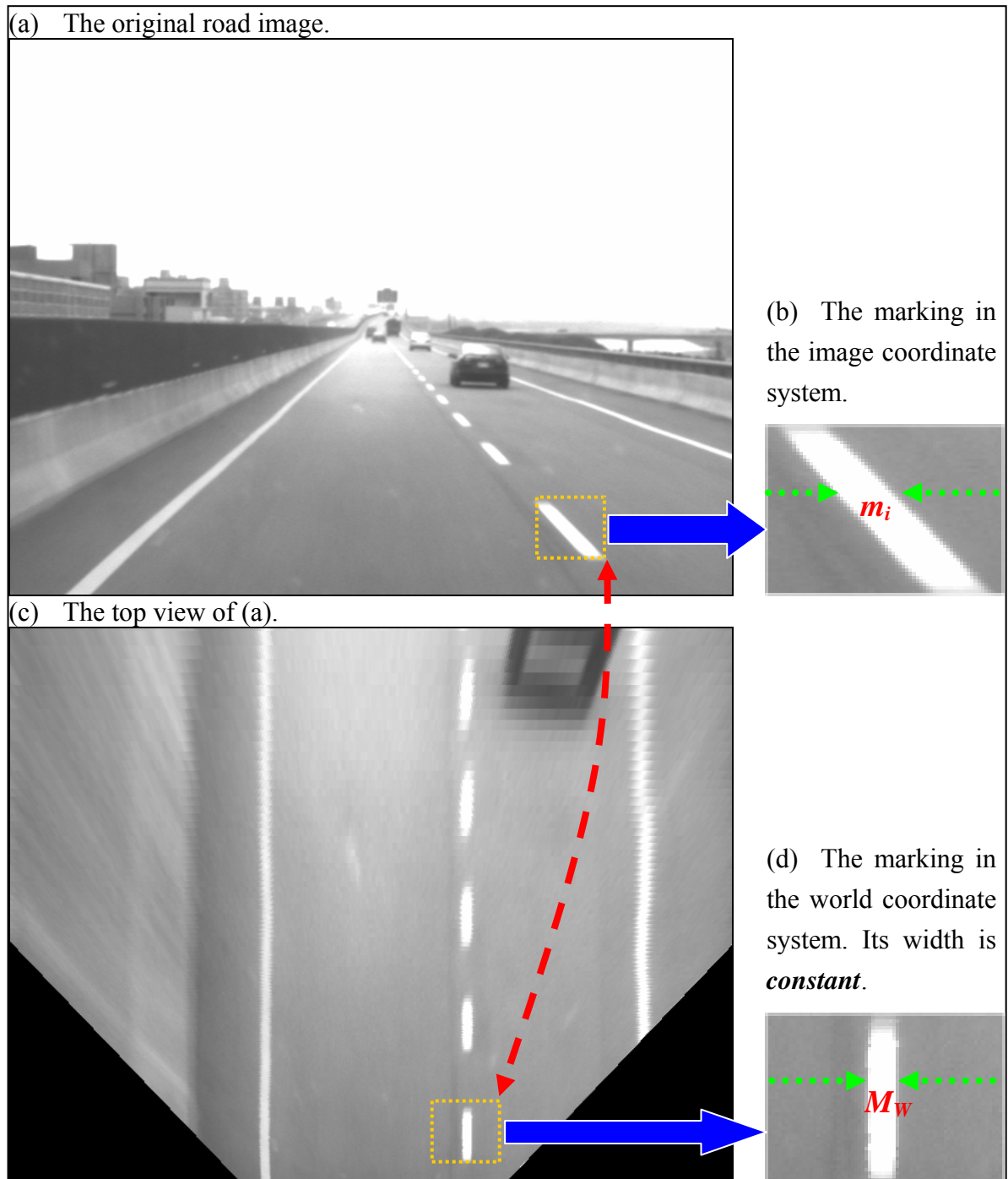
Fig. 4.3   The constant marking width in the world coordinate system. (a) The original road image. (b) The marking in the image coordinate system. (c) The top view of (a). (d) The marking in the world coordinate system. Its width M_W is approximately constant.

CHAPTER 4 LANE DETECTION

Based on the intrinsic nature of the lane markings, the following steps are performed to detect the marking, as illustrated in Fig. 4.4.

(a) Fig. 4.4 (a) shows the marking with the greater intensity and its horizontal profile for a given scanning line. It is clear that the intensity of the marking is greater than that of the road surface.

(b) Due to the higher brightness on the marking, the detection is based on the determination of horizontal dark-light-dark (DLD) intensity transitions [16]. In this thesis the point M is said to be situated at the location of the DLD-transition if its intensity I_M is greater than those at its horizontal left and right neighbors by a distance $m_i/2$, see Fig. 4.4 (b). For a given scanning line, this process searches for the location of the DLD-transition until one transition is found out or the scanning arrives at its ending. Go to step (c) if one transition is determined, or else exit.

(c) Two maximum gradients G_L and G_R within the intervals of $[M - m_i/2, M)$ and $(M, M + m_i/2]$ are determined at the points L and R respectively by the mask in Fig. 4.2, and thus enclose a possible marking region (L, R), as illustrated in Fig. 4.4 (c).

(d) Go to step (e) if the distance between L and R is greater than the threshold $\overline{LR}_{th}$, or else go back to step (b). The threshold $\overline{LR}_{th}$ is related to the minimum possible marking width in the image domain, and usually it is the half of m_i.

(e) If the mean intensity $\bar{I}$ within (L, R) is greater than I_L and I_R respectively, then the marking is determined and thus return the center of (L, R). If not, there could be some deeper valleys within (L, R) as figured in Fig. 4.4 (e). In the unsuccessful case, the flow goes back to step (c) in order to determine a new possible marking region. A new maximum gradient is detected within (L, M) to replace G_L if the mean intensity within (L, M) is less than that within (M, R), or else it is detected within (M, R) to replace G_R, which yields a new possible marking region. Proceeding in a similar fashion, the process will exactly evaluate the marking for a given scanning line.

(a)

The marking with the greater intensity than that on the road surface.

The horizontal profile of the ideal marking for a given scanning line.

(b)

$m_i/2$ $m_i/2$

$I_{M-m_i/2}$ I_L I_M I_R $I_{M+m_i/2}$

$M-m_i/2$ L M R $M+m_i/2$

Searching for the point M satisfying that $I_M > I_{M-m_i/2}$ and $I_M > I_{M+m_i/2}$.

(c)

G_L G_R

$M-m_i/2$ L M R $M+m_i/2$

Two maximum gradients G_L and G_R within the intervals of $[M-m_i/2, M)$ and $(M, M+m_i/2]$ are located respectively, and thus enclose a region (L, R).

(d)

No

$\overline{LR} > \overline{LR}_{th}$?

Yes

No

L M R

(e)

$\bar{I} > I_L \wedge \bar{I} > I_R$?

Yes

L M R
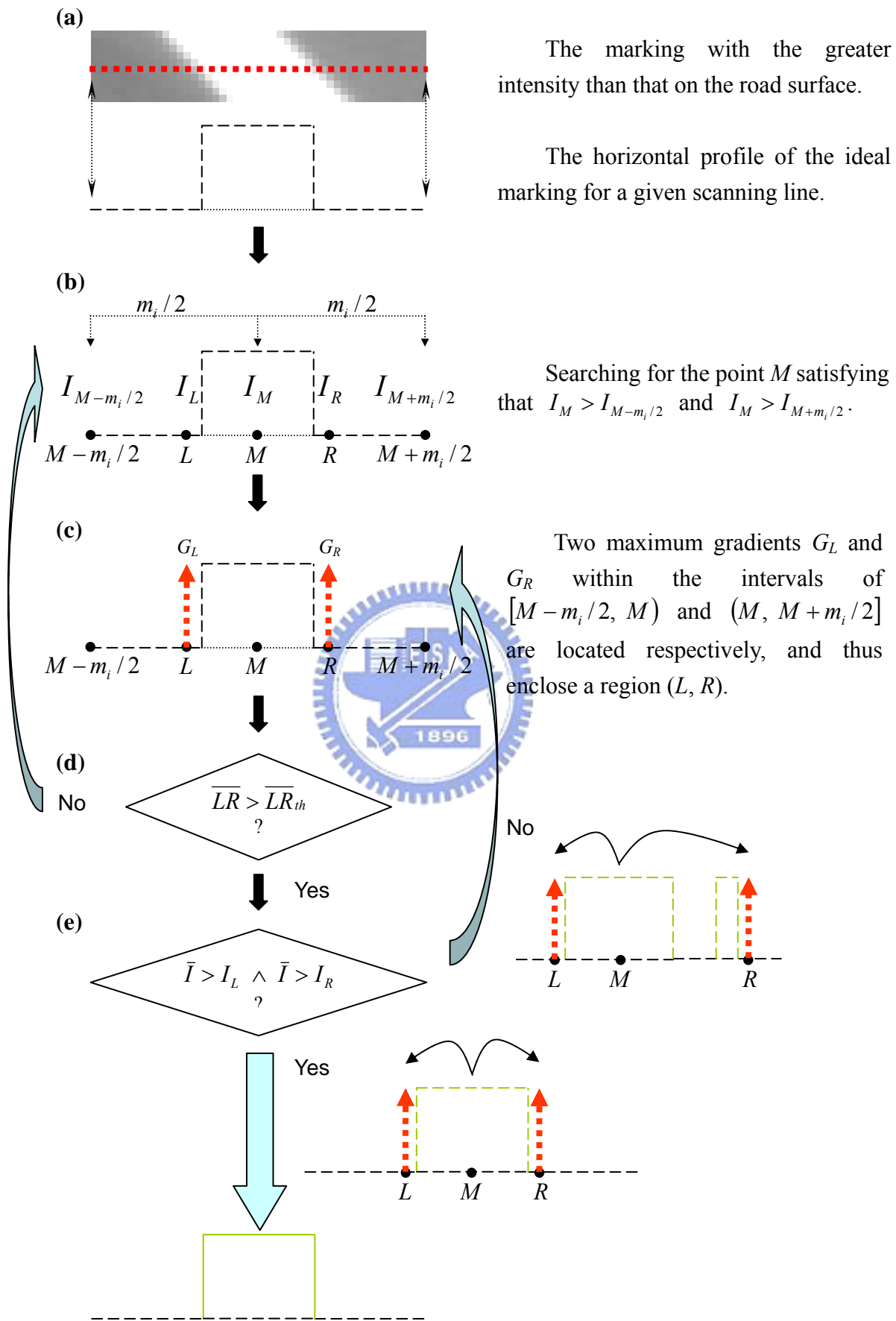
Fig. 4.4   Steps of the marking detection.

4.4 Lane Detection in the Single Mode

4.4.1 Overview

Given a single frame, the topic in this procedure is to determine the lane without the aid of the last frame. In the beginning, there is no information about the lane. However, according to (4-1), the lane has been modeled as a quadratic polynomial with the coefficients (k, m, b), whose probable ranges are as follows [14, 18]:

$$
\begin{aligned}
k &: -\frac{1}{600} \sim \frac{1}{600} \quad (1/m) \\
m &: -\tan(0.09) \sim \tan(0.09) \\
b &: -3.75 \sim 3.75 \quad (m)
\end{aligned}
\tag{4-19}
$$

Fig. 4.5 displays the probable marking ranges of both left and right sides of the lane. Therefore, the area of marking detection at the initial phase is restricted to these two ROIs.
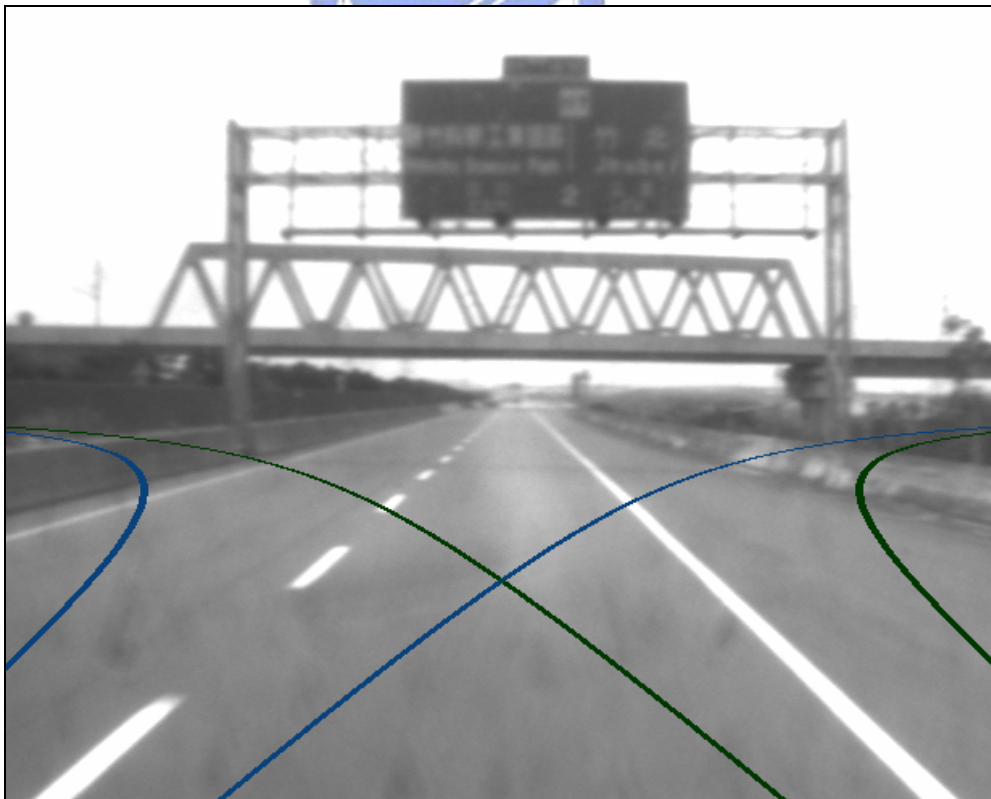
Fig. 4.5　The possible ranges of the markings on both sides of the lane at the initial state.

Since the probable ranges in the bottom of the image are the narrowest, the image plane is divided into n zones ordered according to the direction of the v-axis and the size of the first zone is greater, as shown in Fig. 4.6. The marking detection is performed zone by zone, from bottom to top in the image.

As soon as the left and right markings in a certain zone are determined, an estimate of (k, m, b) is evaluated from (4-10), and the probable left and right markings in the next zone can be predicted by (4-16) and (4-18); moreover, the ROIs can be set up in the detection zone to narrow the searching area. After the same process is iterated for each zone from bottom to up, the markings on both sides of the lane are found out, and the parameters (k, m, b) can be determined. Fig. 4.7 demonstrates the detection results and the predicted ranges for each zone. The details will be described in the following.

Fig. 4.6   The image is divided into n zones, and the markings are detected from bottom to up.
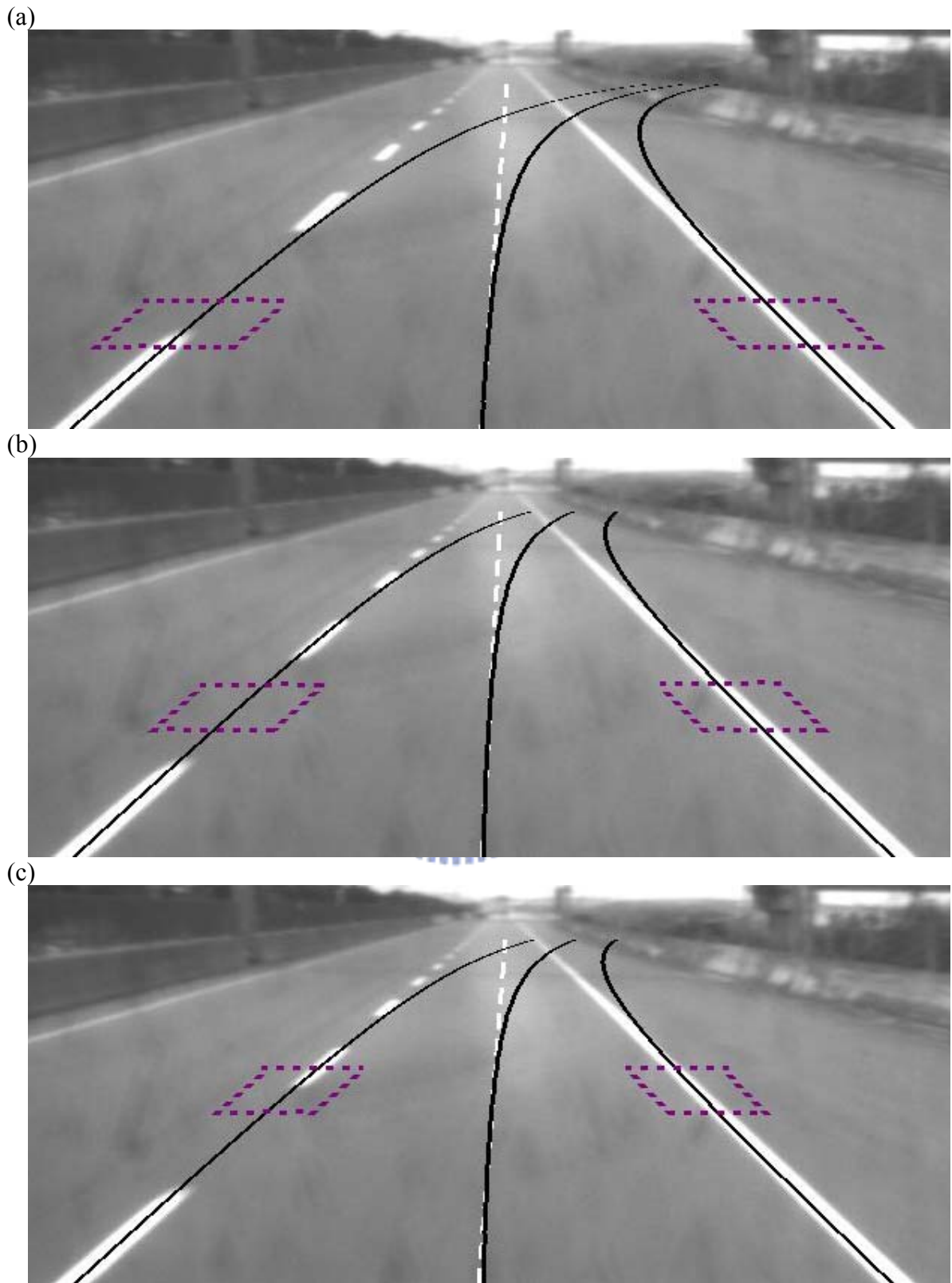
(a)

(b)

(c)

Fig. 4.7   (a)~(f) are the intermediate phases where the zones are detected from bottom to up, respectively. The black solid curve comes from (4-2) while the white dashed line is the approximation of (4-16). The left and right dotted blocks are the detection regions of interest in the next zone. (a), (b), and (c) show the detection ROIs in the zones 1, 2, and 3, respectively.
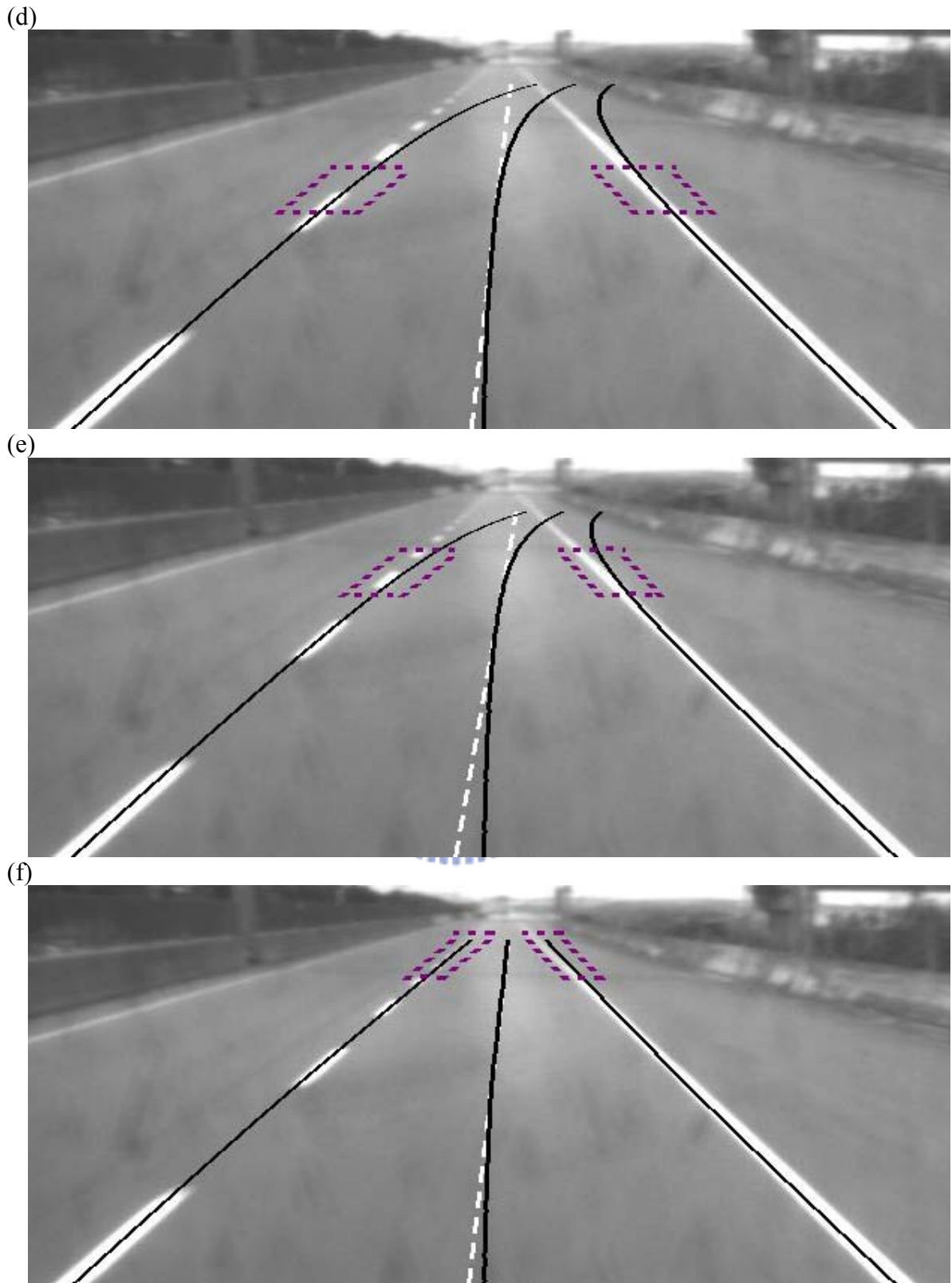
(d)

(e)

(f)

Fig. 4.7   (a)~(f) are the intermediate phases where the zones are detected from bottom to up, respectively. The black solid curve comes from (4-2) while the white dashed line is the approximation of (4-16). The left and right dotted blocks are the detection regions of interest in the next zone. (d) and (e) show the detection ROIs in the zones 4 and 5, respectively. Fig. (f) shows the final detection result, and it is obvious that the rears of the detected and predicted lines match with each other.

4.4.2 Detection flow

Fig. 4.8 shows the flowchart of lane detection in the single mode. In the beginning, the initial possible ranges are searched zone by zone, and the zone is scanned row by row, from bottom to up, to detect the markings. If both marking points on the left and right sides are found out and the distance between them is valid, this procedure is terminated after finishing the current zone. Since the constant lane width in the world domain is assumed, its corresponding width in the image domain can be obtained from (2-32). Thus the distance between the left and right detected pointes is said valid if it approximates to the constant lane width in the image coordinate system.

After the lane detection at the initial state, the next zone is located. The probable regions of the markings in this zone can be predicted by applying (4-16) and (4-18), where v_e comes from the coordinate v near the current zone. The ROIs are determined at the process, namely *Specify ROI*, to narrow the searching ranges. Both markings in these two ROIs are searched and determined. And then the detection results for both markings are processed at the procedure, namely *Decision Tree*. Both the processes of *Specify ROI* and *Decision Tree* will be interpreted in next sections. Proceeding in the same way, the lane can be confirmed after all zones are detected.

```
                          ┌──────────────┐
                          │    START     │
                          └──────────────┘
                                 │
                          ┌──────────────────┐
                          │  Lane Detection  │
                          │ at the initial phase │
                          └──────────────────┘
                                 │
                          ┌──────────────────┐
                          │   zone←zone+1    │
                          └──────────────────┘
                                 │
                          ┌──────────────────┐
                          │ Lane Prediction  │
                          └──────────────────┘
                                 │
                          ┌──────────────────┐
                          │   Specify ROI    │
                          └──────────────────┘
                                 │
                          ┌──────────────────┐
                          │ Marking Detection│
                          └──────────────────┘
                                 │
                          ┌──────────────────┐
                          │  Decision Tree   │
                          └──────────────────┘
                                 │
                          ┌──────────────┐
                          │  v ← v + 1   │
                          └──────────────┘
                                 │
                           ◇ End of v ? ◇
                                 │
                          ◇ End of zone ? ◇
                                 │
                          ┌──────────────┐
                          │     END      │
                          └──────────────┘
```
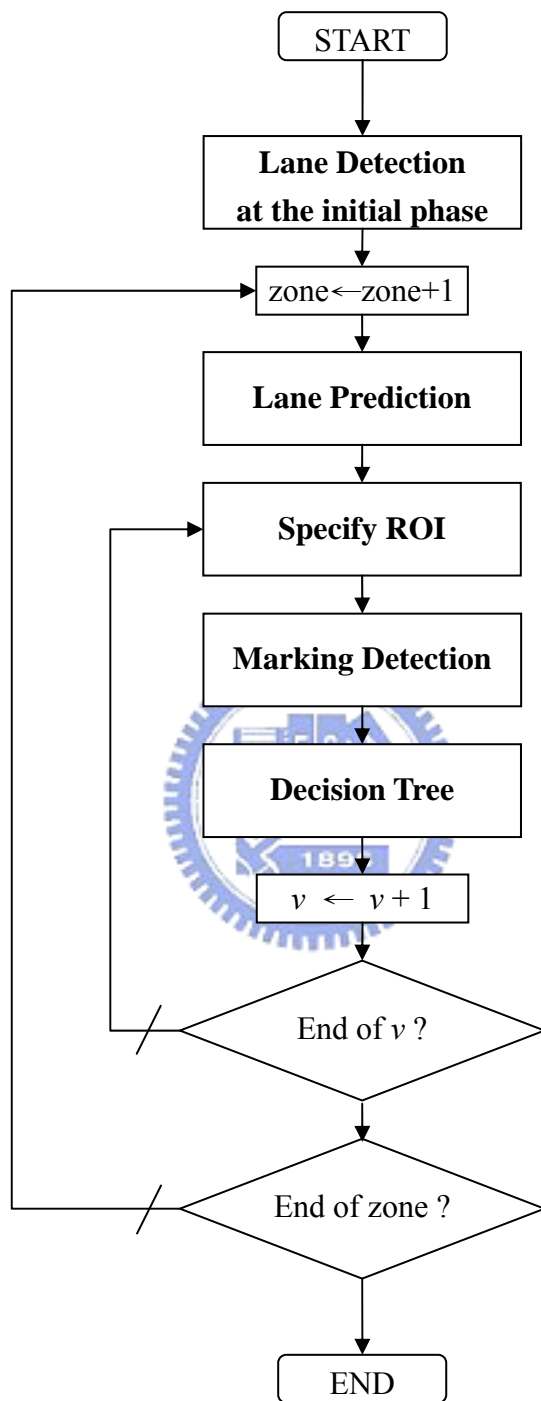
Fig. 4.8   The flowchart of lane detection in the single mode.

4.4.3 Specify the detection region of interest

This procedure is to specify the left and right regions of interest (ROIs) for the marking detection, and these ROIs depend on the parameter *pROI*, composed of properties *MAIN* and *SUB*, initialized to *MAIN*, and determined by the procedure called the *Decision Tree*.

(1) *pROI = SUB*

In this case, the left and right ROIs of the current ordinate v_i are defined as:

$$ROI = \left[u_{i-1} - \lambda_s \cdot m_i, u_{i-1} + \lambda_s \cdot m_i \right] \tag{4-20}$$

where u_{i-1} is the abscissa detected in the last ordinate v_{i-1}, m_i is the constant marking width in the image domain as described in Section 4.3, and λ_s is a constant. Since the abscissa u_i is the target to detect, its probable region is specified in the neighborhood of the last detected abscissa u_{i-1} due to the continuity of the marking.

(2) *pROI = MAIN*

In this condition, the left and right ROIs of the current ordinate v_i are defined as:

$$ROI = \left[u_i - \lambda_m \cdot m_i, u_i + \lambda_m \cdot m_i \right] \tag{4-21}$$

where the abscissa u_i corresponding to the current ordinate v_i is evaluated in the process of the lane prediction, and λ_m is a constant. Since the last abscissa u_{i-1} is not detected, a guess about u_i can come from the combination of (4-16) and (4-18), and a greater λ_m than λ_s is used due to the unknown abscissa u_{i-1} of the lane sides.

4.4.4 Decision tree

After the marking detection is performed in both left and right specified regions of interest for a given ordinate v_i, as illustrated in Fig. 4.8, one of four cases as stated below will happen. This process is to judge whether both left and right detected marking points belong to the boundaries of the lane, and to update the control parameters if true. Two control

parameters, namely *pROI* and *pMode* respectively, will be updated in this process. The control parameter *pROI* is the enumeration composed of two identifiers, namely *MAIN* and *SUB*, and it is used to decide the ROIs for the marking detection, as proposed in Section 4.4.3. The other control parameter *pMode* is the enumeration composed of four identifiers, namely *BOTH*, *LEFT*, *RIGHT*, and *NONE*, and it responses the result of detecting both left and right marking points. Fig. 4.9 illustrates the flowchart of the *Decision Tree*, and four conditions are discussed as follows:

(1) Both markings are found:

In this case, both left and right marking points are found out, and the distance between them is compared to the constant lane width in the image domain. If it is legal, the lane width in the world domain is updated and both marking points are added to the fitting data accompanied with a greater weight for the weighted-least-squares approximation.

(2) Only the left marking is found:

In this situation, only the left marking point is found out. If *pROI* = *SUB* or *pMode* = *LEFT*, proceed to the process called the *Update Left*. In the process of *Update Left*, the right marking point can be estimated as what is the left point plus the lane width; afterward both points are added to the fitting data with a smaller fitting weight, and finally *pROI* is assigned as *SUB*. The conditions of *pROI* = *SUB* or *pMode* = *LEFT* are based on the continuity of the marking, and they mean *that the last marking point is detected, and that the current marking point in the neighborhood of the last marking point is found out*; we believe two points belong to the same marking thanks to the marking continuity.

(3) Only the right marking is found:

In this condition, if *pROI* = *SUB* or *pMode* = *RIGHT*, then the left marking point is approximated as what is the right point minus the lane width, and the process of *Update Right* similar to the case (2) is driven.

(4) No marking is found:

Since no marking is found in this case, no point is added to the fitting data. We just

assign *pROI* and *pMode* as *MAIN* and *NONE*, respectively.

```
Marking Detection
```

Both ? → Valid Width ? → Update Both → *pMode = BOTH*
→ *pROI = MAIN*

Left ? → *pROI = SUB* ∨ *pMode = LEFT* → Update Left → *pMode = LEFT*
→ *pROI = MAIN*

Right ? → *pROI = SUB* ∨ *pMode = RIGHT* → Update Right → *pMode = RIGHT*
→ *pROI = MAIN*
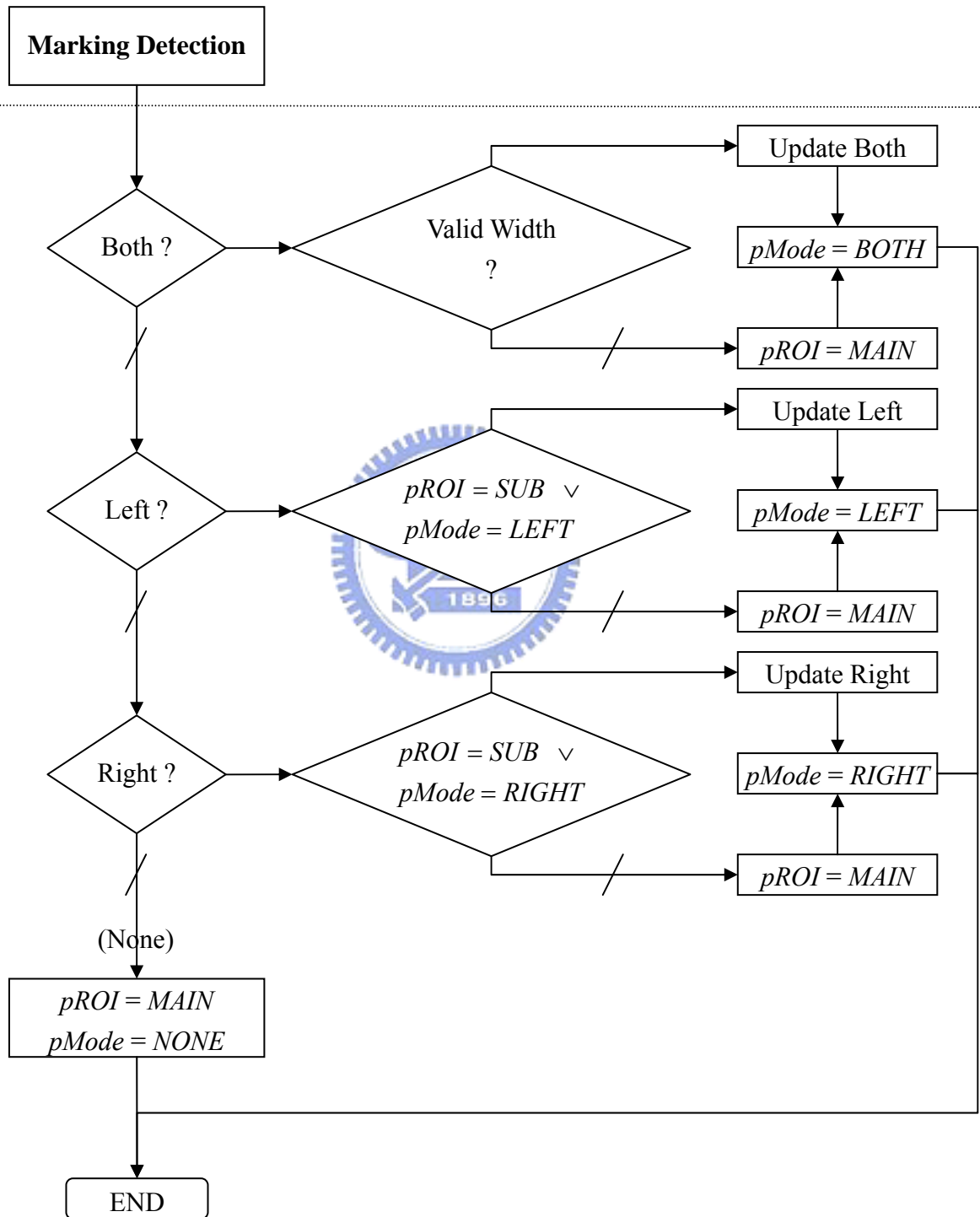
(None)

pROI = MAIN
pMode = NONE

```
END
```

Fig. 4.9   The flowchart of *Decision Tree*.

4.5 Lane Detection in the Successive Mode

The purpose in this successive mode is to detect the lane with a prior knowledge of the lane detected on the last frame. Due to the slight variation between two successive frames, the last determined lane parameters (k, m, b) can be regarded as an estimate about the lane on this frame. In addition, both estimated boundaries of the lane on the current frame can also be evaluated from the estimated lane.

```
                    ┌─────────┐
                    │  START  │
                    └─────────┘
                         │
                         ▼
              ┌─────────────────────┐
      ┌──────▶│     Specify ROI     │
      │       └─────────────────────┘
      │                  │
      │                  ▼
      │       ┌─────────────────────┐
      │       │  Marking Detection  │
      │       └─────────────────────┘
      │                  │
      │                  ▼
      │       ┌─────────────────────┐
      │       │    Decision Tree    │
      │       └─────────────────────┘
      │                  │
      │                  ▼
      │         ┌───────────────┐
      │         │  v ← v + 1    │
      │         └───────────────┘
      │                  │
      │                  ▼
      │            ╱──────────╲
      └───────────╱  End of v ? ╲
                  ╲            ╱
                   ╲──────────╱
                         │
                         ▼
                    ┌─────────┐
                    │   END   │
                    └─────────┘
```
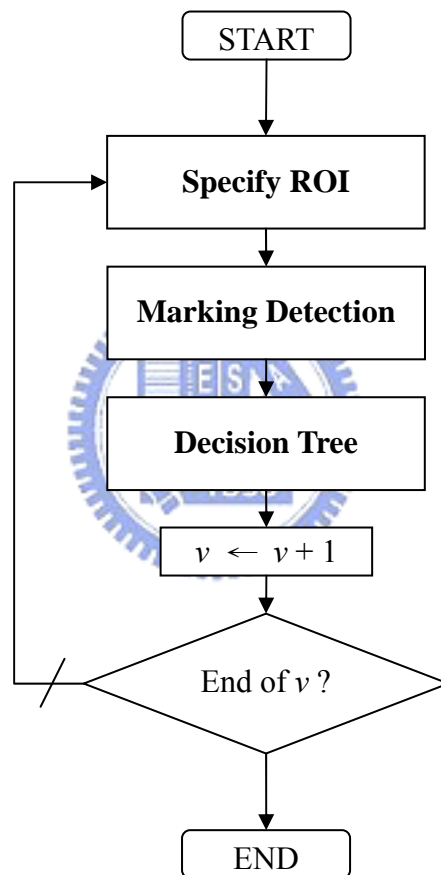
Fig. 4.10 The flowchart of lane detection in the successive mode.

Fig. 4.10 shows the flowchart of the lane detection in the successive mode. The main regions of interesting markings can be specified by the neighborhood of the estimated lane boundaries. And thus the lane on the current frame can be determined by detecting directly the main or sub ROIs in the same way as mentioned in the last section. After finishing scanning

every interesting row, all marking points in the image are found out. Since the slight variation between two successive frames is assumed, the current lane parameters (k, m, b) can, for the purpose of the robust detection, be evaluated from the data of a mixture of the current and last marking points, where the marking points on the last frame are assigned a smaller fitting weight.

The process is repeated in the successive mode until it fails, and then returns to the single mode. The detection in the successive mode is dominant since it takes the majority, and therefore, for reducing the processing time, the algorithm in the successive mode should be simple and effective as we do.

4.6 Update of Lane Parameters

The update of lane parameters involves two topics. At first, the lane tendency is necessary in order to warn the drivers in bad situations or to supply the smart vehicle with the lane information for the purpose of tracking the lane automatically. Thanks to the detection algorithm involving the prediction phase, the lane, modeled as a quadratic curve with parameters (k, m, b), is determined as soon as the detection finishes. However, the physical lane parameters, such as the offset, the orientation, or the curvature, are usually desired in the practical applications, and they will be discussed in Section 4.6.2.

On the other hand, the second topic is concerning the 3-D reconstruction of the lane. In words, the goal here is to reconstruct the lane information about the road inclination and the lane width for the next detection stage or the vehicle controller.

The algorithm of lane detection proposed in this thesis is based on the condition of the constant lane width. Nevertheless, due to the vibration in motion, the variation of the road inclination, or the illegal assumption of the constant lane width, some errors may exist in the results of the formulas deduced from the computer vision. Thus, it is necessary to calibrate the 3-D lane parameters. This will be presented in Section 4.6.1.

4.6.1 3-D reconstruction

Applying (4-7) and (4-8) to (2-27) and rearranging it by (4-15), it yields

$$v_M = [e_v \cdot m_\theta] + \left[-\frac{e_v}{e_u} \cdot \frac{H}{W} \right] \cdot \Delta u \qquad (4\text{-}22)$$

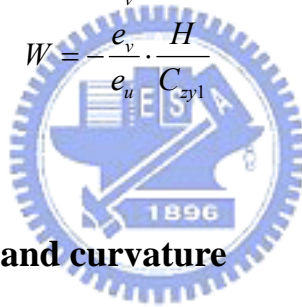Using the weighted-least-squares method, the coefficients of (4-22), i.e.

$$C_{zy0} = e_v \cdot m_\theta \qquad (4\text{-}23)$$

$$C_{zy1} = -\frac{e_v}{e_u} \cdot \frac{H}{W} \qquad (4\text{-}24)$$

can be solved. Finally, m_θ and W can be obtained from (4-25) and (4-26), respectively.

$$m_\theta = \frac{C_{zy0}}{e_v} \qquad (4\text{-}25)$$

$$W = -\frac{e_v}{e_u} \cdot \frac{H}{C_{zy1}} \qquad (4\text{-}26)$$

4.6.2 Offset, orientation, and curvature

Since the lane has been modeled as a quadratic polynomial with the coefficients (k, m, b), from the fundamental calculus it is easy to obtain the offset, orientation, and curvature of the lane, which are listed as follows:

$$Offset(Y_r) \qquad = X_r(Y_r) = k \cdot Y_r^2 + m \cdot Y_r + b \qquad (4\text{-}27)$$

$$Orientaton(Y_r) = X_r{}'(Y_r) = 2 \cdot k \cdot Y_r + m \qquad (4\text{-}28)$$

$$Curvature(Y_r) = \frac{X_r{}''}{\left(1 + X_r{}'^2\right)^{3/2}} = \frac{2 \cdot k}{\left(1 + (2 \cdot k \cdot Y_r + m)^2\right)^{3/2}} \qquad (4\text{-}29)$$

Chapter 5 Experimental Results

5.1 Results of Obstacle and Lane Detection

In the obstacle and lane detection system, two cameras are mounted top and bottom on our experimental vehicle, and both road images are captured simultaneously. Fig. 5.1 shows the results of obstacle detection on a hill road. On the other hand, Fig. 5.2 and 5.3 display the results of obstacle and lane detection when the experimental vehicle is running on the expressway and freeway with the velocities of 80 km/hr and 110 km/hr, respectively. It is clear that the vehicles on the road can be determined. In addition, the roadsides such as the median can also be marked.
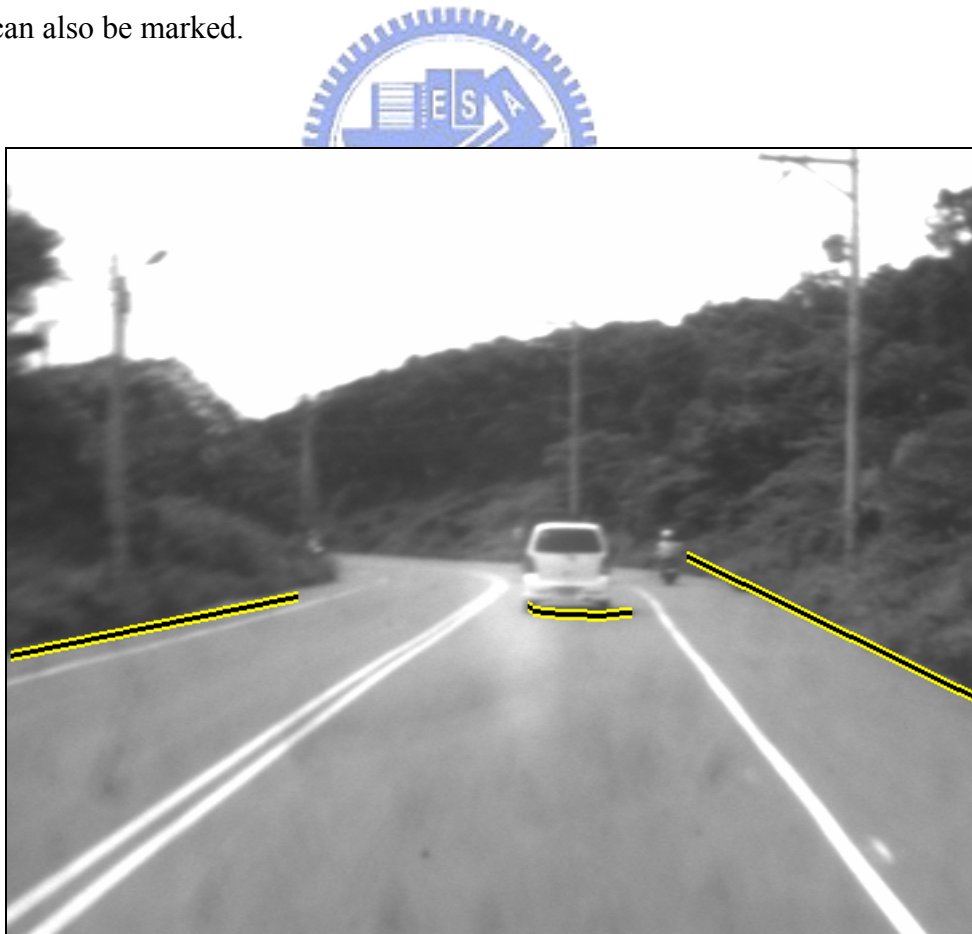

Fig. 5.1   Results of obstacle detection on a hill road.

Fig. 5.2 Results of obstacle and lane detection on the expressway.
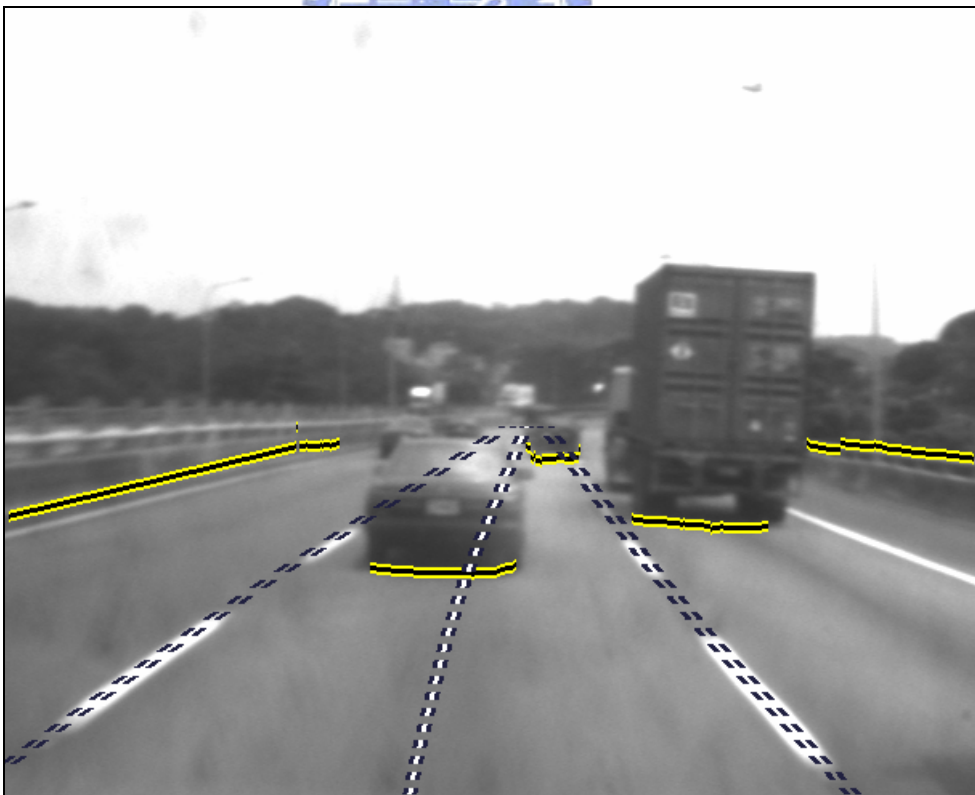

Fig. 5.3 Results of obstacle and lane detection on the freeway.

On the other hand, the lane detection algorithm proposed in this thesis can also be performed alone using a single monochromatic camera without the obstacle detection system. Several road conditions, such as the straight or crooked cases, shadows or sunlight conditions, the roads interfered with the text or vehicles, are tested, and the results are very satisfactory, as displayed in Fig. 5.4~5.8. The shadows on the road surface will result in the variation of the brightness. The gray values of the texts on the road surface are similar to those of the lane markings. In addition, the traffic in downtown is usually heavy so that the markings are often covered by vehicles. The proposed algorithm can work in all cases, even if only left or right lane side is available. On the other hand, the lane detection system can also be performed in the night or rainy environment, as shown in Fig. 5.9 and 5.10 respectively. The results demonstrate the proposed algorithm is very robust.
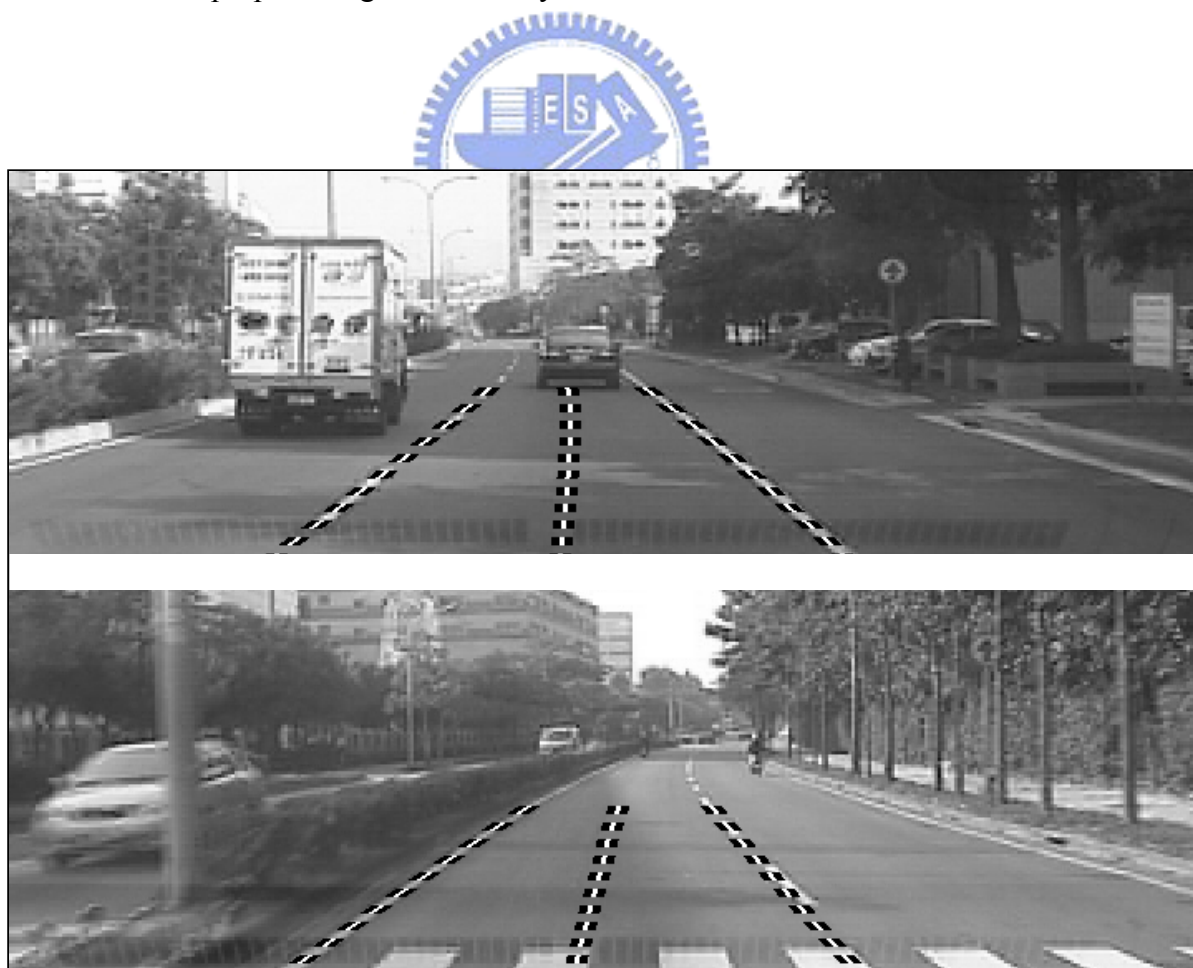
Fig. 5.4　Results of lane detection on the straight roads.
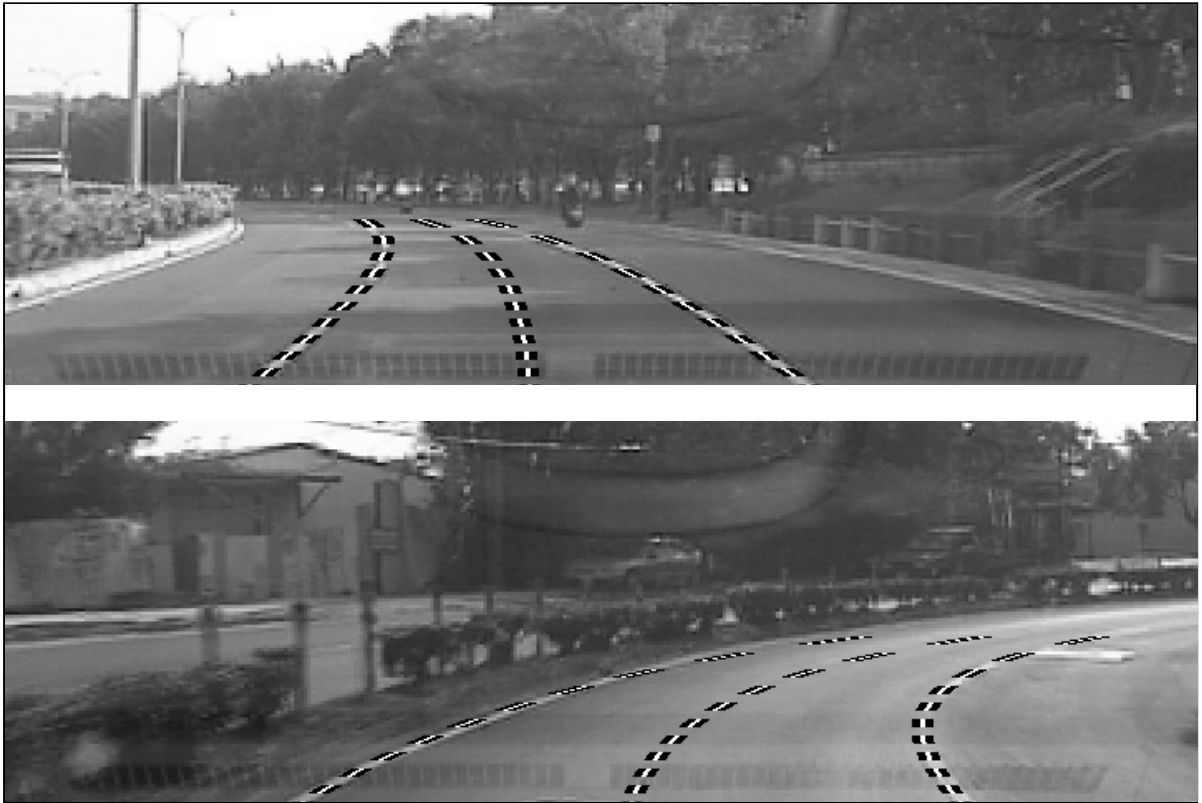
Fig. 5.5   Results of lane detection on the crooked roads.
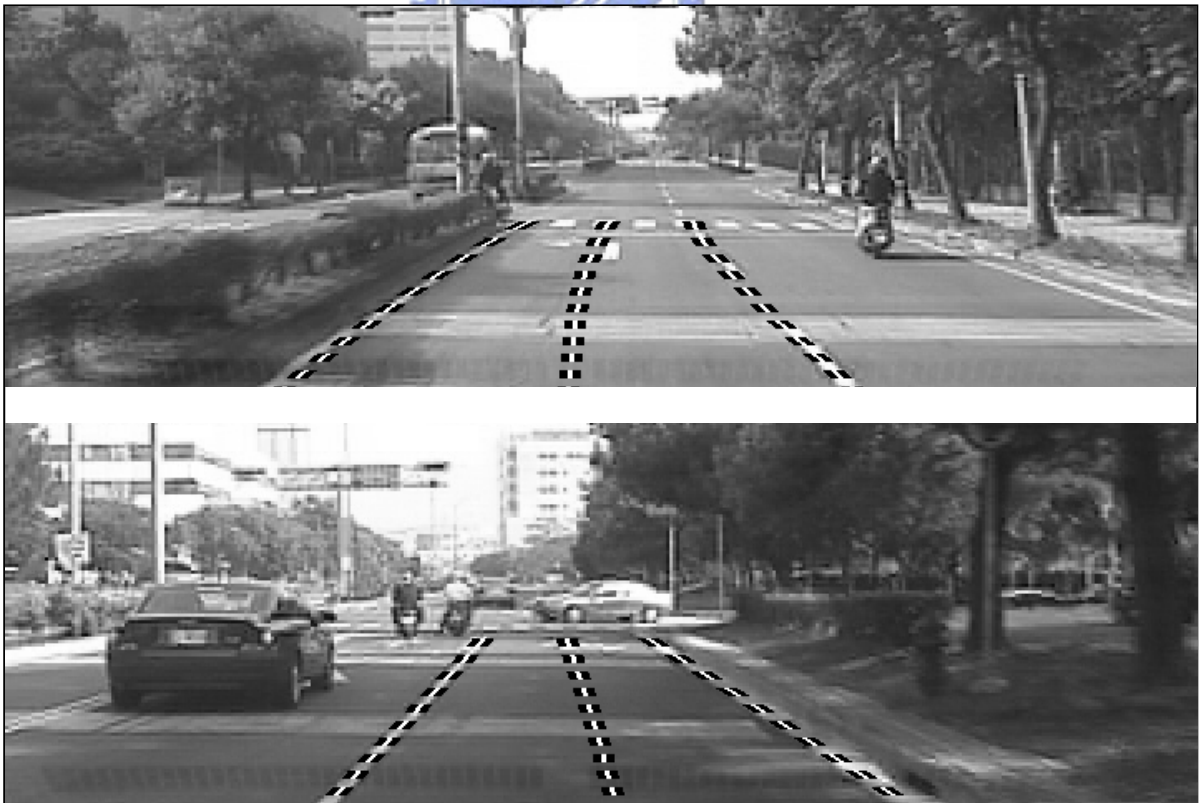
Fig. 5.6   Results of lane detection on the roads with shadows or the sunlight.
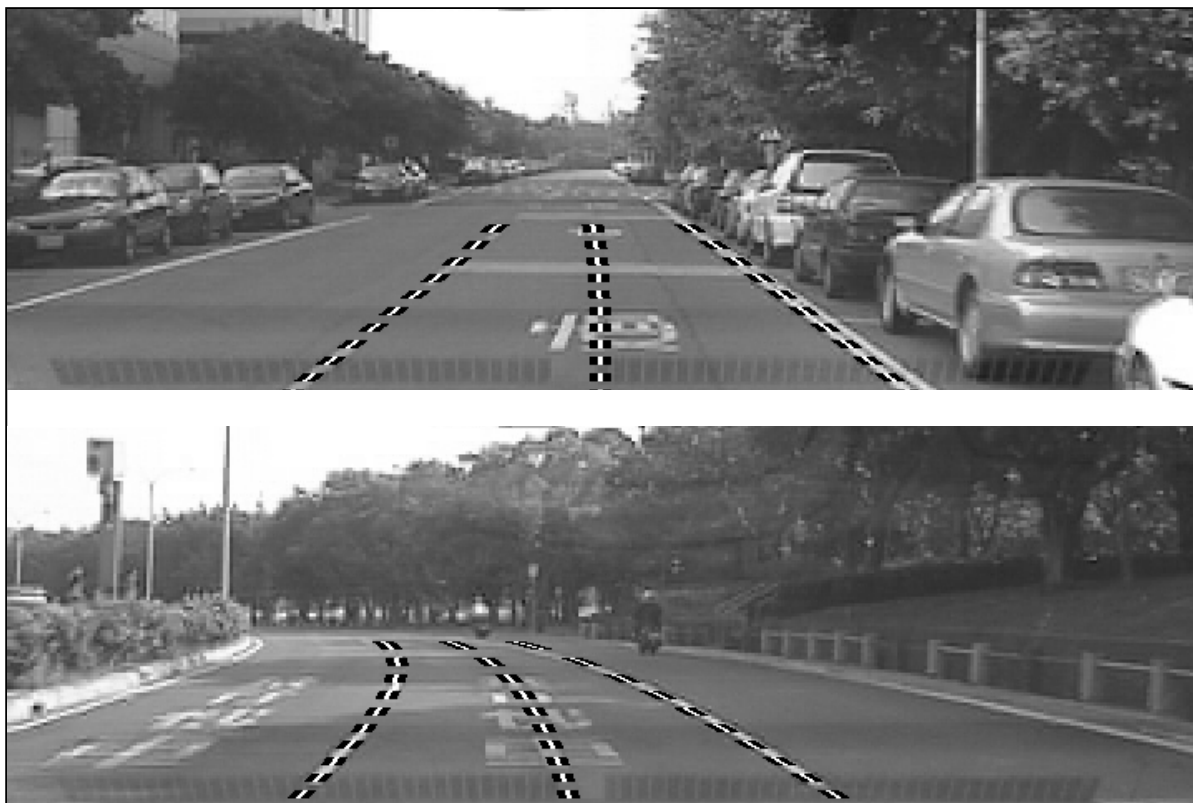
Fig. 5.7   Results of lane detection on the roads interfered with the text.

Fig. 5.8   Results of lane detection on the roads affected by the vehicles.

(a) The original road image.

(b) The detection result of (a).

Fig. 5.9 Result of lane detection on the night road. (a) the original night road image. (b) the detection result of (a).

(a) The original road image.

(b) The detection result of (a).
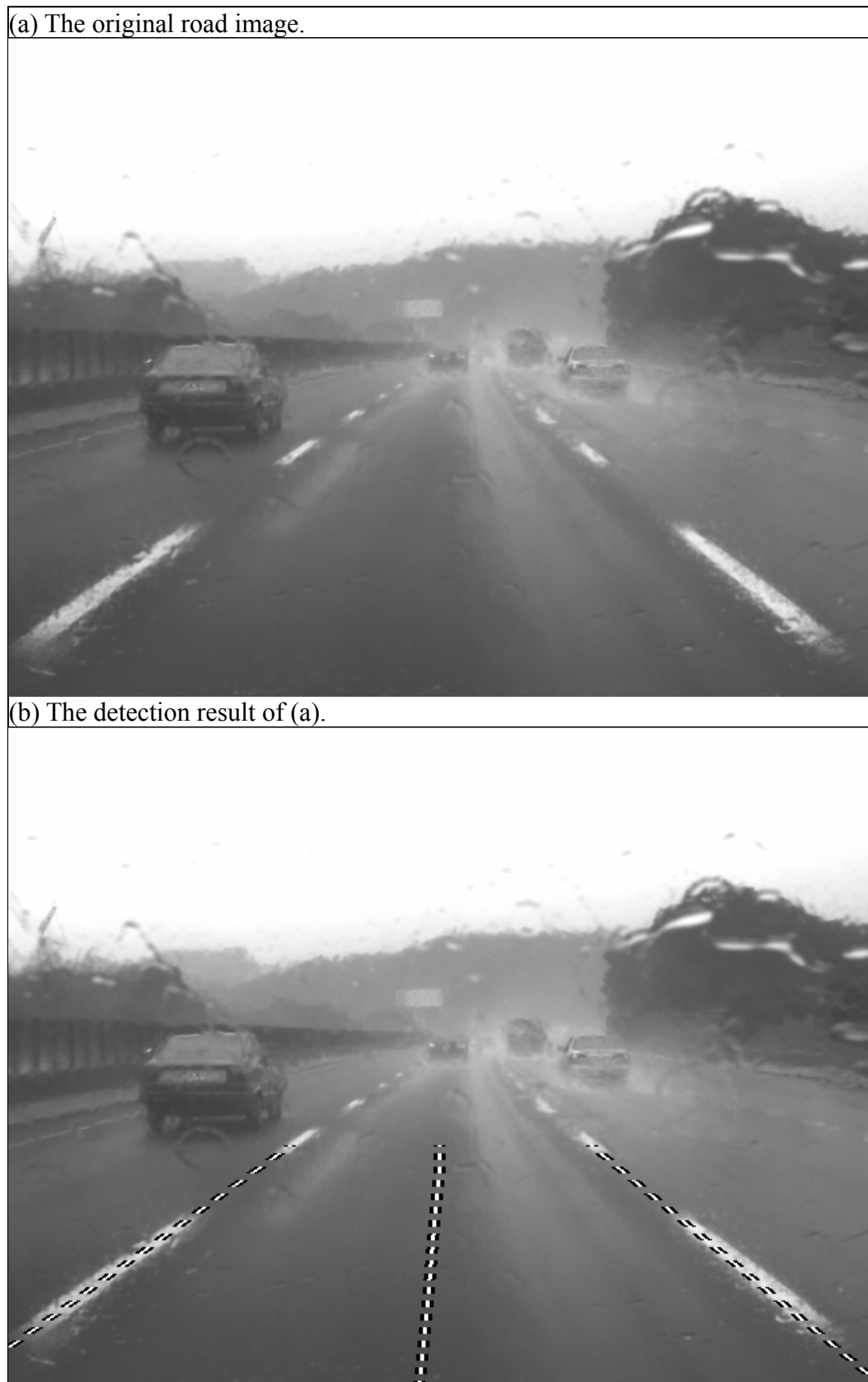
Fig. 5.10 Result of lane detection on the rainy road. (a) the original rainy road image. (b) the detection result of (a).

CHAPTER 5 EXPERIMENTAL RESULTS

Fig. 5.11 and 5.12 display the experimental results of real-time lane detection on our experimental vehicle, namely TAIWAN iTS-1 as shown in Fig. 13, running on the freeway under 110 km/hr. The road image sequence of size 644×493 is captured with the frame rate of 30 fps by the Domino Alpha 2 board and the Hitachi KP-F3 CCD camera mounted on the smart vehicle, and then the image is processed by the proposed algorithm of lane detection running on the PC platform of 2.6-GHz CPU and 512-MB RAM. The average processing time is less than 1 ms per frame. In addition, the lane detection system can be treated as the vision system of the automatic vehicle by integrating the controller of the steering wheel. This work has been implemented on the experimental car, TAIWAN iTS-1, running on the expressway and freeway with the velocities of 90 km/hr and 110 km/hr respectively. TAIWAN iTS-1 is the first smart car in Taiwan capable of hand-free driving on the real road, which verifies the practicability and robustness of the proposed lane detection system.
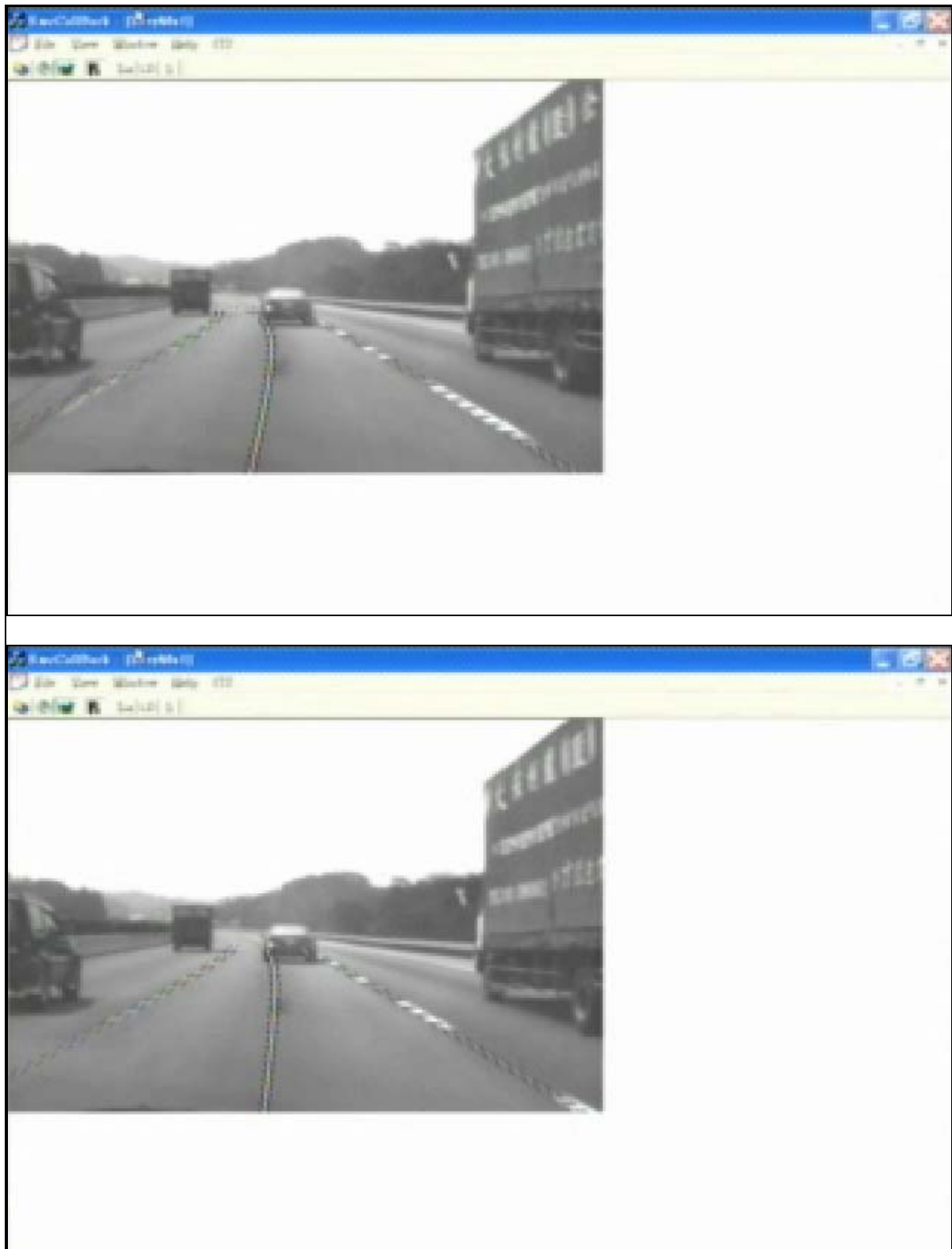
Fig. 5.11 Results of the real-time lane detection on the freeway of the sunny day.

Fig. 5.12 Results of the real-time lane detection on the freeway by night.

(a) TAIWAN *i*TS-1 is running on the freeway under 110 km/hr

(b) The CCD camera is mounted on TAIWAN *i*TS-1.

Fig. 5.13 TAIWAN *i*TS-1. (a) TAIWAN *i*TS-1 is running on the freeway under 110 km/hr. (b) The CCD camera is mounted on TAIWAN *i*TS-1.

5.2 Discussion

Since the GOLD system developed in the ARGO project is famous in the region of smart vehicles [2, 8], it will be compared with the algorithm proposed in this thesis. The GOLD system uses two left and right cameras to detect the obstacle and the left is used to detect the lane markings. The GOLD system removes the perspective effect by transforming both road images into the top views, and the detection is performed in the world coordinates; the obstacles are determined if two triangles in the difference image between remapped views can be joined and the lane markings are detected based on the constant lane width, which may fail when the assumption of the flat road is illegal. The comparison between the GOLD system and the algorithms proposed in this thesis is presented as follows:

(1) Based on the assumption of the flat roads, the GOLD system may not be suitable for all real situations. Fig. 5.14 presented in the literature [8] demonstrates that the GOLD system fails in the case of a non-flat road where the lane width diverges. However, the calibrations on the road inclination and the lane width are considered in (4-22) in this thesis in order to provide the precise information for the next frame.

(a) The road is not flat.　　(b) The remapped image.
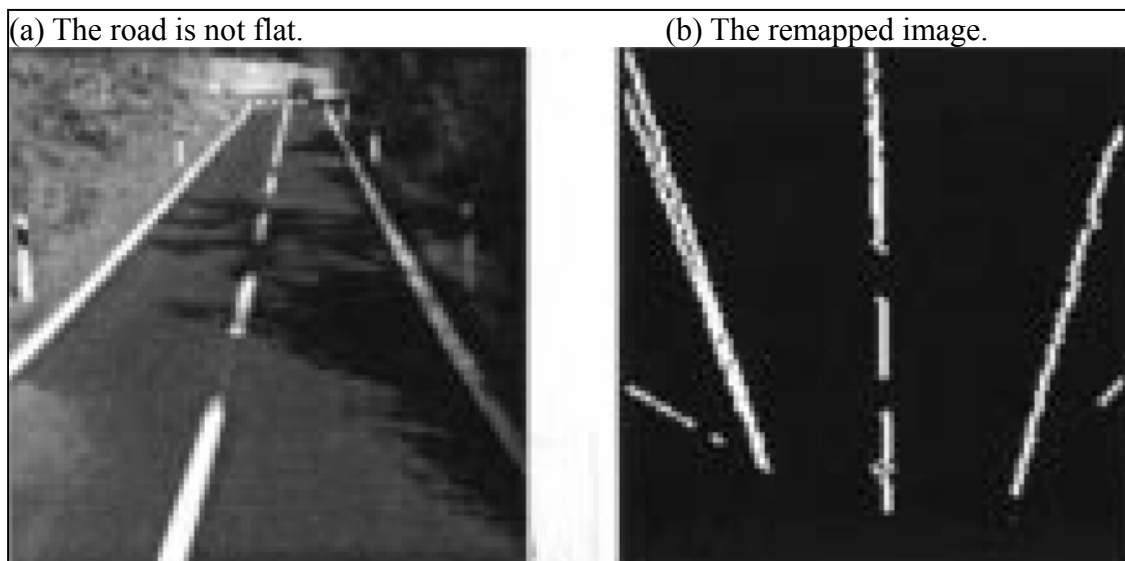
Fig. 5.14 The GOLD system fails in the case of a non-flat road [8]. (a) the road is not flat. (b) the remapped image.

(2)   A smaller image range can be used in the GOLD system after transforming the images into the top view, as illustrated in Fig. 5.15. However, the system proposed in this thesis keeps all image information because the detection is performed in the image domain.

(a)   The original road image.

Loss region

Remapped region
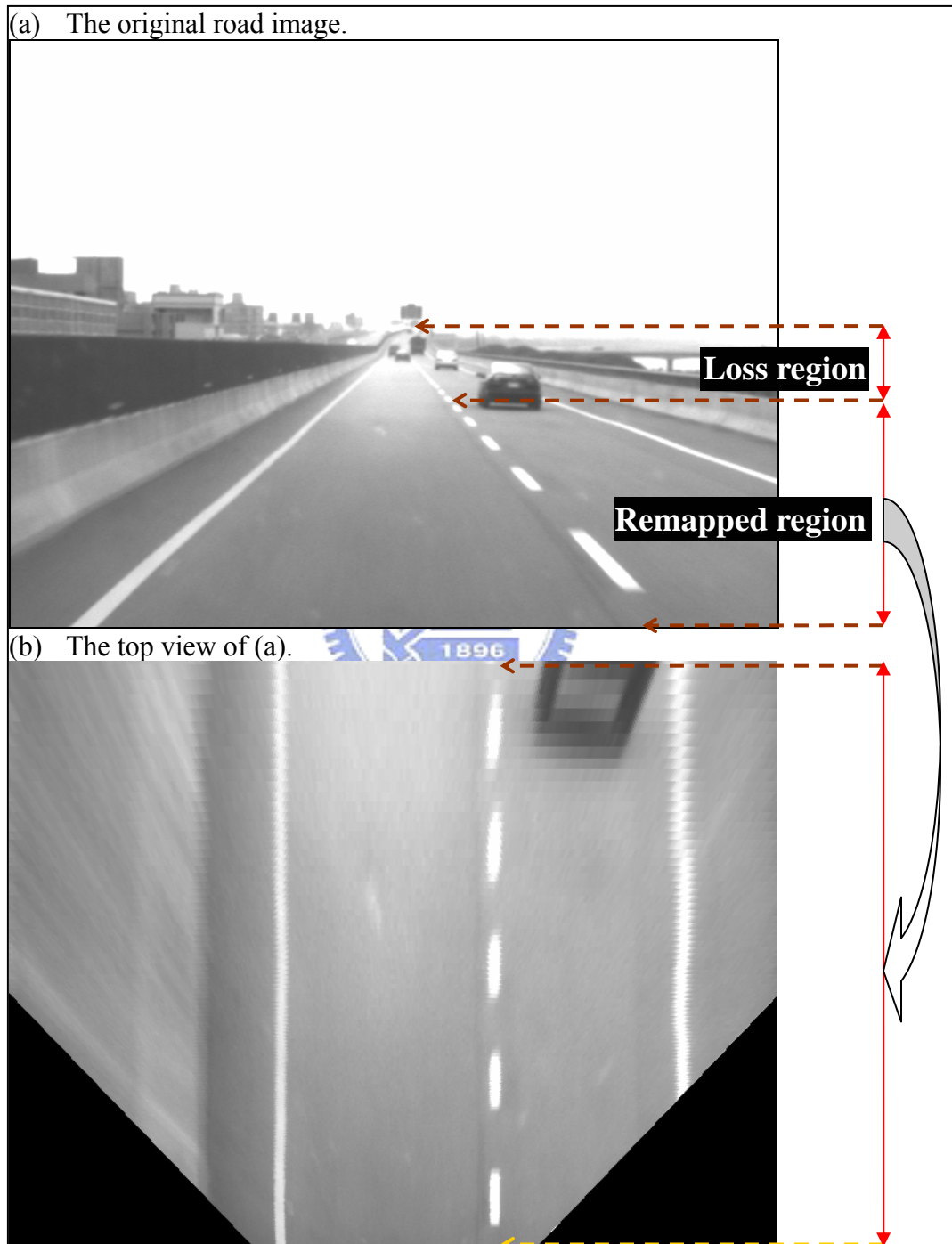
(b)   The top view of (a).

Fig. 5.15 The useful region in the world domain is smaller than that in the image domain. (a) The original road image. (b) The top view of (a).

(3)   Both algorithms rely on the constant lane width. The algorithm proposed in this thesis is performed in the original image, transforms the constant lane width from the world domain into the image domain by (2-32), and takes only one operation per image row. However, the GOLD system works in the world coordinates by mapping the whole original image into the top view pixel by pixel, so that it takes more complex operations and more time than that proposed in this thesis.

(4)   For the obstacle detection system, the roadside obstacles can be determined in this thesis. However, such a function is not considered in the GOLD system.

(5)   The lane detection algorithm proposed in this thesis, based on a parametric lane model, is more robust against the interferences such as shadows, textures, or other vehicles.

(6)   The proposed algorithm of lane detection can predict the lane tendency, and thus can determine the detection ROIs so as to narrow the searching ranges. Hence, the time is saved. However, the GOLD system searches the whole remapped image in the world domain, and therefore it has a higher computational load.

(7)   No matter what kind of algorithms of lane detection, the lane geometry is usually fitted into a curve since the goal of lane detection is to supply the drivers with the lane information. Based on the geometric lane model, the proposed algorithm involves the prediction of the lane tendency, and no more complex procedure must be taken to obtain the lane tendency while finishing the detection. Besides, even if the detection has not been finished, the approximate lane tendency can be acquired from the prediction procedure. However, extra operations must be taken in the GOLD system in order to obtain the lane tendency.

Chapter 6 Conclusions

Both algorithms of generic obstacle and lane detection based on the techniques of computer vision are proposed in this thesis. Two monochromatic CCD cameras are mounted top and bottom respectively on the vehicle in order to detect the generic obstacles, and the top camera is also used to detect the lane. The quasi-horizontal boundaries in the top road image are detected in order, and each detected boundary could belong to either the ground or the obstacle. The criterion to distinguish between them is to predict the corresponding ground and obstacle boundaries in the bottom image by the stereo vision. The detected boundary in the top image belongs to the obstacle if it is more related to the obstacle boundary predicted in the bottom image than to the ground boundary predicted in the bottom image.

After that, the obstacles in the road image can be determined, and the remainder image part without obstacles is used to detect the lane, so that the result of lane detection is not affected by the obstacles. On the other hand, the lane detection algorithm proposed in this thesis can be performed alone using a single monochromatic camera. Based on the geometric lane model, it can generate a robust result. Besides, the detection region of interest can be estimated to narrow the searching area. Eventually, the 3-D lane geometry is reconstructed to update the road inclination and lane width. Therefore the proposed algorithm is available in the case of non-flat roads.

The lane detection system has been verified in some environments such as the expressway or freeway, the straight or crooked roads, shadows or sunlight conditions, the night or rainy cases, and the roads interfered with the text or vehicles. The average time of lane detection is less than 1 ms per frame of size 644×493 on the PC platform of 2.6-GHz CPU and 512-MB RAM. Besides, the lane detection system has been integrated with the controller of the steering wheel on the automatic car, TAIWAN iTS-1. TAIWAN iTS-1 is the

CHAPTER 6 CONCLUSIONS

first smart car in Taiwan capable of hand-free driving on the expressway and freeway with velocities of 90 km/hr and 110 km/hr respectively, which demonstrates the practicability and robustness of the proposed lane detection system.

Another fundamental function of the smart vehicle is the leading vehicle tracking. In this case, the accurate distance and orientation of the leading vehicle must be determined. Sometimes there is a desire to follow the leading vehicle on the road. A practical application is to stop and to go with the leading vehicle in a traffic jam. Therefore the system of the leading vehicle tracking is an important topic in the future.

Reference:

[1] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi, and M. Porta, "Artificial Vision in Road Vehicles," *in Proc. IEEE*, vol. 90, July 2002.

[2] A. Broggi, M. Bertozzi, A. Fascioli, C. Guarino Lo Bianco, and A. Piazzi, "Visual Perception of Obstacles and Vehicles for Platooning," *IEEE Trans. Intelligent Transportation Systems*, vol. 1, pp. 164-176, Sept. 2000.

[3] A. Giachetti, M. Campani, and V. Torre, "The Use of Optical Flow for Road Navigation," *IEEE Trans. Robotics and Automation*, vol. 14, pp. 34-48, Feb. 1998.

[4] R. Alix, F. Le Coat, and D. Aubert, "Flat World Homography for Non-Flat World On-Road Obstacle Detection," *in Proc. IEEE, on Intelligent Vehicles Symposium*, pp. 310-315, 9-11 June 2003.

[5] R. Okada, Y. Taniguchi, K. Furukawa, and K. Onoguchi, "Obstacle Detection Using Projective Invariant and Vanishing Lines," *in Proc. IEEE, on Computer Vision*, vol. 1, pp. 330-337, 2003.

[6] R. Okada and K. Onoguchi, "Obstacle Detection Based on Motion Constraint of Virtual Planes," *in Proc. IEEE/RSJ, on Intelligent Robots and System*, vol. 1, pp. 61-66, 30 Sept.-5 Oct. 2002.

[7] C. Demonceaux, A. Potelle, and D. Kachi-Akkouche, "Obstacle Detection in a Road Scene Based on Motion Analysis," *IEEE Trans. Vehicular Technology*, vol. 53, pp. 1649-1656, Nov. 2004.

[8] M. Bertozzi and A. Broggi, "GOLD: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection," *IEEE Trans. Image Processing*, vol. 7, pp. 62-81, Jan. 1998.

[9] R. Labayrade, D. Aubert, and J.-P. Tarel, "Real Time Obstacle Detection in Stereovision on Non Flat Road Geometry Through V-Disparity Representation," *in Proc. IEEE, on Intelligent Vehicles Symposium*, vol. 2, pp. 646-651, Versailles, 17-21 June 2002.

REFERENCE

[10] R. Labayrade and D. Aubert, "A Single Framework for Vehicle Roll, Pitch, Yaw Estimation and Obstacles Detection by Stereovision," *in Proc. IEEE, on Intelligent Vehicles Symposium*, pp. 31-36, 9-11 June 2003.

[11] Gang Yi Jiang, Tae Young Choi, Suk Kyo Hong, Jae Wook Bae, and Byung Suk Song, "Lane and Obstacle Detection Based on Fast Inverse Perspective Mapping Algorithm," *in IEEE Conf. on Systems, Man, and Cybernetics*, vol. 4, pp. 2969-2974, 8-11 Oct. 2000.

[12] K. Kluge and S. Lakshmanan, "A Deformable-Template Approach to Lane Detection," *in Proc. IEEE, on Intelligent Vehicles '95 Symposium*, pp. 54-59, Detroit, 25-26 Sept. 1995.

[13] C. Kreucher and S. Lakshmanan, "LANA: A Lane Extraction Algorithm that Uses Frequency Domain Features," *IEEE Trans. Robotics and Automation*, vol. 15, pp. 343-350, April 1999.

[14] A. Takahashi, Y. Ninomiya, M. Ohta, and K. Tange, "A Robust Lane Detection Using Real-time Voting Processor," *in Proc. IEEE/IEEJ/JSAI, on Intelligent Transportation Systems*, pp. 577-580, 5-8 Oct, 1999.

[15] Yue Wang, Eam Khwang Teoh, and Dinggang Shen, "Lane Detection Using B-Snake," *in Proc. IEEE, on Information Intelligence and Systems*, pp. 438-443, 31 Oct.-3 Nov. 1999.

[16] J. Goldbeck and B. Huertgen, "Lane Detection and Tracking by Video Sensors," *in Proc. IEEE/IEEJ/JSAI, on Intelligent Transportation Systems*, pp. 74-79, 5-8 Oct. 1999.

[17] J.P. Gonzalez and U. Ozguner, "Lane Detection Using Histogram-based Segmentation and Decision Trees," *in Proc. IEEE, on Intelligent Transportation Systems*, pp. 346-351, 1-3 Oct. 2000.

[18] R. Chapuis, R. Aufrere, and F. Chausse, "Accurate Road Following and Reconstruction by Computer Vision," *IEEE Trans. Intelligent Transportation Systems*, vol. 3, pp. 261-270, Dec. 2002.

[19] Young Uk Yim and Se-Young Oh, "Three-Feature Based Automatic Lane Detection Algorithm (TFALDA) for Autonomous Driving," *IEEE Trans. Intelligent Transportation Systems*, vol. 4, pp. 219-225, Dec. 2003.

REFERENCE

[20] R. Klette, K. Schluns, and A. Koschan, <u>Computer Vision: Three-Dimensional Data from Images</u>, Springer, New York, 1998.

[21] R. C. Gonzalez and R. E. Woods, <u>Digital Image Processing</u>, Prentice Hall, Upper Saddle River, New Jersey, 2002.

[22] S. Theodoridis and K. Koutroumbas, <u>Pattern Recognition</u>, Academic Press, San Diego, California, USA, 1999.

VITA

姓名：賴則全（ Tze-Chiuan Lai ）

性別：男（ Male ）

生日：民國七十年二月十二日 （ 02.12.1981 ）

籍貫：高雄市（ Kaoshiung City ）

學歷：

09/2003~07/2005　　交通大學電機與控制工程研究所碩士班
09/1999~07/2003　　成功大學工程科學系
09/1996~07/1999　　高雄中學

榮譽紀錄：

(1) 中華民國斐陶斐榮譽會員(交通大學分會推薦)
(2) 交通大學電機與控制工程研究所碩士班第一名
(3) 交通大學書卷獎(碩一上下學期)
(4) ９４年度「朱順一合勤」獎學金
(5) 第一屆機動車輛創新設計競賽智慧電子化機能創新設計組金質獎
　　—作品名稱"智慧型車輛自動駕駛系統"
(6) 第一屆機動車輛創新設計競賽智慧電子化機能創新設計組銀質獎
　　—作品名稱"具有肇事現場重建功能的影音行車紀錄器"
(7) ９２學年度教育部大專校院通訊競賽研究所組入圍獎
　　—作品名稱"結合小波轉換與離散餘弦轉換技術之高速視訊編碼技術與其在智慧
　　型影音監控系統的實現"
(8) 中華民國斐陶斐榮譽會員(成功大學分會推薦)
(9) 成功大學工程科學系第一名畢業
(10)成功大學書卷獎(大一至大三每學年均獲得)
(11)成功大學工科系基金會獎學金(大一至大三)
(12)成功大學８８學年度全校微積分學科實力測驗甲等(Top 2%)

VITA

Publication List：

Conference

[1] Bing-Fei Wu, Chao-Jung Chen, Chung-Cheng Chiu, and **_Tze-Chiuan Lai_**, "A Real-Time Robust Lane Detection Approach for Autonomous Vehicle Environment," *Proceedings of the Sixth IASTED International Conference on Signal and Image Processing (SIP 2004)*, pp. 518-523, Hawaii, USA, August 23-25, 2004.

[2] **_Tze-Chiuan Lai_** and Gwo-Bin Lee, "A Novel Micro Mixer Using Magnetic Activation," posted at the *First International Meeting on Microsensors and Microsystems (IMμ2)*, Tainan, Taiwan, January 12-14, 2003.

[3] *賴則全*, 李國賓, "新式磁致動式微混合器之研究," *中華民國力學學會第二十六屆全國力學會議*, 雲林, 台灣, December 20-21, 2002.

VITA

VITA

VITA

賴則全同學
獎獲94年度
ZyXEL 合勤
獎學金
朱順一　敬贈
中華民國九十五年六月

保證純紋銀製品
We guarantee all are made of pure silver.

陳彥霖
陳昭榮
洪堯俊
賴則全
君作品：
結合小波轉換與離
散餘弦轉換技術之
高速視訊編碼技術
與其在智慧型影音
監控系統的實現

參加本部九十二學年度大專校院通訊科技
專題製作競賽暨亞太地區名校邀請觀摩賽
研究所組競賽
經評定晉級複賽並參與展示
特予證明

證明書

台顧字第0930056665號

部長 杜正勝

中華民國九十二年五月二十九日

NATIONAL CHENG KUNG UNIVERSITY
CERTIFICATE OF MERIT

This is to certify that _Lai Tze-Chiuan_
a student of Department of _Engineering Science_
College of _Engineering_ of this University
was granted the _excellent_ Achievement
Award for his / her participation in the Calculus
Contest within the top _2_ percent of all the
students participated in the academic year 2000.

Cheng-I Weng
President

獎狀

本校工學院工科系學生 賴則全
參加本校舉辦微積分學科實力測驗其成績名列
前千分之二十 經評審榮獲甲等 特頒給
獎狀以資鼓勵
此狀

校長 翁政義

VITA

中華民國九十三年三月三十日

張俊彥

國立交通大學校長

獎狀

（92）學（一）字第二〇三號

電機與控制工程學系研究生

賴則全在本校九十二

學年度第一學期學業成

績優異，殊堪嘉許。特

頒此狀，用示獎勉。

中華民國九十三年十月十五日

張俊彥

國立交通大學校長

獎狀

（92）學（一）字第二三五號

電機與控制工程學系研究生

賴則全在本校九十二

學年度第二學期學業成

績優異，殊堪嘉許。特

頒此狀，用示獎勉。

中華民國八十九年十一月十一日

翁鴻山

國立成功大學

代理校長

獎狀

本校工學院 工科系 一年級

學生 賴則全 於八十八學

年度在校品學兼優依本校優良

學生書卷獎獎勵辦法規定合

予發給獎狀以資獎勵

此狀

中華民國九十年十一月十一日

高強

國立成功大學

校長

獎狀

本校工學院 工科系 二年級

學生 賴則全 於八十九學年

度在校品學兼優依本校優良

學生書卷獎獎勵辦法規定合

予發給獎狀以資獎勵

此狀

中華民國九十一年十一月十一日

高強

國立成功大學

校長

獎狀

本校工學院 工科系 三年級

學生賴則全於九十學年度在

校品學兼優依本校優良學生

書卷獎獎勵辦法規定合予發

給獎狀以資獎勵

此狀

中華民國八十九年十一月十一日

系主任 楊瑞珍

國立成功大學 工程科學系 學

生 賴則全 八十九學年度在校表

現傑出符合本系獎勵優秀學生之

規定特頒獎學金及獎狀以資鼓勵

此狀

獎狀

（89）

成大工科字第〇〇五號

中華民國九十年十一月十一日

系主任 楊瑞珍

國立成功大學 工程科學系 學

生 賴則全 九十學年度在校表現

傑出符合本系獎勵優秀學生之規

定特頒獎學金及獎狀以資鼓勵

此狀

獎狀

（90）

成大工科字第〇〇〇號

中華民國九十一年十一月十一日

系主任 楊瑞珍

國立成功大學 工程科學系

學生 賴則全 於民國九十一年

在校表現傑出符合本系獎勵優秀

學生之規定特頒獎學金及獎狀以

資鼓勵

此狀

獎狀

（91）

成大工科字第〇〇四號