CMOS Image Sensors and Algorithms for Traffic Detection Applications

A Dissertation

Presented in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

with a

Major in Electrical Engineering

in the

College of Graduate Studies

University of Idaho

by

Bingxing Wu

Major Professor: Suat U. Ay, Ph.D.

Committee Members: Mohsen Guizani, Ph.D.; Sameh Sorour, Ph.D.;

Ahmed Abdel-Rahim, Ph.D.

Department Administrator: Mohsen Guizani, Ph.D.

May 2018

AUTHORIZATION TO SUBMIT DISSERTATION

This dissertation of Bingxing Wu, submitted for the degree of Doctor of Philosophy with a major in Electrical Engineering and titled "CMOS Image Sensors and Algorithms for Traffic Detection Applications," has been reviewed in final form. Permission, as indicated by signatures and dates given below, is now granted to submit final copies to the College of Graduate Studies for approval.

Major Professor: _____ Date _____
                          Suat U. Ay, Ph.D.

Committee
Members: _____ Date _____
                          Mohsen Guizani, Ph.D.

                 _____ Date _____
                          Sameh Sorour, Ph.D.

                 _____ Date _____
                          Ahmed Abdel-Rahim, Ph.D.

Department
Administrator: _____ Date _____
                          Mohsen Guizani, Ph.D.

ABSTRACT

Extracting information about vehicles or pedestrians detected by vision-based systems is vital for smart surveillance applications and traffic monitoring systems. These systems must have the ability to extract real-time parameters form the observed objects, such as number, location, speed, size, acceleration, temperature, signs, occupants, etc., necessary for next-generation intelligent transportation systems.

Both image sensors and algorithms used in the vision-based systems require low data rate (leads to smaller communication bandwidth), low complexity (leads to simple and smalls implementation and lower cost), and low power for fast, smart and efficient operation. These requirements could be addressed by the proposed trapezoid (TZOID) multi-resolution complementary metal oxide semiconductor (CMOS) active pixel sensor (APS) imager that has customized pixels in the pixel array allowing efficient pixel-resolution mapping (PR-M) method.

The focus of this study is to develop techniques for designing multi-resolution CMOS APS imagers and to simplify the feature extraction algorithms with straightforward camera calibration methods while achieving low power, low complexity and low data rate for next generation traffic detection systems. Fundamental problems associated with designing pixel array of a multi-resolution CMOS image sensor, PR-M mapping method, and feature extraction algorithms with straightforward camera calibration method are addressed in this research.

The technical inquiry is divided into four research objectives. First objective is the investigation of video-based traffic detection and general camera calibration/mapping methods. The main outcome of this objective is to illustrate the drawbacks of current techniques and to develop multi-resolution CMOS imager sensors and camera systems. The second objective is to model the relationship between pixel's size and pixel's spatial resolution in real-world for design and development of trapezoid pixel arrays that could generate low data rate at significantly reduced power consumption. The third objective is to develop camera calibration and mapping methods without specific calibration tools or iterative procedures. The fourth objective is to develop intelligent feature extraction

algorithms, specifically for vehicle and pedestrian detection with low complexity and computational requirements.

The intellectual merits include: (1) providing a model for pixel's physical size and related real-world spatial resolution, (2) understanding the challenges in design of multi-resolution pixel array for traffic detection, (3) providing a general framework on multi-resolution pixel array design for traffic applications with smaller frame memory, (4) understanding the challenges in camera calibration and feature extraction from two-dimensional (2D) images, (5) developing efficient and accurate features extraction algorithm with low complexity camera calibration procedures for vehicle and pedestrian detection.

A TZOID multi-resolution CMOS APS image sensor that extracts real-time parameters from the observed objects without requiring any specific patterns or manmade markings for traffic applications is reported in this work. The TZOID imager sacrifices detection accuracy tolerable for alleviating superfluous data to achieve low communication bandwidth and low computation power. A new P-RM method that deduces the relationship between pixel's size and its spatial resolution was used for both deriving pixel design equations for the TZOID image sensor and for extracting location and speed of the detected objects. Additionally, P-RM method was also used for height estimation and three-dimensional (3D) reconstruction of scenes captured by the imagers.

After computer simulations, both controlled laboratory and actual world experiments were performed with trapezoid and standard (rectangular) CMOS image sensors. The tests on 3D mensuration using proposed P-RM method showed overall accuracy of better than 98% in laboratory environment and better than 96% accuracy in real world pedestrian height estimation cases. It is also shown that the proposed P-RM method provides a general algorithm for 3D information extraction and reconstruction of 2D objects.

The TZOID multi-resolution CMOS APS image sensor integrated circuit (IC) was fabricated in a 0.18µm CMOS process. It was tested with P-RM method and its performance parameters were compared with an off-the-shelf standard CMOS image sensor IC. Tests showed that TZOID generates 98% less data than that of standard CMOS image sensors while trading off 2% detection accuracy.

ACKNOWLEDGMENTS

DEDICATION

This dissertation is lovingly dedicated to my parents, Peifu Wu, Haili Sun, my wife Wencan li, and my grandparents.

Their support, encouragement, and constant love have been the main motivation throughout my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1   - INTRODUCTION

Controlling traffic signals are mostly depend on monitoring vehicles on signalized intersection approaches. Current detection methods used in signalized intersection approaches are based on binary detection of objects at specific points on their approach, such as by using loop detectors. However, binary detection does not provide dynamic properties of approaching vehicles accurately, such as their location, speed, acceleration/deceleration, and expected time of arrival at the stop bar location, etc. If the traffic controller has access to these information by using detectors that have ability to monitor continuously and accurately, it would optimize traffic flow achieving better energy efficiency by minimizing delay and elimination of unnecessary stops.

Continuous monitoring of traffic can be achieved by using three types of detector systems; infrared, radar, and video/image sensor [1]. Infrared detection devices utilize both active and passive methods. The passive infrared detectors cannot extract the vehicles speed, while the active ones may not be reliable and require regular maintenance [1], [2]. Radar detectors could not determine the size or profiles of vehicles, and usually require specific mounting locations and conditions [3]. Besides, both infrared and radar detectors are complicated systems that might be prohibitively expensive to be utilized in intersections. Video based system composes of a camera with charge-coupled devices (CCD) or complementary metal-oxide semiconductor (CMOS) imager sensors mounted on the traffic light poles to generate real-time traffic parameters such as, vehicle counts, vehicle speed, direction, and path.

Incredible advances made in semiconductor industry in the last few decades were driven by shrinking feature sizes in CMOS technology. In every CMOS process generation, as transistors became smaller, faster and cheaper with continuous technology scaling, more transistors were integrated on microchips enabling denser and faster integrated circuits (ICs). In parallel, CMOS image sensors advanced with the features of camera-on-a-chip, including high-speed, high-dynamic range, low-power consumption, and low manufacturing cost [4]. CMOS image sensors has received a growing interest and become the dominant image sensor technology within the last two decades.

Unlike the CCD imagers, CMOS image sensors do not require dedicated manufacturing processes. They are built in the same mainstream CMOS electronic manufacturing processes used for manufacturing analog, digital, and mixed signal ICs, such as digital logic ICs, microprocessors, memory, and application specific integrated circuits (ASICs). Therefore, CMOS image sensors can be integrated with peripheral digital and analog electronics on the same chip. This allowed integration of complete camera systems on a single IC, reducing the component, packaging, and system costs, [5].

CMOS image sensor based camera systems are being widely used in security, surveillance, and traffic detection applications due to ease of installation and maintenance [6], [7]. These systems are often required to have the ability to extract real-time parameters from the observed objects, such as number, location, size, speed, acceleration, etc. For instance, it is important to determine the vehicle size, vehicle location and vehicle speed in real-time for modern intelligent transportation systems [8].

Typically, the camera is placed high above the ground (higher than 12 meters) having a bird's eye view of the road in vision-based traffic monitoring systems [9]. These systems usually need high-speed cameras and digital signal processors (DSPs) or standalone computer nodes with complex camera calibration algorithm [6], [10]–[12]. Camera calibration algorithm is required for traffic monitoring to determine the relationship between two-dimensional (2D) captured image by the camera and three-dimensional (3D) object exist in the field of view (FOV) of the camera, which is necessary for extracting vehicle's location and speed [7], [13], [14]. If the systems and algorithms used in real-time traffic detection have none or few feature extraction capabilities, they will produce large amount of unimportant data, which will increase both communication bandwidth and system power consumption. As a result, more powerful computation capability is required that necessitates high performance platforms. Consequently, to meet low-power, low-cost and light-weight requirements of next generation traffic monitoring systems, a custom designed CMOS image sensor with a straightforward calibration method to extract object's features is required.

Several methods have been utilized for image sensors to eliminate irrelevant image data and alleviate the burden of image processing and calculation power in traffic monitoring platforms. The methods include providing direct frame difference output, utilizing multi-resolution image sensors, and including on-chip clustering algorithms, [15]–[18]. Although

these custom image sensors and image pre-processing methods used by them reduces the generated image data, they still require camera calibration algorithms for traffic monitoring applications.

Various camera calibration methods have been published for photogrammetry and mensuration for traffic applications [7], [10], [19]–[23].These could be classified into two categories; conventional and self-calibration methods.

Conventional calibration methods have been widely investigated in the past. It usually requires a specific object with marked geometric patterns placed in 3D world [19]–[23], or uses road-lane markings [7], [13], [14], [24] as calibration targets. However, all these conventional methods require a specific pattern in 3D world with a rectangular shaped pixel-array camera having redundant data output. The size of these patterns must be known, though the patterns could be road-lane markings.

The methods based on self-calibration extract intrinsic and extrinsic parameters of the camera system without requiring any calibration rig or patterns in 3D world. It requires multiple images of a dynamic scene taken at different locations at the same time or multiple images of a static scene taken at different times [25]–[30]. Main disadvantages of the self-calibration methods are that they are nonlinear, highly sensitive to noises, and many parameters are determined using iterative procedures.

These techniques are shown to be reliable and time-tested for deriving intrinsic and extrinsic parameters of the camera setup. However, they have the disadvantages of complex algorithm with iterative procedures requiring significant computational resources that limit their application scenarios. They are also inefficient for real-time applications and require high communication bandwidth.

To avoid the problems, a custom designed multi-resolution CMOS image sensor with the capability of extracting "only" the dynamic properties of incoming traffic and the facile features of vehicles is desirable. In addition, a new straightforward calibration and mapping methods are required. The new methods would work with both standard rectangular shape pixel array and customized shape pixel array image sensors to generate the relationship between 2D image and 3D world which is free of the disadvantages of the existing methods.

1.1   Motivation and Goals

The focus of this research is to develop a multi-resolution CMOS active pixel sensor (APS) imager with a straightforward calibration method and feature extraction capability for traffic detection application. A custom designed multi-resolution CMOS APS imager can have a non-rectangular shaped pixel array with variety of pixel sizes. Each pixel's size is determined and designed according to desired spatial resolution to achieve less unimportant data with enough detection accuracy. Research efforts are focused on two aspects to realize this special multi-resolution image sensor: (1) designing a trapezoid pixel array by investigating and developing the model between pixel's physical size in pixel array and pixel's spatial resolution in real world; (2) generating a straightforward method on calibration and feature extraction with enough detection accuracy by pixel-resolution mapping (P-RM) method without using specific real world patterns or iteration procedures.

There are several ways to reduce the irrelevant data, and alleviate the burden of image processing in motion detection such as direct frame difference output, utilizing multi-resolution image sensors, and clustering algorithm [15]–[18]. However, these methods do not give the relationship between physical size of the pixel in the array and related pixel's spatial resolution in real world. On the other hand, these image sensors are essentially general rectangular shape imagers which have no ability to adjust balance between detection accuracy and amount of data generated by pixels. Additionally, camera calibration procedures have to be applied on the imagers before detection and mensuration start [25], [27]–[29].

Thus, *the motivation* of this work is to develop a custom CMOS image sensor with specifically designed pixel array for traffic detection applications with innovative mapping algorithms and intelligent feature extraction methods. To achieve these, the research is divided into *five research goals*. *The first* goal is the investigation of fundamental principles, limits, and problems associated with vision-based traffic detection resulting in a design strategy including both sensor design and algorithm design. *The second* goal is the investigation and development of camera model to achieve a specific model (pixel-resolution model) for pixel's physical size and related pixel's spatial resolution. *The third* goal is to design and fabricate a TZOID CMOS imager based on pixel-resolution model for traffic applications. *The fourth* goal is to develop feature extraction algorithms for designed image

sensor with straightforward camera calibration algorithms based on pixel-resolution model. ***The fifth*** goal is to develop a TZOID CMOS image sensor based vision system which is suitable for traffic detection with high accuracy and low data rate.

1.2 Applications

The TZOID is designed for traffic detection, specifically for extracting "only" the dynamic properties of incoming vehicles or moving persons. Thus, the primary commercial markets for TZOD imager are traffic detection and surveillance monitoring applications including vehicle location detection, vehicle speed detection, vehicle size mensuration, pedestrian moving speed detection, and pedestrian height estimation. The algorithms including P-RM method, feature extraction method and straightforward calibration method in this research work could be used with any other 2D imagers directly. Besides, these 2D imagers could utilize the algorithms developed in this research for 3D mensuration without iterative procedures and by using single image.

1.3 Contributions

This research resulted in several advancements in the field of CMOS image sensors with multi-resolution pixel array, and vision-based traffic detection including image mensuration and camera calibration.

The initial phase of this research work explored sensors used in traffic applications including related advantages and disadvantages that motivated the first goal of this research. This part of the research resulted in the investigation of fundamental principles, limits, and problems associated with vision-based traffic detection enlightening a whole research map including sensor design and algorithm design.

The TZOID CMOS image sensor using trapezoid pixel array with varying sizes of pixels in rows and columns was developed and manufactured in 0.18μm CMOS process and published in 2016, [31]. The design model for pixel's physical size and related pixel's spatial resolution in real world was developed and implemented. It was demonstrated that the pixel (array) design model and pixel-resolution model used in TZOID imager for achieving different desired distance resolutions played an important role in multi-resolution image

sensor design. These models could be used for building next generation image sensors detecting long distance with low data rates, faster speed, and low power consumption [32].

The second contribution of this study is achieving P-RM method enlightened by pixel-resolution model. A unique 2D image to 3D real world mapping method that can recover dimension information from 2D image for object's location, size, and speed detection with high detection accuracy was developed and integrated into TZOID based camera system [32].

The third contribution of this study is the development of a simplified camera calibration method using detected object, such as vehicle and pedestrian, as calibration target instead of using specific patterns. The method could be implemented on both regular (rectangular) imager and on proposed custom TZOID imager with high accuracy (better than 95%) and less computation power, which provides fast and flexible operations.

The fourth contribution of the research is the realization of feature extraction algorithms including pedestrian height estimation algorithm based on P-RM method. The algorithms work successfully with both TZOID and regular commercial imagers providing high accuracy (better than 96%) for traffic detection. The pedestrian height estimation algorithm works without iterative procedures, that also able to reconstruct 3D scenes with 2D captures images.

All these unique solutions and approaches were integrated and tested in controlled laboratory and real-world environments. Design details and measurement results are presented in subsequent chapters.

The overall outcome of this research is the design and implementation of the world's first trapezoid pixel array multi-resolution CMOS image sensor with flexible camera calibration technique and accurate feature extraction algorithms based on straightforward P-RM method with pixel-resolution model.

1.4   *Organization of the Thesis*

Chapter 1 introduces the research motivation, goals, and applications. It also summarizes the research contributions and the organization of this thesis.

Chapter 2 gives general background on vision-based traffic detection. First, general information on traffic detection by different types of sensors is provided. Second, a brief

introduction on vision system mounting and viewing considerations is presented. Third, typical vision detection algorithms are illustrated.

Chapter 3 gives general background on CMOS active pixel sensor (APS) imagers. First, general information on CMOS image sensors are provided. Second, a brief historical background on CMOS image sensors is presented followed by a comparison of CMOS APS and CCD technologies and technical trends. Third, CMOS image sensor architectures, their functional sub-blocks and design requirements are discussed.

Chapter 4 discusses camera calibration methods. First, the pin-hole camera model with basic terminologies is introduced. Second, current camera calibration techniques are presented in two categories; conventional camera calibration method and self-calibration method, respectively.

The remaining chapters discuss the primary contributions and measurement result of this work including; 1) pixel-resolution model with modified camera model and derived equations, 2) pixel-resolution mapping method including calibration method and feature extraction algorithms, 3) design and development of TZOID CMOS image sensor, 4) experiments of TZOID imager and related algorithms.

P-RM method is presented in Chapter 5. First, modified camera model is illustrated. Second, the relationship (pixel-resolution model, P-RM) between pixel's physical size and pixel's actual spatial resolution in real world is derived. Third, feature extraction algorithms including location detection, speed detection, and dimension detection based on P-RM method are explained. Typical image processing algorithms including image enhancement, foreground/background segmentation and edge detection are reviewed. Finally, flexible camera calibration method without specific pattern is discussed.

Chapter 6 presents design details of TZOID image sensor. First, the background of TZOID imager is presented. Second, pixel design equations based on P-RM model and details of TZOID pixel array development are discussed. Third, the image sensor architecture and details of analog and digital building blocks are illustrated. Finally, the actual TZOID chip with a custom TZOID imager-based camera system is shown.

Chapter 7 reports experimental results of this research work. First, experiments result on camera calibration and feature extraction algorithms by using commercial camera are presented including both vehicle detection and pedestrian detection. Second, testing results on

TZOID imager integrated with feature extraction algorithms is demonstrated. Third, the performance and efficiency of the TZOID is illustrated after comparing with a standard off-the-shelf image sensor.

Chapter 8 summarizes major accomplishments achieved in this research and presents ideas for future research.

CHAPTER 2 - VISION BASED TRAFFIC DETECTION

By the late 1980s, video-based detection systems were marketed in the U.S. and elsewhere, generating sufficient interest to warrant research to determine their viability as an alternative to inductive loop detection systems. In 1990, the California Polytechnic State University began testing 10 commercially available video image processing systems in the United States. Evaluation results indicated that most systems generated vehicle count and speed errors of less than 20 percent over a mix of low, moderate, and high traffic densities under ideal conditions [33] .

Video cameras transmit closed circuit television (CCTV) imagery to a human operator for interpretation. More advanced cameras of today use video image processing to automatically analyze the scene of interest and extract information for traffic monitoring using high-speed digital cameras. Typically, the imagery is digitized in hardware that is hosted in personal computer (PC) architecture. The PC also accommodates application-specific software used to calculate the desired traffic parameters. Video image processors can replace several in-ground inductive loop detectors by a single above-the-road camera and signal processing that provide wide area detection of vehicles and promise lower maintenance costs. Video detection has the potential to classify vehicles and report vehicle presence, volume, occupancy, and speed for each class and for each lane observed, [33]–[35].

2.1 Mounting and Traffic Viewing Considerations

Table 2-1 shows how processing of upstream or downstream imagery influences video detector performance. The primary advantage of upstream viewing is that incidents are not blocked by the resulting traffic queues. However, tall trucks may block the line of sight and headlights may cause blooming of the imagery. Downstream viewing offers advantages of camera concealment so that driver behavior is not altered, easier identification and tracking of vehicles using information contained in the tail lights, and better acquisition of vehicle tracks because the vehicles are closer to the camera at track initiation.

Table 2-1. Vision System Characteristics as Used in Upstream and Downstream Viewing.

Upstream Viewing	Downstream Viewing
• Headlight blooming and glare from wet pavement; • More blockage from tall trucks; • With infrared imagery, there is no difference in information obtained from headlights or tail lights when a tracking algorithm is used; • Traffic incidents are not blocked by resulting traffic queues.	• Camera concealed from drivers; • More information from tail lights available for braking indication, vehicle classification, and turning movement identification; • More information is available to a tracking algorithm from tail light viewing; • Easier to acquire vehicles that are closer to the camera.

Based on line-of-sight considerations, the detection distance at which video detectors can differentiate two closely spaced vehicles along the surface of a road is a function of camera mounting height (h), inter-vehicle distance or gap (d_{gap}), and vehicle height ($h_{vehicle}$) as shown in Figure 2-1. The maximum detection distance (d_{max}) along a roadway could be derived as 2.1 according to similar triangle theory [36].

$$d_{max} = h \times \frac{d_{gap}}{h_{vehicle}}$$

(2.1)

Other factors to be considered when installing cameras used in video detector systems include, vertical and lateral viewing angles, number of lanes observed, stability with respect to wind and vibration, and image quality.
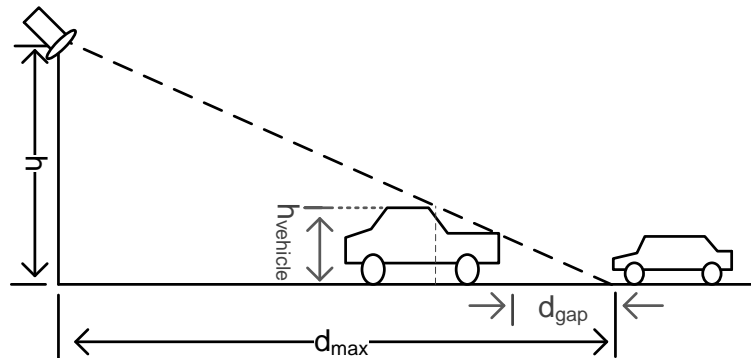
Figure 2-1. Concise camera mounting geometry with upstream viewing.

Video detectors tolerate an oblique view of the highway if the mounting height is high such as 45 to 50 feet (13.7 to 15.2 m). For lower heights, it is about 18 to 25 feet (5.5 to 7.6 m). In all cases, a mounting location centered over the area of interest may be required. The number of lanes of imagery analyzed by the video detector becomes important when the required field of view is larger than the video detector's capability. For example, if the video detector provides data from detection zones in three lanes, but five must be observed, that video detector may not be appropriate for the application. Video detector's sensitive to large camera motion may be adversely affected by strong winds.

Image quality and interpretation can be affected by cameras that have automatic iris and gain controls. In tests conducted by California Polytechnic Institute at San Luis Obispo, these systems were disabled. In other video detectors, the signal processing is tailored to take advantage of automatic light control systems. Using the same camera for automatic vehicle detection with a video detector and video surveillance with pan, tilt, and zoom features requires the camera to be repositioned for each application. If the field of view is not returned to the calibrated value for video detector operation, the performance of the video detector is adversely affected. It may be technically feasible, however, to reposition the camera at previously established video detection zones after it has been panned, tilted, or zoomed to view an incident location for verification and identification. In this case, one camera can be used for both applications. If the remote control of cameras and their return to calibrated fields of view is not feasible, then separate cameras may be required to perform automated traffic data collection and video surveillance. When two cameras are used, a lower cost camera system will generally suffice for the video detector mission as the pan, tilt, and zoom features are not required [36].

2.2   Introduction on Video Detection Algorithms

One of the first applications of digital images was in the newspaper industry, when pictures were first sent through submarine cables between London and New York. The first computers powerful enough to carry out meaningful image processing tasks appeared in the early 1960s. The birth of what we call digital image processing today can be traced to the availability of these machines and to the onset of the space program during same period. It took the combination of these two developments to bring into focus the potential of digital

image processing concepts. Work on using computer techniques for improving images from a space probe began at the Jet Propulsion Laboratory (Pasadena, CA, USA) in 1964 when pictures of the moon transmitted by *Ranger 7* were processed by a computer to correct various types of image distortion inherent in the on-board television camera, [37].

Nowadays, image processing is among the rapidly growing technologies, which includes algorithms that could enhance or manipulate images to extract useful information from the acquired images while playing an important role in two principal application areas: improvement of pictorial information for human interpretation, and processing of image data for storage, transmission, and representation for autonomous machine perception. Typical applications in machine perception are automatic character recognition, industrial machine vision for product assembly and inspection, military reconnaissance, screening of blood samples, satellite imagery for weather prediction and environmental assessment, and traffic detection [37]–[39].

Typically, three steps including, foreground segmentation, vehicle or pedestrian detection and vehicle or pedestrian tracking, are used in traffic detection application with vision-based system.

2.2.1 Foreground Segmentation

Foreground estimation and segmentation is the first stage of general visual surveillance systems. The foreground is defined as every object that is not a fixed furniture of a scene, where fixed could normally mean months or years. This definition conforms to human understanding, but it is difficult to algorithmically implement. A background model of some kind can be used to accumulate information about the scene background of a video sequence. The model is then compared to the current frame to identify differences, provided that the camera is stationary. This concept lends itself well for computer implementation but leads to problems with slow-moving traffic. Any car should be considered foreground, but stationary objects are missed due to the lack of motion [8]. Three typical foreground segmentation solutions for motion detection are presented concisely as following, and related implementations with examples are illustrated in Chapter 5.

2.2.1.1   *Frames Subtraction*

Frames subtraction methods employ a threshold technique over the inter-frame difference. The straightforward method for foreground segmentation is frame differencing. Frames subtraction methods detect moving objects region from current input image by performing a difference on two or three consecutive inter-frames. Meanwhile, a threshold criterion is provided to decide the region of the interest. The difference map is usually binarized using a predefined threshold value to obtain the motion region. The advantage of this method is easy to realize and its low computation requirements. However, the disadvantage of these methods is low efficiency, especially when there is no prior knowledge exists, [40].

The advanced techniques constrained to pixel-wise independent decision, as a step forward of inter-frame difference methods assuming intrinsically that the detection of temporal changes is equivalent to the motion detection [41]. The shortcoming of this method is that the assumption is valid when either large displacements appear, or the object projections are sufficiently textured. To overcome this drawback, temporal change detection masks and filters have been introduced [42]. The [42] presents a new variation framework for detecting and tracking multiple moving objects in image sequences. Motion detection is performed using a statistical framework for which the observed interframe difference density function is approximated using a mixture model. This model is composed of two components, namely, the static (background) and the mobile (moving objects) one. Both components have zero-mean and obey Laplacian or Gaussian law. This statistical framework is used to provide the motion detection boundaries. Additionally, the original frame is used to provide the moving object boundaries. Then, the detection and the tracking problems are addressed in a common framework that employs a geodesic active contour objective function. This function is minimized using a gradient descent method. A new approach named Hermes is proposed, which exploits aspects from the well-known front propagation algorithms and compares favorably to them. Very promising experimental results were provided using real video sequences [36]. The application of these masks improves the efficiency of the change detection algorithms, especially in the case where a prior knowledge about the size of the moving objects is available.

2.2.1.1.1   Background Subtraction

Background subtraction is a widely used approach for detecting moving objects from (static) cameras. The basic concept of background subtraction is that the moving object in the frames (images) could be detected by obtaining the difference between the current frame, *C(t),* and a reference frame, *B*, which is often called the "background image", or "background model". The flow chart of background subtraction is shown in Figure 2-2. Flow chat of background subtraction.. Typically, the background image is a reference including the scene with no moving objects, which is kept regularly updated so as to adapt to the changes caused by weather or illumination [43].
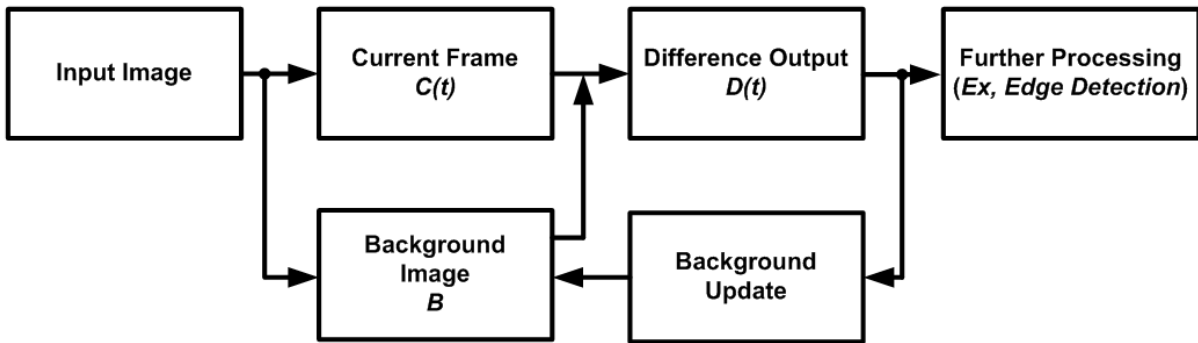
Figure 2-2. Flow chat of background subtraction.

$$D(t) = |C(t) - B|$$
(2.2)

$$M(t) = \begin{cases} 1, D(t) > T \\ 0, D(t) \leq T \end{cases}$$
(2.3)

The basic mathematical expressions for background subtraction are given in (2.2) and (2.3), where *M(t)* is foreground segmentation or moving object (*M(t) = 1*), if the value of *D(t)* is larger than (pixel) threshold, *T*. The threshold can be constant or dynamic. On the other hand, if *D(t) is* smaller than *T*, *M(t)* will be regarded as part of background (*M(t) = 0*).

Background subtraction provides most complete feature information, but the problem of these techniques is that it is extremely sensitive to dynamic scene changes due to lighting

and extraneous events [8]. Consequently, it is important to generate a proper background B for obtaining moving object correctly.

Different methods have published on background modeling including; running Gaussian average [44], [45], mixture of Gaussians [46], [47], temporal median filter [48], kernal density estimation (KDE) [49], and sequential kernel density (SKD) approximation [50].

One useful tool for building such representations is statistical modeling, where a process is modeled as a random variable in a feature space with an associated probability density function (pdf). The density function could be represented parametrically using a specified statistical distribution, which is assumed to approximate the actual distribution, with the associated parameters estimated from training data. Alternatively, nonparametric approaches could be used. These estimate the density function directly from the data without any assumptions about the underlying distribution. This avoids having to choose a model and estimating its distribution parameters. A nonparametric technique is called the kernel density estimation. It estimates the underlying density by avoid having to store the complete data, and it is quite general. In this technique, the underlying pdf is estimated with (2.4), where K is a "kernel function" (typically a Gaussian) centered at the data points in feature space, x_i, and α_i are weighting coefficients (typically uniform weights are used). This property makes these techniques quite general and applicable to many vision problems where the underlying density is not known [51].

$$\hat{f}(x) = \sum_i \alpha_i K\left(x - x_i\right) \tag{2.4}$$

In this research work, kernel density estimation method is utilized for building representations for the background image for moving vehicle and person detection followed by edge detection.

2.2.1.2 *Graph Cuts*

The foreground segmentation problem can be represented as a graph of a Markov random field (MRF). Every pixel of the images is represented by a node in the graph. The vertices between nodes and sources are set to a weight related to the data (data constraint).

Sources represent the labels for a pixel, which in this case, the foreground and the background. Vertices between nodes are used to introduce a smoothing constraint to avoid very small foreground or background regions. The graph cut completely separates the source and sink nodes and leaves the nodes connected to either source or sink node to indicate that this pixel corresponds to the respective label. The advantage of graph cuts is that the solution for this optimization problem can be found in polynomial time. A general introduction to graph cuts is given in [52].

2.2.2 **Vehicle Detection and Classification**

Vehicle detection methods can be classified into the following three categories: 1) knowledge-based; 2) motion-based; and 3) wavelet-based. Knowledge-based methods employ a prior knowledge detect the position of vehicle in ROI. Motion-based techniques detect vehicles using optical flow. Wavelet-based approaches detect vehicles wavelet neural network or wavelet based function [8].

2.2.2.1   *Knowledge-based Methods*

Knowledge-based methods employ a prior knowledge to decide whether the region of interest (ROI) is vehicle or not. The information used could be symmetry, color, shadow, vertical/horizontal edges, and texture.

Symmetry, as one of the main signatures of man-made objects, has been used often for object detection and recognition in computer vision. The observation of vehicles from the stationary cameras are generally symmetrical in the horizontal and vertical directions. This has been used as a cue for vehicle detection in several studies, though it is sensitive to noise [53].

Although few existing systems use color information to its full extent for vehicle detection, color is a very useful cue for vehicle detection, lane road following, etc. Several prototype systems have investigated the use of color information as a cue to follow lanes, or segment vehicles from background. Techniques based on motion and color segmentation have been introduced to detect moving objects by taking advantage of color images. Vehicle segmentation is obtained by frame differentiation while color segmentation consists of a split-and-merge algorithms, [54].

By investigating image intensity, it was found that the area underneath a vehicle is distinctly darker than any other areas on an asphalt paved road. It is found that the shadow underneath a vehicle is a good cue to detect vehicles. Various methods have been proposed and published for the suppression of the moving cast shadow to enhance the detection accuracy of vehicles, [55].

Using constellation of vertical and horizontal edges has shown to be a strong cue for vehicle detection too. There are two types of edges-based detection methods: conventional gradient based edge detectors, and morphological edge detectors. The conventional gradient-based edge detection has found wide acceptance in image processing applications, such as Sobel operator and generalized Hough transform (GHT). However, morphological edge detectors have shown better performance than conventional edge detectors while having a lower computational cost, [54]. Morphological edge detection uses a combination of mathematical morphological operation like dilation and erosion to detect edges. The morphological edge detection highlights sharp grey-level transitions in the input image. However, these edge-based shape detection methods all suffer from the same problem, loss of information during the edge detection stage.

The presence of vehicle in an image causes local intensity changes. Due to general similarities among all vehicles, the intensity changes follow a certain texture pattern. This texture information can be used as a cue to locate the possible vehicle in region of interest for vehicle detection. Only regions with high entropy were considered for future processing. Another texture-based segmentation method uses co-occurrence matrices. The co-occurrence matrix contains estimates of the probabilities of co-occurrences of pixel pairs under predefined geometrical and intensity constraints. The Gabor filter is used to extract different textures and then verified each candidate of vehicles using a classifier, [56].

In the localization process in vehicle detection, experience suggests that better performance could be obtained by exploring task-based a priori knowledge. One of the advantages of 3D model methods is dealing with occlusion. Horizontal and vertical edges are grouped into vehicles using a probabilistic framework. The grouped vehicles are used for tracking in a highway surveillance application. All methods that employ 3-D modeling trade off additional prior information for higher computational complexity. It is necessary to establish the correspondence between the image features and the model features. As a result,

how to generate, and verify the correspondence plays a key role. Ghosh et al proposed a 3D model matching scheme to classify vehicles into various types such as, wagons, sedan, and hatchback in [38].

2.2.2.2   *Motion-based Methods*

The key information mentioned before are using spatial features to distinguish between vehicles and background. Another cue can be employed is motion of vehicles via the calculation of optical flow. Pixels on the images appear to be moving due to the relative motion between the sensor and the scene. The vector field of this motion is referred as optical flow. Motion-based vehicle detection methods use characteristics of flow vectors of moving objects over time to detect moving regions in an image sequence. Optical flow based method can be used to detect independently moving vehicles from camera. However, most of them are computationally complex and very sensitive to noise because it's not very easy to derive a reliable and dense optical flow estimate from a stationary camera. It is also hard to use to analyze the video streams in real-time without having a specialized hardware [8], [56].

2.2.2.3   *Wavelet-based Methods*

Wavelet transform methods have recently been recognized as useful tools for various applications such as signal image processing. For vehicle detection based on wavelet analysis, the motion is characterized via the entire 3-D spatio-temporal data volume spanned by the moving vehicle in the image sequences. These methods generally consider motion as a whole characterizing its spatio-temporal distributions. Wavelet-based methods by taking advantage of spatio-temporal motion characterization are able to acquire better performance both in spatial and temporal information of vehicle motion. Their advantage is low computational complexity and a simple implementation. However, these methods are susceptible to noise and to variations of timings of movements [41].

2.2.3   **Tracking**

Tracking is used to measure vehicle paths in video sequences. This is performed in the following two steps; 1) features for the object or foreground regions are generated in every video frame and 2) a data association step has to provide correspondences between the

regions of consecutive frames based on the features and a dynamic model. Temporal consistency constraints are required to avoid confusion of tracks and to smooth noisy position outputs of detectors. The data association step can use the same distance measure as machine-learning algorithms. The classification result and location in the image is typically included in the feature for this association. There are several techniques used in tracking such as, Kalman filter, spatial-temporal Markov random field, graph correspondence, and event cones [8].

2.3   Summary

Overview on vision-based traffic detection was presented in this chapter. A brief historical background on vision-based traffic detection was followed by mounting and traffic viewing consideration. Finally, an introduction on video detection algorithms was illustrated.

CHAPTER 3   - CMOS IMAGE SENSORS

A solid-state image sensor, also called an "imager," is a semiconductor device that converts photon energy incident on the photo sensitive element (i.e. pixel) into electrical current or voltage. Photons absorbed by the photosensitive materials energize electrons in the valence band into conduction band generating ideally one electron-hole pair per absorbed photon. Image sensor collects these photo electrons and generates electrical signals. Then these signals could easily be read, transferred, processed and stored by the peripheral analog and digital circuitry.

In modern semiconductor industry, silicon is the most widely used material in mass production of very large-scale integration (VLSI) ICs. Moreover, it is also suitable for image sensors because the band gap energy of silicon matches the energy of visible light spectrum. Therefore, silicon can be used for manufacturing image sensors [57].

In this chapter, a brief technological and historical background on complementary metal-oxide semiconductor (CMOS) image sensors with emphasis on active pixel sensor (APS) devices is presented. First, historical background on image sensors is introduced. Second a comparison of CMOS APS and charge-coupled device (CCD) technologies are presented. Third, a generic CMOS image sensor architecture and its functional blocks are introduced. Finally, a summary of the chapter is presented.

3.1   Historical Background on Image Sensors

In the 1960's, before CCD and CMOS APS were born, first generation of image sensors were proposed and developed by several groups around the world. In 1967, Weckler proposed using PN-junctions as pixels operating in photon flux integrating mode, [58]. The generated photo current discharges the initial charge stored on the parasitic capacitance of the PN-junction. The change left in the stored capacitance will be proportional to time integral of the photocurrent. He proposed addressing the pixels using a PMOS switch and converting the current pulse into a voltage pulse using a series of resistors. In 1968, an image sensor with 100x100 array of photodiodes operating in a photon flux integration mode was reported in [59]. Noble described several configurations of self-scanned 2D silicon image detector arrays

with buried photodiodes to reduce dark current and charge integration amplifier for readout in 1968, [60]. In 1969, Chamberlain reported an improved model of the sensor with integral scanning [61]. The main issue with these early solid state image sensors was the fixed-pattern noise (FPN) due to pixel-to-pixel variations. FPN was explored in 1970 by Fry, Noble, and Rycroft, [62].

In 1970, the first CCD image sensor array was reported by W. S. Boyle and G. E. Smith, [63]. CCD image sensors could achieve low FPN and smaller pixel size with higher resolution due to simplicity of the CCD pixel structures. After CCD's introduction, the main focus of research and development had been CCD sensor performance for decades. Criteria such as quantum efficiency, pixel fill factor (fraction of photo sensitive area to total pixel area), dark current, charge transfer efficiency, smear, readout rate, lag, readout noise, full well, and dynamic range favored CCDs over other solid state imagers. Number of pixels integrated in image sensors has been increasing steadily due to the resolution demand of new TV formats (i.e. HDTV) and scientific applications. In parallel, pixel size has been reduced to increase circuit density on sensor and to reduce cost. Because of their low image quality due to immature manufacturing technology, MOS image sensors had received little or no attention and research effort in 1970s and 1980s.

Two independently motivated efforts had led to the rebirth of MOS and CMOS image sensors in early 1990s. The first effort was to create low cost, reasonable performance, highly functional single-chip imaging systems by researchers at the University of Edinburgh in Scotland and Linkoping University in Sweden. The second was the effort by researchers at the U.S. Jet Propulsion Laboratory (JPL) to design highly miniaturized, low-power, instrument imaging systems for next-generation deep space exploration spacecraft needed by the U.S. National Aeronautical and Space Agency (NASA). The technology developed at JPL was subsequently transferred to AT&T Bell Labs, Kodak, National Semiconductor and several other major US companies, including Photobit (later Aptina, today On Semiconductor). These efforts have led to significant advances in CMOS image sensors and the development of the CMOS APS imagers. The CMOS APS imager performance had become competitive with CCDs in terms of read noise, and dynamic range as of early 2000s. A noteworthy one is the CMOS APS imager developed by A. Krymski *et al*. from Photobit Corp. that was featured as 500 frames per second frame rate, 1024 x 1024 array size, 8-bit digital output with 450mW

power consumption in 1999, [64]. By 2009, market share of CMOS image sensors suppressed CCDs in practically every application market.

3.2   *Comparison of CMOS Imager and CCD Imager Technologies*

Charge-coupled devices (CCDs) and CMOS image sensors are both mainly build in silicon substrates. Optical properties of the substrate dictate image sensor response to light. Therefore, CCDs and CMOS image sensors have similar responses to light. The main difference between CCDs and CMOS imagers are readout procedures and process steps used during fabrication.

The CCD imager is basically an analog shift register that shifts electrons collected in pixels to an output amplifier. The virtues of the CCD include its high sensitivity, high fill-factor, and large array formats. The high sensitivity arises from high net quantum efficiency, the high fidelity of reading out the CCD, and the low noise output amplifier [4]. However, the need for nearly perfect charge transfer operation is the Achilles' heel of CCDs, which requires extremely precise and expensive fabrication process, precluding efficient integration of camera-on-chip, [5].

CCD has analog output(s) running at high frequency with a large voltage swing. Driving analog data off-chip requires very large bias current to charge output pad capacitors at the serial data rate because the analog output must settle quickly to preserve dynamic range. In contrast, the CMOS APS imager with on-chip ADC(s) has digital output. Driving digital output pad requires less current since digital output is much more robust and settling is not an issue. The overall power consumption of pixel readout circuits in CMOS APS imagers such as pixel source followers and charge amplifiers are significantly lower compared to CCD output amplifiers, [65].

Because of suffering from large fixed-pattern noise (FPN) caused by dark current non-uniformity, early CMOS image sensors were suspected by many skeptics, though they have the ability of low power, high operation speed, and better cost-efficiency. However, with the recent introduction of pinned photodiode (PPD) technology, CMOS image quality issues are being resolved rapidly. The PPD active-pixel configuration and micro-lens techniques, together with on-chip signal-processing circuits, offers even lower temporal noise performance than that of CCD imagers. Moreover, the reduction of pixel sizes in CMOS

image sensors has been remarkable in recent years, due to the smaller process technologies and the introduction of shared-pixel configurations [57].

With those technology improvements, CMOS image sensors have surpassed CCDs in most markets, such as market of smart phones, digital still cameras and digital single lens reflex (DSLR) cameras. The state-of-the-art of CMOS image sensors is supplanting the CCDs with excellent image quality, higher speed and pixel rates, faster readout rate, lower power consumption, and economical fabrication process.

3.3   CMOS Image Sensors

When a flux of photons is incident on a semiconductor material, the photons that have energy exceeding the semiconductor's bandgap energy generate electron hole pairs. Photon energy is given by (3.1).

$$E_{photon} = h\nu = \frac{hc}{\lambda} \geq E_g \tag{3.1}$$

Because the band gap energy, E_g, of silicon is 1.1 eV, light with wavelengths longer than 1100 nm will not have enough energy to be converted to electric signals. On the other hand, silicon material absorbs photons with wavelengths only shorter than 1100nm [57].

The collected charge is converted into measurable voltage or current quantities. The analog voltage or current is buffered, processed and converted to digital data. These operations can be done at different levels. In CCDs charge transfer and buffering are on chip and signal processing and digital conversion are off chip. In CMOS image sensors buffering is done in pixels. Readout, signal processing and digital conversion may be global, column level or in each pixel.

CMOS image sensor system in general composes of a pixel array, row and column addressing circuits, column readout circuits, analog signal processing (ASP) unit and analog to digital converter block as shown in Figure 3-1. Column parallel image sensors would have an ASP and ADC for each column of pixels with a shift register or a multiplexer for reading out the ADC outputs. In column series architectures, a global ADC is used for all columns [5].

The voltage and current reference, timing generator, and logic controls are also required for a single chip imager system.
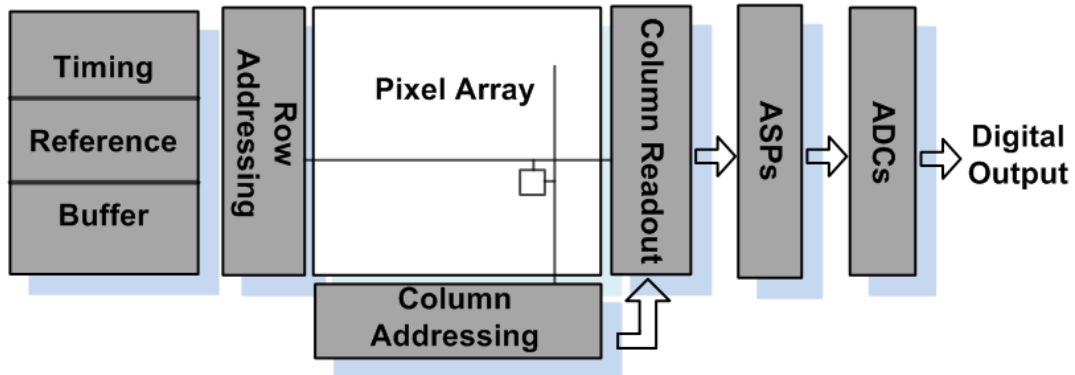
Figure 3-1. CMOS imager sensor block diagram.

3.3.1 CMOS Pixels

There are two basic structures used for accumulating charges, which are main part of a pixel. The first one is a photodiode, and the second one is a photogate as shown in

Figure 3-2 and Figure 3-3, respectively. These two basic structures have evolved into three fundamental approaches to build pixels in CMOS imagers: photodiode-type passive pixel, photodiode-type active pixel, and photogate-type active pixel, [5].
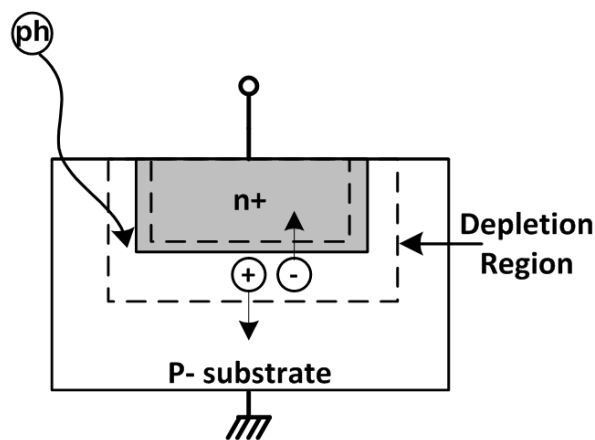
Figure 3-2. Cross-sectional view of photodiode.
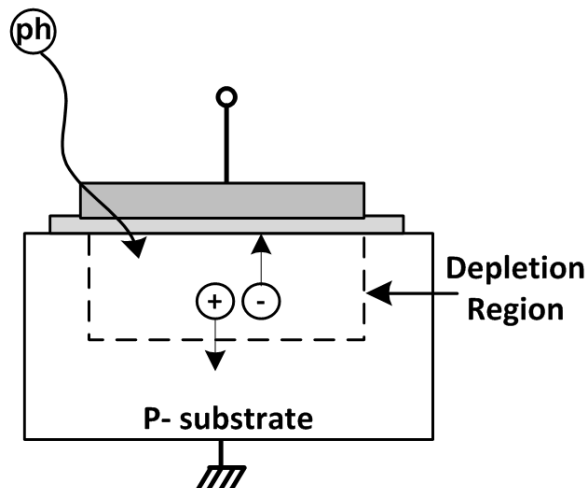
Figure 3-3. Cross-sectional view of a photogate.

3.3.1.1   *Photodiode and Photogate*

The cross section view of photodiode is shown in Figure 3-2. P-substrate of it is grounded and the n+ region is first reset at a positive voltage. It then becomes electrically floating with the reverse bias condition being held. Electrons excited by photons tend to collect at the n+ region, reducing this region's potential, while holes flow to the ground terminal. In this case, the electrons are the signal charge [57].

In Figure 3-3, the cross section of a photogate is displayed. When a positive voltage is applied to the gate electrode, the holes in the surface of the substrate are repelled and a depletion layer is formed under the gate oxide, while the majority of carriers (holes) are depleted [57], and photo generated charge under the gate are collected in the depletion region.

3.3.1.2   *Passive CMOS Pixels*

The photodiode-type passive pixel was first proposed by Weckler in 1967, [58], [59]. Figure 3-4 shows the basic structure of a passive pixel which consists of a photodiode and an access switch. The voltage on the photodiode is reset to the column bus voltage as the photodiode is accessed in the pixel. The charge transferred to the photodiode, is converted to an output voltage by the column amplifier through column bus. The fill factor is high because only one access transistor is integrated in each pixel along with the photodiode. As a result, quantum efficiency of the passive pixel structure is high and pixel size could be small.
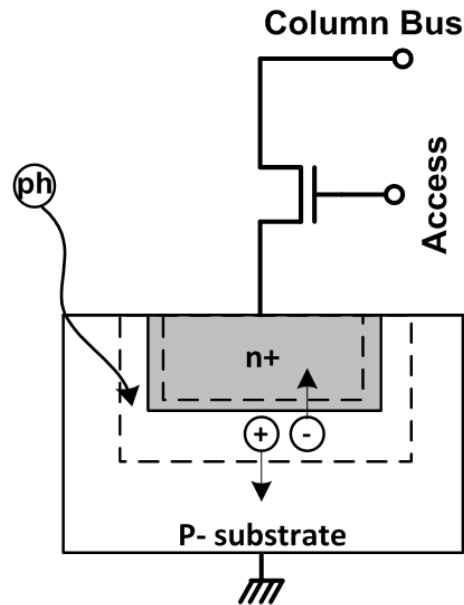
Figure 3-4. Passive CMOS pixel schematic.

The readout noise is the biggest problem with the passive pixel. Readout noise in a passive pixel is typically on the order of hundred(s) of electrons r.m.s. while it is less than few electrons r.m.s. in a CCD. Another issue is scaling problem. The passive pixel also does not scale well to larger array sizes and faster pixel readout rates since more pixels connected to a column bus increases the bus capacitance. Increased bus capacitance and faster readout speed results in higher readout noise and higher power consumption. Passive pixel sensors also suffer from large fixed pattern noise from column amplifiers.

3.3.1.3 *Active Pixels*

An active pixel means a pixel including an in-pixel amplifier, which is typically a source follower type amplifier. The image sensor equipped with active pixel array is named active pixel sensor (APS). The in-pixel amplifiers are enabled only during pixel readout. Therefore, the power consumption of pixels is relatively low, and the readout could be addressed in pixel level. A typical three-transistor (3-T) active pixel configuration is shown in Figure 3-5. Basically, all CMOS active pixels have the same operation, though complex pixel topology with added transistors for improving performance is available [57].
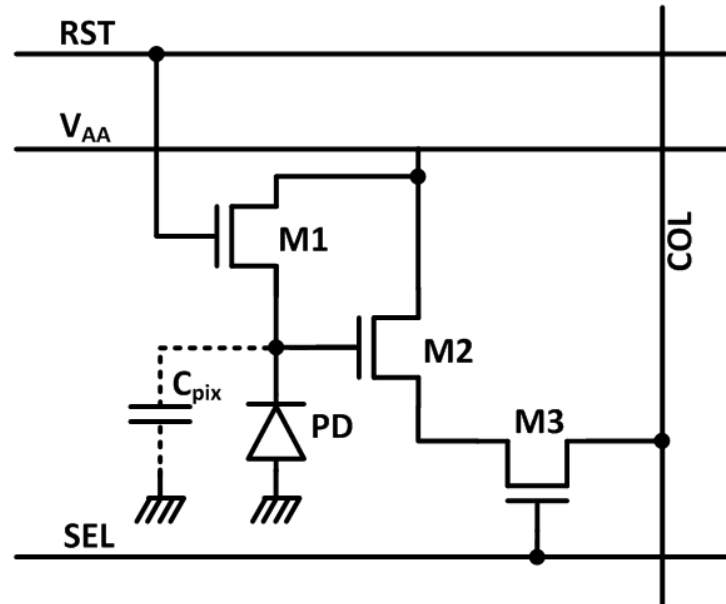
Figure 3-5. Configuration of 3-T active pixel including source follower type amplifier readout.

3.3.1.3.1   Photodiode Type CMOS Active Pixels

The photodiode-type active pixel was proposed for solving the issues with passive pixel in 1968, [60]. However, the first high-performance photodiode-type active pixel was demonstrated in 1995 by JPL researchers, [66]. The photodiode-type APS pixels have high quantum efficiency as there is no overlying polysilicon. Typical photodiode-type APS uses 3T structure per pixel as shown in Figure 3-6, where M1 is the reset transistor, M2 is the pixel source follower and M3 is the row select transistors.

Reset transistor resets the photodiode node to reset level V_{RST} in the beginning of integration period, and then the voltage is stored on the depletion capacitance of the photodiode. The built in electric field drifts photo generated electrons to N-region and holes to the grounded P-region. Drifted photo generated electrons discharge the photodiode capacitance during the integration period. The final voltage, V_{pix}, at the end of integration period dependents on the light level. Pixels receiving more light will discharge faster than those receiving less light. The M3 is controlled by the signal from row addressing circuits. When a row select pulse is applied at the gate of M3, the M2 and a bias current load form a source follower circuit. The photodiode voltage, V_{pix}, is buffered by the source follower, and

the buffered output voltage at a $Vout_{pix}$ node is sampled on a sample-and-hold capacitor. The column-addressing circuit scans the sampled signals during the horizontal scanning period.
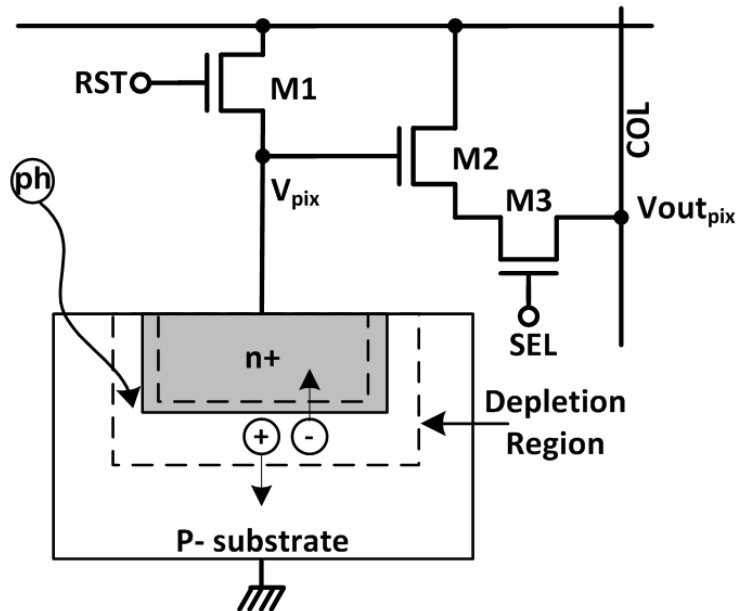
Figure 3-6. Photodiode type 3T CMOS APS Pixel.

More complex CMOS APS pixel structures with added transistors for improving performance are also available. However, all CMOS active pixels have the similar basic operation as described earlier.

3.3.1.3.2   *Photogate Type CMOS Active pixels*

The photogate type APS was developed in 1993 for high performance scientific imaging and low light applications at JPL, [67]–[69]. Photogate concept was adapted from CCD technology for low noise charge collection. The photogate type APS pixel schematic is shown in Figure 2-7.

Signal charge is integrated under the photogate while the floating diffusion node (FD) and transfer gate (TX) are added to the pixel for correlated double sampling and low noise readout, which cause photogate-type active pixel is larger than those based on photodiode-type due to the added components. Floating diffusion (FD) node is reset and the reset level is read first. Then the charge under the photogate is transferred to the FD node by pulsing the photogate and turning the transfer (TX) gate on. The FD node voltage is read after the charge

transfer again. The difference between the values before and after charge transfer is the absolute signal value accumulated under the PG during integration period.

In photogate type pixel, the surface of the channel has better quality than that of the photodiode leading to lower surface dark current. In addition, there is no stress related dark current component in photogate. These lead to a very low dark current.
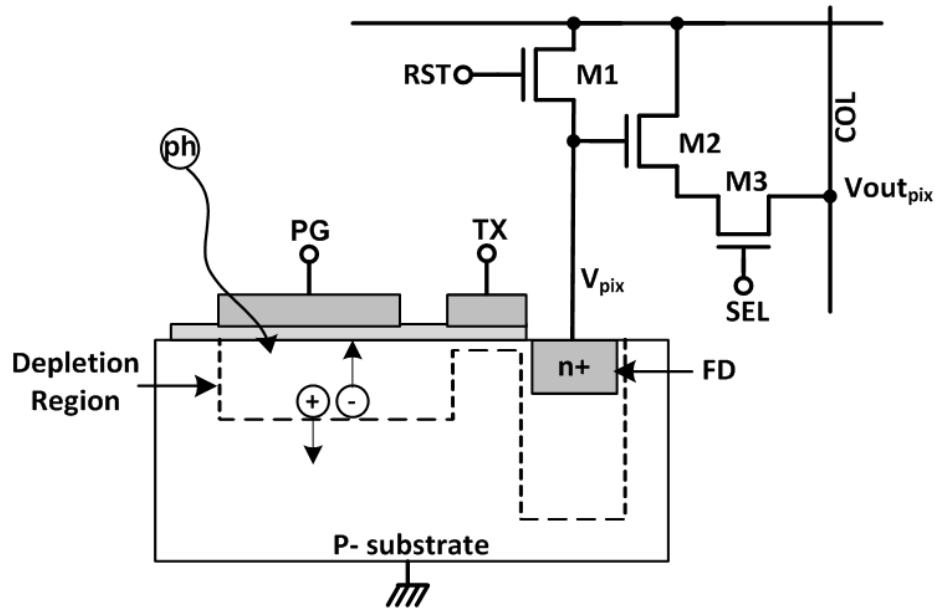
Figure 3-7. Photogate type CMOS APS Pixel.

3.3.2 Analog Signal Processing

Once the voltage at the source follower output is stored on sample and hold capacitors, pixel readout is completed for an active pixel sensor. Reset noise, 1/f noise, and fixed pattern noise (FPN) due to the pixel-to-pixel process parameter variations are largely suppressed by correlated double sampling (CDS) operation in ASP [5]. Typically, sample and hold circuit is followed by a programmable gain charge amplifier in the ASP. After the CDS and gain operations in ASP, the pixel data is sent to the analog-to-digital converters (ADCs). Other analog signal processing functions may also be implemented in the ASP.

Table 3-1. Comparison of different ADC topologies integrated in CMOS image sensors.

ADC Type	Resolution	Speed	Power	Features
Flash	8 bits	10 Msps – 1Gsps	>50 mW	+ Extremely Fast - Highest power consumption - Large die size - Expensive
SAR	8 -16 bits	75 Ksps -2 Msps	~0.1 mW	+ High resolution + Lowest power consumption - Limited sampling rate - Low input bandwidth
Ramp	>14 bits	<100 Ksps	~1 mW	+ High resolution + Low power consumption - Low speed
Pipeline	10-14 bits	10 Msps – 100 Msps	>10 mW	+ On-chip self-calibration + Digital error correction + Fast speed - Requires 50% duty cycle
Sigma-Delta	>14 bits	> 200 Ksps	>10 mW	+ Very high resolution + High input bandwidth + Digital on-chip filtering - limited sampling rate

3.3.3 Analog to Digital Converter (ADC)

Analog pixel outputs can be converted to digital code on chip in a CMOS imager. Depending on the system architecture, a single global ADC operating at high speed, column ADCs operating at lower speeds or in pixel ADCs might be used.

CMOS image sensor typically includes ADC in the same chip for digital output which has better noise immunity than analog output in addition to lower power consumption. The on-chip ADC could also reduce the cost on overall system with fewer component counts.

The ADC used in CMOS image sensor should support video data rates depending on the resolution and frame rate. The ADC usually has at least 8-bit resolution with low integral

non-linearity (INL) and differential non-linearity (DNL) so that the image quality is not degraded. The power consumption and area of ADC should be considered as well.

There are different topologies of ADC are integrated in CMOS image sensor [70]–[73]. The comparison of different topologies is listed and compared in

Table 3-1.

3.4   Summary

In this chapter, a concise historical background on solid state image sensor was introduced. In addition, the comparison between CMOS imager and CCD imager was illustrated. Finally, the basics of CMOS image sensor including pixel structures and related functional blocks were presented.

CHAPTER 4 – CAMERA CALIBRATION

Vision systems are being widely used in security, surveillance, and traffic detection applications due to ease of installation and maintenance [6], [7]. These systems are often required to have the ability to extract real-time parameters from the observed objects, such as number, location, size, speed, etc. For instance, it is important to determine the vehicle size, vehicle location and vehicle speed in real-time for modern intelligent transportation systems [8]. Therefore, a technique or algorithm which could determine the relationship between two-dimensional (2D) captured image by the camera and three-dimensional (3D) object exist in the field of view (FOV) of the camera is necessary for extracting object's location, speed and size in traffic and surveillance application [7], [13], [14].

Camera calibration is used for determining intrinsic and extrinsic parameters or the camera systems including geometric and optical characteristics of the camera, and the parameters of 3D position of the camera with respect to a real-world coordinates, so as to achieve accurate mapping between the 2D image plane and 3D world image, [21]. Various camera calibration methods have been published for photogrammetry, [74], [75], machine vision, and computer vision applications, [7], [20], [21], [25], [29], [76]–[79]. These techniques could be classified into two categories; conventional calibration methods and self-calibration methods.

In this chapter, camera calibration techniques are reviewed in the literature. First, the projection methods for camera are concisely illustrated. Second, a pin-hole camera model followed by basic terminologies is introduced. Finally, both conventional calibration methods and self-calibration methods are presented.

4.1 Projection Methods

A space is a collection of points in the realm of mathematics. In plane geometry, Euclidean space refers to the 3D physical space defined by three perpendicular (orthogonal) axes. Projections are formed by the intersection of lines (projectors) which emanate from a center of projection (COP) with a projection plane. Typically, projection methods are broadly grouped under parallel projection and perspective projection, the schematics of which are shown in Figure 4-1 and Figure 4-2, respectively [79].
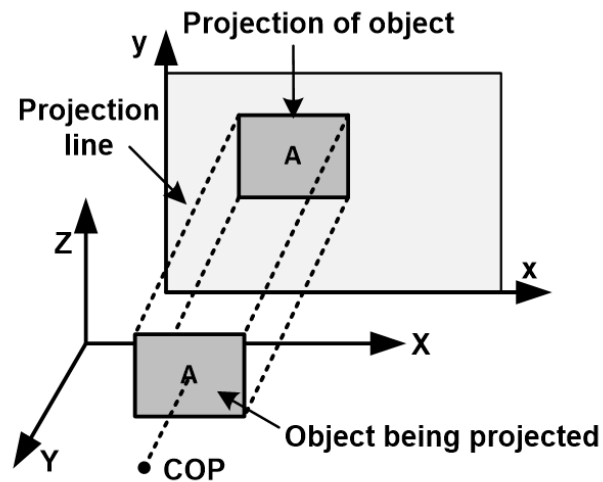
Figure 4-1. Schematic of parallel projection.
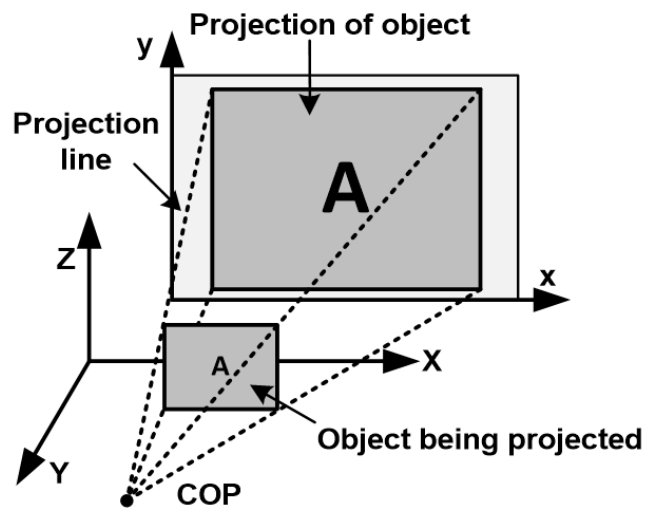
Figure 4-2. Schematic of perspective projection.

4.1.1   **Parallel Projection**

A parallel projection is a projection of an object in three-dimensional space onto a projection plane, where the projection lines are parallel to each other as shown in Figure 4-1. It is a basic tool in descriptive geometry.

There are two types of parallel projection: orthographic (parallel) projection and oblique (parallel) projection. In orthographic projection, all the projection lines are orthogonal

to the projection plane, resulting in every plane of the scene appearing in affine transformation on the viewing surface. On the other hand, oblique projection is a parallel projection with two tilted directions, in which the projection lines are not orthogonal to the projection plane.

4.1.2 Perspective Projection

In early Renaissance Italy, the projective geometry was initiated for an accurate depiction of three-dimensional scenes on two-dimensional surfaces or planes by the painters, who wanted to produce a convincing illusion of 3D depth in their architectural paintings [80]. For instance, Figure 4-3 shows a painting, *The Delivery of the Keys*, created by Pietro Perugino who show good understanding of perspective on his works.

The edges of a pair of straight railway are parallel lines in 3D real world. However, in the 2D image they appear to converge as they recede towards the horizon as shown in Figure 4-4. Any pair of parallel lines appears to meet at the point of the horizon corresponding to their common direction. Moreover, any two horizontal planes appear to come together in the distance and intersect in the horizon line.

Figure 4-3. An early Renaissance era painting, *The Delivery of the Keys*, applying perspective geometry by Pietro Perugino.

In perspective projection, parallel lines that are not parallel to the projection plane converge to a vanishing point. The concept of vanishing points is well illustrated in Figure 4-5. A vanishing point is a point on the image plane of a perspective projection where the two-dimensional perspective projections of mutually parallel lines in three-dimensional space appear to converge.

Figure 4-4. Image of a pair of straight railways.
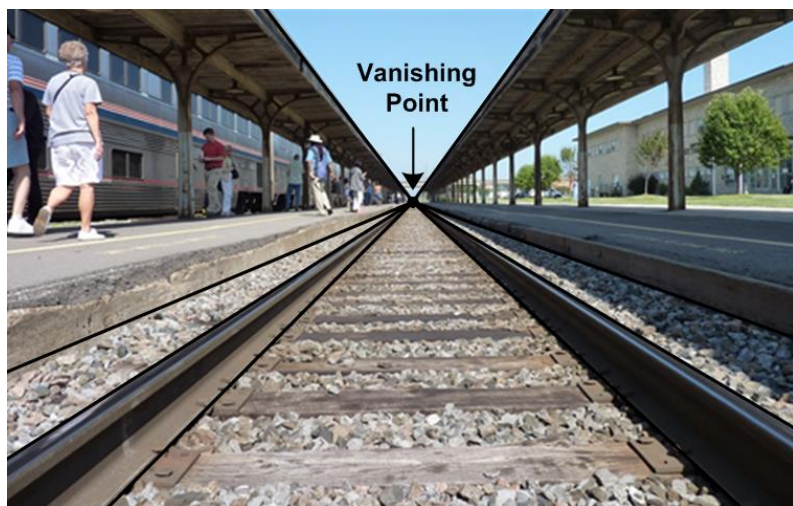
Figure 4-5. Image of a pair of straight railways labeled vanishing point.

In imaging, projection refers to the process of converting the brightness of a 3D object scene into a spatial intensity distribution on a 2D image plane (projection plane) through perspective transformation with COP placed at the geometric center of the lens. The optical

imaging process is a combination of 3D world coordinates to 3D camera coordinates of each object point followed by perspective projection, which is about 3D image scene to 2D image plane.

4.2 Pin-hole Camera Model

There is a strict geometric relationship between the 3D spatial points and the points in the 2D image, derived from pin-hole imaging principle. A basic pin-hole camera model is shown in Figure 4-6, which is based on Fung's model [13] and Haralick's perspective transformation [81].
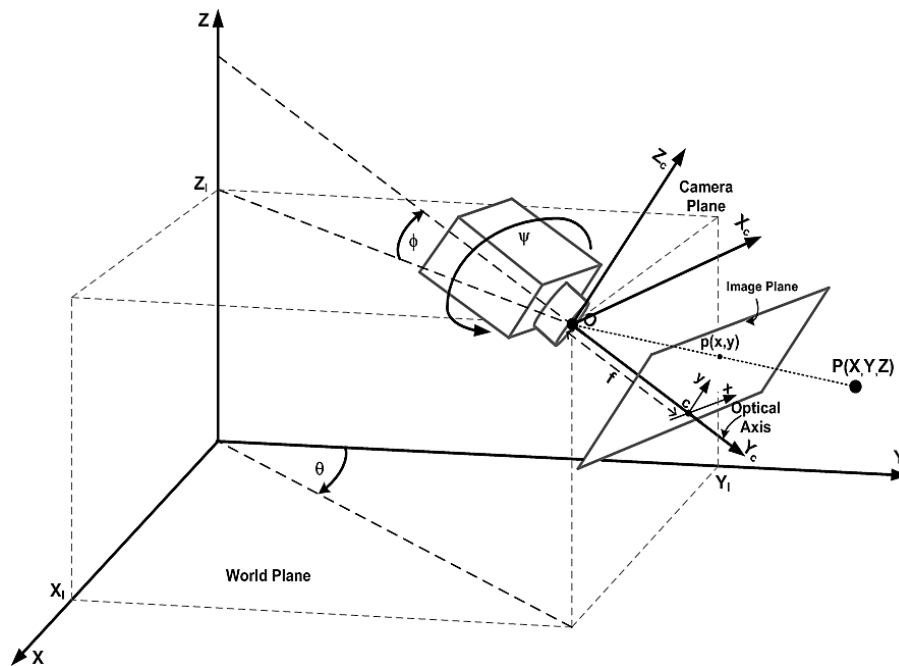
Figure 4-6. Pin-hole camera model.

In this model, *P(X,Y,Z)* is assumed as an arbitrary point in the 3D world coordinates, and *p(x,y)* is assumed to be point *P* projected onto the 2D image coordinates. The camera is mounted at certain height, which is the perpendicular distance from the center of the camera lens to the X-Y plane, and looking down with certain tilt angle, ϕ, pan angle, θ, swing angle, ψ, and focal length, *f*. Moreover, the tilt, pan and swing angles are the extrinsic parameters, which are used to determine the relationship between world coordinates and camera coordinates, besides 3 translation vectors, X_l, Y_l, and Z_l. On the other hand, the parameters only depending

on the inner configuration of camera are named intrinsic parameters, such as focus length (f), and image center that is the intersection of the optical axis with the image plane.

Specifically, pan-tilt cameras, which have ability to rotate around X_c-axis for tilting and Z_c-axis for panning with fixed swing angle 0 degree, are commonly used in road/traffic detection and surveillance [25], [28].

According to homogeneous coordinates representation and perspective transformation in 3D [81] with both intrinsic and extrinsic parameters, equation (4.1) with (4.2) could be achieved.

$$
\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} =
$$

$$
\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & f & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}
\begin{bmatrix} \cos\psi & \sin\psi & 0 & 0 \\ -\sin\psi & \cos\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & \sin\phi & 0 \\ 0 & -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & -x_l \\ 0 & 1 & 0 & -y_l \\ 0 & 0 & 1 & -z_l \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}
$$

$$(4.1)$$

$$
\begin{aligned}
x &= x'/w \\
y &= y'/w \\
z &= z'/w
\end{aligned}
$$

$$(4.2)$$

Then the x and y value of point p on the image plane projected from P, could be achieved by (4.3) and (4.4) derived from (4.1) and (4.2).

$$
x = \frac{f[X(\cos\theta\cos\psi + \sin\theta\sin\phi\sin\psi) + Y(\sin\theta\cos\psi - \cos\theta\sin\phi\sin\psi) + Z\cos\phi\sin\psi]}{-X\sin\theta\cos\phi + Y\cos\theta\cos\phi + Z\sin\phi + z_l/\sin\phi}
$$

$$(4.3)$$

$$
y = \frac{f[X(-\cos\theta\sin\psi + \sin\theta\sin\phi\cos\psi) + Y(-\sin\theta\sin\psi - \cos\theta\sin\phi\cos\psi) + Z\cos\phi\sin\psi]}{-X\sin\theta\cos\phi + Y\cos\theta\cos\phi + Z\sin\phi + z_l/\sin\phi}
$$

$$(4.4)$$

The inverse projection, X and Y value of point *P* in real world, could also be derived in terms of the same set of camera parameters as showing (4.5) and (4.6).

$$X = \frac{\sin\theta(z_l/\sin\phi + Z\sin\phi)(x\sin\psi + y\cos\psi) + \cos\theta(z_l + Z)(x\cos\psi - y\sin\psi) - Zf\cos\phi\sin\theta}{x\cos\phi\sin\psi + y\cos\phi\cos\psi + f\sin\phi}$$

(4.5)

$$Y = \frac{-\cos\theta(z_l/\sin\phi + Z\sin\phi)(x\sin\psi + y\cos\psi) + \sin\theta(z_l + Z)(x\cos\psi - y\sin\psi) + Zf\cos\phi\cos\theta}{x\cos\phi\sin\psi + y\cos\phi\cos\psi + f\sin\phi}$$

(4.6)

Equations (4.5) and (4.6) could be used to determine the 3D coordinate of a point in the image above the ground level, as $Z \neq 0$. The height of the point needs to be entered the equation to estimate its *X* and *Y* in 3D world coordinates. If the height information for that point is not available, it can be determined by transforming two points with the same *X* and *Y* coordinates with different heights, *Z*, from 2D to 3D using (4.5) and (4.6). Then enter Z into the equation for the upper point by doing iteration until the X and Y coordinates of both points match those in 3D space.

For simplicity, equations (4.7) and (4.8) could be used when *Z=0*.

$$X = \frac{z_l\sin\theta/\sin\phi(x\sin\psi + y\cos\psi) + z_l\cos\theta(x\cos\psi - y\sin\psi)}{x\cos\phi\sin\psi + y\cos\phi\cos\psi + f\sin\phi}$$

(4.7)

$$Y = \frac{-z_l\cos\theta/\sin\phi(x\sin\psi + y\cos\psi) + z_l\sin\theta(x\cos\psi - y\sin\psi)}{x\cos\phi\sin\psi + y\cos\phi\cos\psi + f\sin\phi}$$

(4.8)

All the equations introduced above are based on the ideal pin-hole camera model, the ideal of which means there is no lens-distortion or camera assemble mismatch. Nonlinear optimization techniques could be used for the distortion [20].

4.3   Current Camera Calibration Methods

Camera calibration is used for determining intrinsic and extrinsic parameters of the camera systems including geometric and optical characteristics of the camera, and the parameters of 3D position of the camera with respect to a real-world coordinates, so as to achieve accurate mapping between the 2D image plane and 3D world image, [21]. Various camera calibration methods have been published for photogrammetry, [74], [75], and machine and computer vision applications, [7], [20], [21], [25], [29], [76]–[78]. These could be classified into two categories; conventional and self-calibration methods.

4.3.1   **Conventional Calibration Methods**

Conventional calibration methods have been widely investigated in the past. It usually requires a specific object with marked geometric patterns placed in 3D world, such as diamond-shaped calibration pattern, [19], 16 squares pattern [20], chessboard calibration pattern [21], parallel circles [82], virtual grid [83], and others, [22], [23]. However, these man-made patterns are hardly practical for many applications, such as security and traffic surveillance. Techniques utilizing road-lane markings as calibration targets were also proposed, [10], [13], [14], [24], which reduces calibration complexity and increases their practicality in real-time applications.

Fukui [19] used a diamond-shaped calibration pattern placed on a wall to determine the location and horizontal deviation of the camera with respect to the calibration pattern. The method assumes that the optical center and the calibration pattern must be set at the same height, and the dimensions of the calibration pattern must be known as priority. Practically, although diamond-shaped objects may be found on the road, camera systems are usually located at different heights with variant mounting conditions, thus it could not be general method for real traffic detection application.

In [84], a method is presented that instead of using man-made patterns, the corner between a wall and a ceiling was used as a calibration pattern. The three intersecting lines of the planes form a 'Y-shape,' which varies according to the viewing angle of the camera. The camera parameters can be calculated if the distance between the camera and ceiling is known

in advance. However, it is still hardly being applicable in the practical traffic detection, though walls and ceilings can be found indoors for experiment testing.
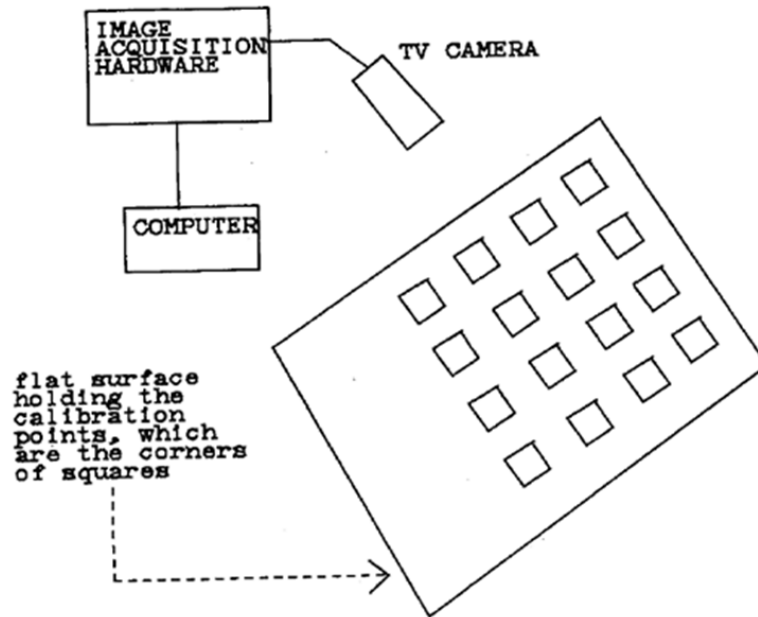
Figure 4-7. Schematic diagram of experimental setup for camera calibration using monoview coplanar set of points [20].

A versatile camera calibration technique depending on 16 (4x4) black squares is proposed in [20] and depicted in Figure 4-7. The 4 corners of each square are detected by using image binary, edge detection and special interpolation technique before starting calibration. Then, perspective projection with pin-hole geometry was used to transform these points of corners for calibration, and optimization techniques such as steepest descent were used to correct lens distortion. The major drawback of this approach is that a large amount of 3-D data (64 corners) has to be handled during calibration, and a special or specific standard block should be prepared, otherwise, it is still impossible to find 16 squares on the road as calibration points as shown in Figure 4-8.
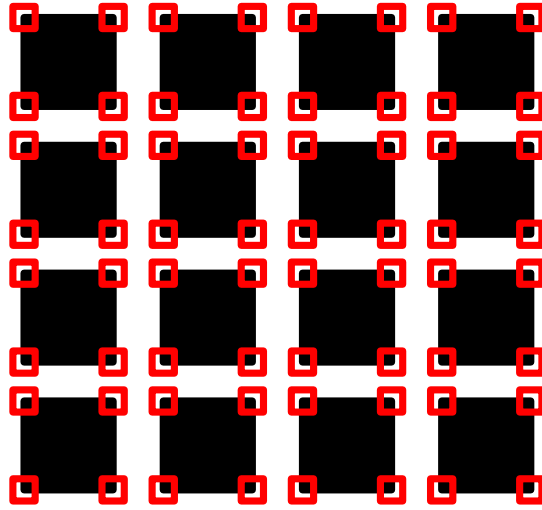
Figure 4-8. A 4x4-square array for calibration proposed in [20].
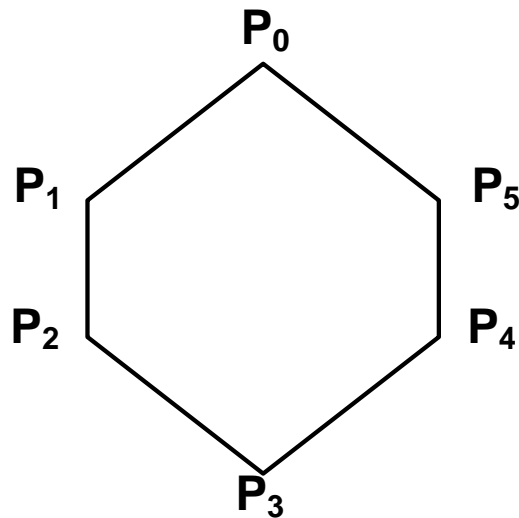
Figure 4-9. The hexagonal calibration target used in [85].

A hexagon as shown in Figure 4-9 is employed as the calibration target to generate a vanishing line of the ground plane from its projected image in [85]. Theoretically, a vanishing line can be formed when the hexagon is projected from a 3-D scene onto a 2-D image in Figure 4-10. The vanishing line includes useful geometric hints about the camera orientation parameters and the focal length, from which the orientation parameters can be solved analytically [81]. For none ideal cases, a least-squares algorithm can be employed to fit a straight line to the vanishing points, and the camera parameters can be calculated based on the geometric hints of the fitted line. The hexagonal pattern is more feasible than the 4x4-square

array to be handled simplifying the feature extraction in relevant image processing. However, it is still facing the difficulty of finding perfect hexagonal objects on the road.
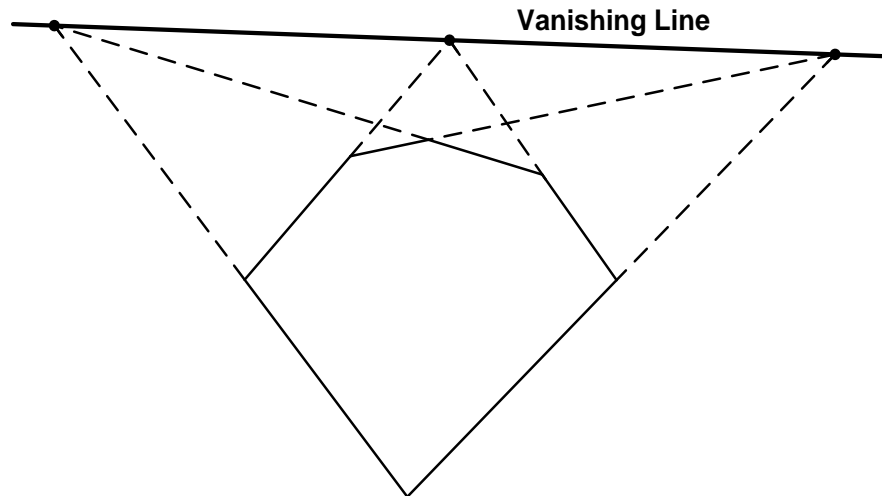
Figure 4-10. Hexagonal calibration target projected from a 3-D image onto a 2-D image with vanishing line in [85].

The road-lane markings were initially used for calibration, [86], where the edges of the road curb is utilized to generate calibration model. Four points (a, b, c and d) were chosen arbitrarily from the left and right edge as shown in Figure 4-11. The camera height, h, pan angle, θ, and tilt angle, ϕ, are assumed known. Thus, the vanishing point can be used to compute the rest of the camera parameters. To compare with the prior techniques, it is not required to set or use specific pattern(s) for calibration. On the other hand, only one vanishing point is assessed for the calculation. However, in reality, the edge of the road curbs may not clear or even not exist in the field of view. Moreover, this approach also requires large amount of calculation.

Vanishing points and vanishing lines which provide information on the direction of lines and orientation of planes were extracted in [7], [9]–[11] to recover the image with affine properties of the perspective structure that could be used to estimate length of straight lines with reference height of a certain object in the scene. Although these are mature methods without requiring knowledge of the camera's pose, the information in the image, which does not include much geometric cues, may not be sufficient to generate vanishing points or vanishing lines. In [87], a rectangular marker object with known size was used on the reference plane instead of

generating vanishing point or lines, which introduced back-projection method to obtain the height information. However, it still requires a specific pattern in 3D world which is not always suitable for real world applications.
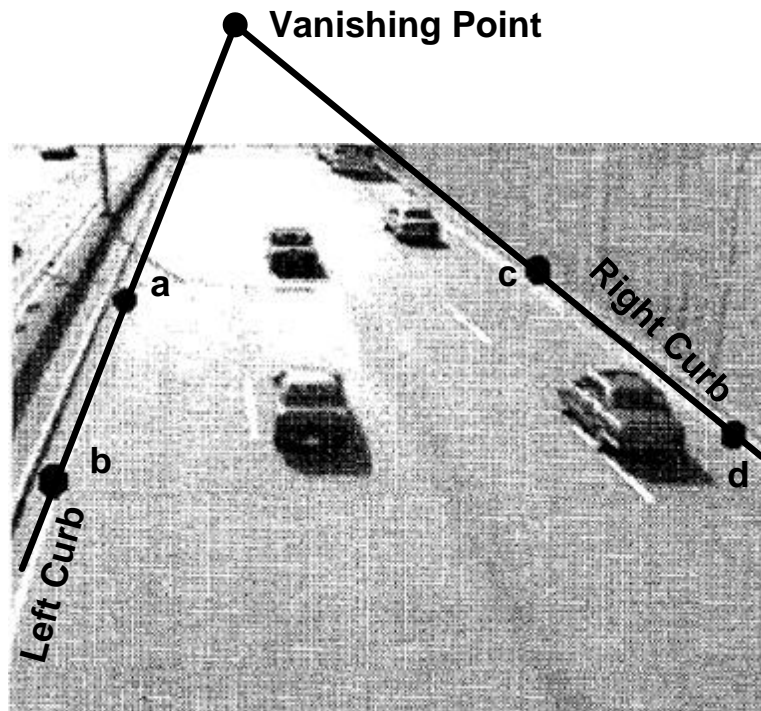
Figure 4-11. Selected points of right and left road curbs for generating vanishing point as calibration hint [86]

In [88], a three-stage dynamic camera calibration method is proposed. Instead of using fixed objects on the road for calibration, parallel and perpendicular lines on the road are identified through the analysis of extracted vehicle blobs as shown in Figure 4-12. An activity map of the traffic flow is created to identify lines that are parallel to the road direction, and the bottom edges of vehicles are used to identify perpendicular lines. Consequently, two vanishing points could be formed to calculate the camera parameters. Basically, it is still two-vanishing-point camera calibration method. The major drawback of this model is that it would be inaccurate in situations where vehicles change lanes frequently, and also requiring a huge amount of computation power.

A rectangular shape defined by road-lane markings which forms two vanishing points for calibration is shown in Figure 4-13[13]. The model requires the lane width (w), which

should be obtained from the road construction database or directly measured on the road. In [13], the proposed pattern used for calibration is not difficult to found in practical road, and then tow-vanishing-point calibration method could be used to compute camera parameters through pin-hole camera models. However, if two side-by-side road-lane markings are not completely aligned, then a rectangle is unlikely to be formed. It will still require large amount of calculation since that the model depends on two vanishing points. Moreover, one of the vanishing points might reach infinity that will cause error in the process of estimating parameters, as the camera viewing angle is parallel to the road direction.
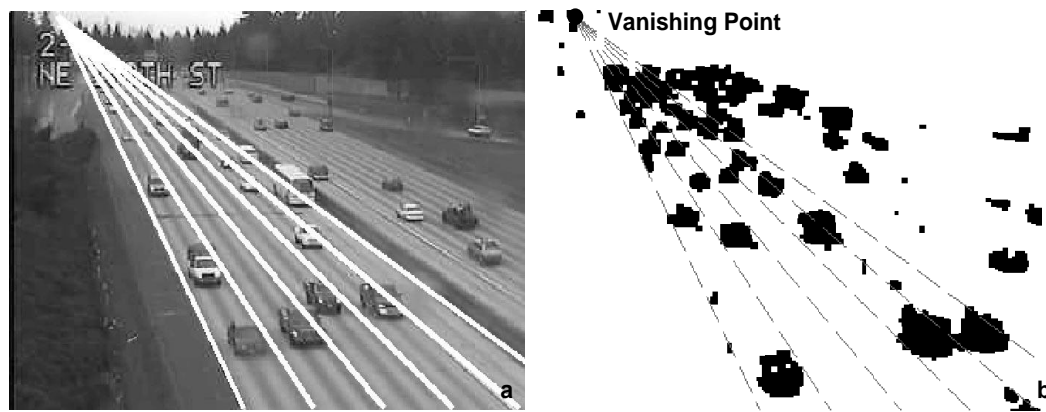
Figure 4-12. Dynamic camera calibration method used in [88]. a) Sample image with superimposed lane boundaries. b) Activity binary image of car flows to identify lines that are parallel to the road direction.

A criterion was proposed to reduce the estimation error for selecting only one of the two vanishing points if one of the points is approaching infinity [14]. The improved model of rectangle calibration method was shown to yield good performance in near-parallel cases. The tradeoff is that an additional parameter, the length of the lane markings, is needed in the model. An improved method that accept non-perfect rectangular is proposed for making calibration by trapezoidal patterns or parallelograms [7]. The model requires four reference points to determine the camera parameters as shown in Figure 4-14, increasing the flexibility and generality of the camera calibration models.
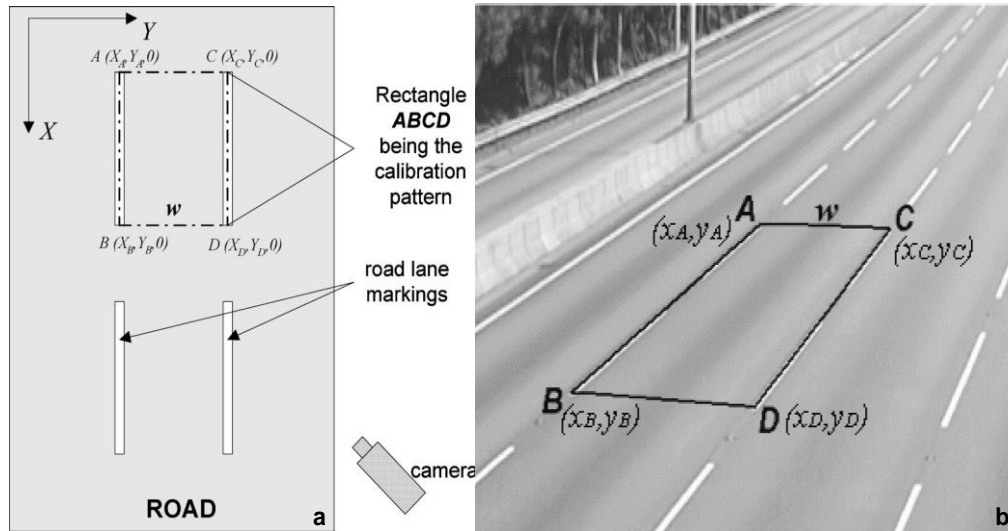
Figure 4-13. Defined rectangle ABCD used for camera calibration. a) Top view of the rectangle ABCD selecting. b) Rectangle ABCD shown in the image. [13]

However, those works used in traffic application are based on direct linear transformation and vanishing-point camera calibration method which requires complex computational steps mapping the coordinates from image domain to real-world domain. Additionally, all those conventional methods require a specific pattern in 3D world and the size of these patterns must be known.
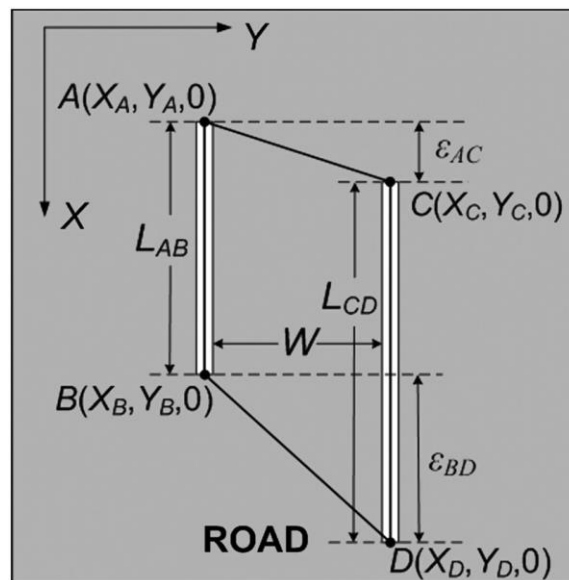
Figure 4-14. 4 points selection for camera calibration [7].

4.3.2 **Self-calibration Methods**

The methods based on self-calibration obtain intrinsic and extrinsic parameters of the camera systems without requiring any calibration rig or patterns in 3D world. It requires multiple images of a dynamic scene taken at different locations at the same time or multiple images of a static scene taken at different times. Maybank and Faugeras illustrated the concept of self-calibration in [29], [30] including Kruppa equations that could be solved for the coefficients of the image of the absolute conic. At least three images are needed to determine both internal and external parameters while mapping the 2D image and 3D world, or vice versa, [25]–[28], [89], [90]. Main disadvantages of the self-calibration method are that it is nonlinear, highly sensitive to noises, and many parameters are determined using complex iteration procedures. To address these, several modifications were proposed such as using active vision, [27], [89], using more constraints [91], [92], and referring to a known camera [90].

Multi-camera based methods are employed in [93], [94], which generate 3D reconstructed image from aligned multiple cameras that looks into a single planar coordinate while using approximate values of intrinsic camera parameters. Multiple camera-based method gives larger surveillance area than that of a single view camera. However, it requires extra computation power for calibrating whole camera coordinates with respect to the common reference plane. Typically, the ground plane is used as reference plane, which is nonlinear, sensitive to noises, and environmental variations. Further, this method requires iterative computational procedures, expensive moving camera systems, and tedious setup of several cameras.

The camera calibration methods described so far rely on the use of Kruppa equations, [29], and at least three images for self-calibration, and special or specific patterns for conventional camera calibration methods. These techniques are shown to be reliable and time-tested for deriving intrinsic and extrinsic parameters of the camera setup to generate the relationship between the coordinates of 2D captured image and the 3D world-coordinates. However, they have the disadvantages of complex algorithms with iterative procedures requiring significant computation resources that limit their application scenarios. They also are inefficient for real-time applications and require high communication bandwidth and high-power consumption.

4.4 Summary

In this chapter, the pin-hole camera model was presented with specific terminologies in addition to the introduction on parallel projection and perspective projection. Current camera calibration methods used for traffic application were reviewed including both conventional and self-calibration methods.

A camera calibration method that does not require specific patterns or iteration procedures with less complexity for traffic application is necessary.

.

CHAPTER 5　– PIXEL-RESOLUTION MAPPING (P-RM) METHOD

This chapter presents pixel-resolution mapping (P-RM) method. First, modified camera model is illustrated. Second, the relationship (pixel-resolution model) between pixel's physical size and pixel's actual spatial resolution is derived. Third, typical image processing algorithms including image enhancement, foreground/background segmentation and edge detection are reviewed followed by feature extraction algorithms including location detection, speed detection, and dimension detection based on P-RM method. Finally, flexible camera calibration method without specific pattern is discussed.

5.1　*Modified Camera Model, Pixel-Resolution Model*

A basic camera model was shown in Figure 4-6, which is based on Fung's model [13]. The camera is mounted at certain height, which is the perpendicular distance from the center of the camera lens to the X-Y plane, and looking down with certain tilt angle, ϕ, pan angle, θ, swing angle, ψ, and focal length, f. Pan-tilt cameras, which have ability to rotate around X_c-axis for tilting and Z_c-axis for panning with fixed swing angle 0 degree, are commonly used in road/traffic detection and surveillance [25], [28]. Consequently, the proposed P-RM method is designed by considering swing angle equaling 0 degree, and the center of the imager is on the optical axis. According to these preconditions, a modified camera model, pixel-resolution (P-R) model, including pixel array is transformed from Figure 4-6 to Figure 5-1. The world coordinates is translated in Figure 5-1, whose origin point is at the projection point of the optical center on the world plane and the Y_w-axis is overlapping on the projection of optical axis on the world plane. The optical center, O_c, at this world coordinates is *(0,0,H)* where *H* is the height of the camera.

In Figure 5-1, P-R model is showing that the detection region of a camera is fixed as the size of image sensor, focal length of the lens and mounting position are known. The labeled points A and B are maximum detection distance, dy_{max}, and minimum detection distance, dy_{min}, respectively, which give the detection range for the imager. Image sensor is usually 2D

rectangular shape composed of number of square size pixels (number of pixels in row, M, times number of pixels in column, N) that are specifically designed and fabricated by IC industry [57]. Because of the tilted sensor plane in the camera, each pixel experiences its own (spatial) resolution on both x and y directions. The resolution of pixels may vary row by row or column by column, which are closely depend on mounting orientations, lens types, and pixels' size. Indeed, the parameters for traffic monitoring such as location, size and speed of vehicles, could be simply extracted from the captured image if each pixels' resolution is known. Furthermore, the connection between an image captured by a mounted camera and real-world scenery could be explained as the pixels relationship with pixels' actual resolution in scenery/FOV is known. As a result, the conventional camera calibration problem could be transferred to the problem on determining pixel's resolution, which initiated the development of the P-R method.
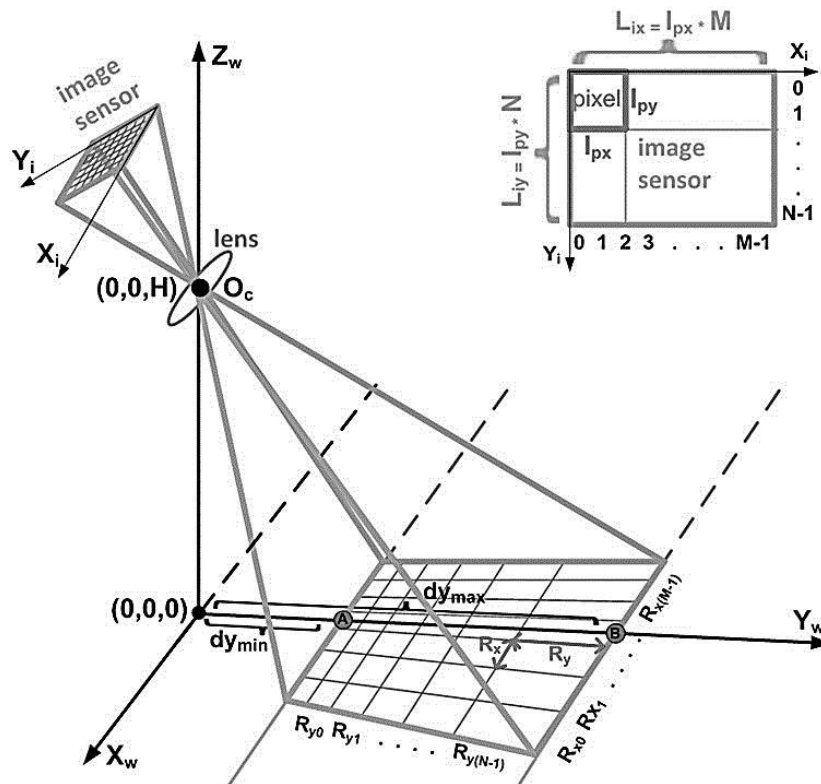
Figure 5-1. P-R model for TZOID sensor designing and P-RM method building.

5.2 P-RM Method

The proposed pixel-resolution mapping (P-RM) method, which could reconstruct image for mensuration in traffic monitoring application, is driven by identifying each pixel's resolution on the image plane based on P-R model. In other words, the actual size of the 3D world image could be recovered by each pixel's 2D resolution. Diagrammatic procedure of P-RM method is shown in Figure 5-2, which intuitively illustrates the concept of this method.

The camera has a field of view (FOV) in the scene which includes the object in gray color. The object in the FOV is captured by the camera, the pixel size of which is usually constant in modern commercial image sensor, generating a square shape raw image as shown in the middle of Figure 5-2. However, the spatial resolution of each pixel is varying depending on related detection distance. As a result, in the 2D raw image, the shape or the size of the captured object is usually distorted comparing the one in the real 3D world. The distortion caused by 3D perspective projection in the original 3D world image could be restored by using the spatial resolution of each pixel instead of pixel's physical size so that the reconstructed 2D image would have actual dimensional scale of 3D world. Consequently, it is important and essential to know each pixel's resolutions for realizing P-RM method.
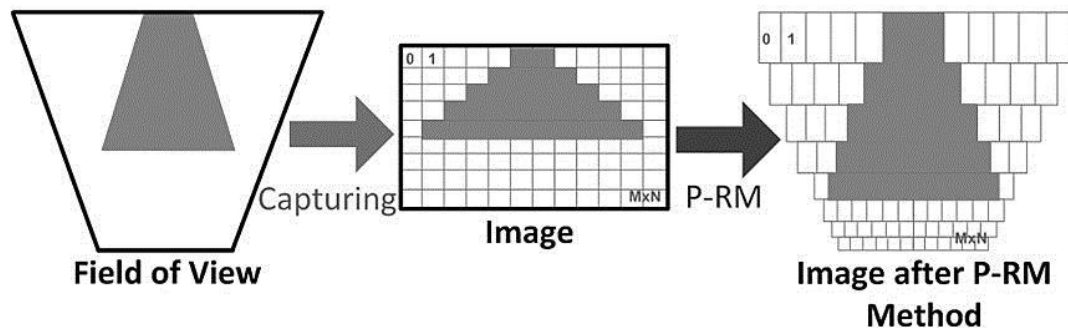
Figure 5-2. Diagrammatic procedure to reconstruct an image by P-RM method.

5.2.1 **Pixel Resolution Determination**

The resolution of each pixel consists of two parts; resolution on row *(X_i)* direction and resolution on column *(Y_i)* direction, which are marked as R_x and R_y, respectively. Before continuing on explaining the R_x and R_y as shown in Figure 5-3 and Figure 5-4, following

conditions are assumed. The camera is mounted at height, H, and tilted with ϕ. The camera lens has focal length of f. The pixel size is described as l_{py} and l_{px} on column direction and row direction, respectively. The size of image sensor is L_{ix} by L_{iy}, which is the total length of a row multiplied by the total width of a column. Total number of pixels is N by M including N rows and M columns. The projection of optical center toward image sensor plane is assumed at the geometric center of pixels arrays, and the distortion from the lens is ignored.

5.2.1.1   *Pixel Resolution in Column Direction, R_y*

Side view geometry of modified camera model is shown in Figure 5-3, which illustrates the model of pixel resolution in Y_i direction, R_y, and achievable maximum and minimum detection distances.
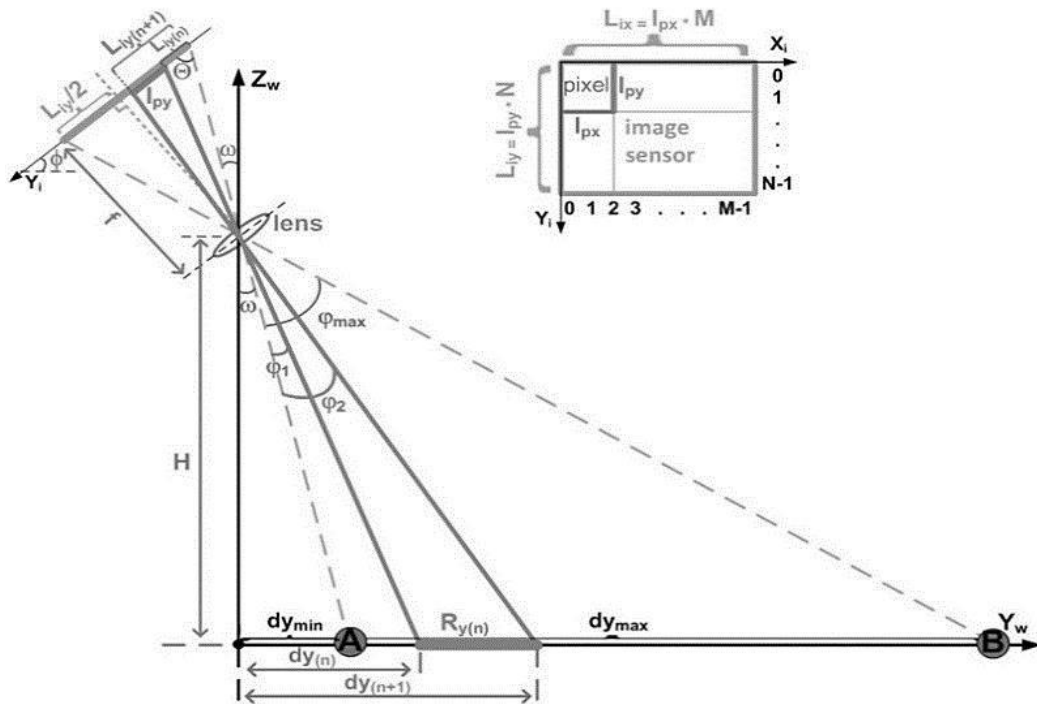
Figure 5-3. Side view of P-R model for R_y determination.

Point A is the closest, and point B is the furthest side the senor could view. Pixels at the bottom of the pixel array *(Row (M-1))* observe the furthest area (reaching up to point B), dy_{max}, the sensor could detect, while the closest region (point A), dy_{min}, could be detected by

the pixels at top of the array *(Row (0))*. The distance between farthest and closest detection line is the detection range, dyrange, which could be determined by (5.1).

$$dy_{range} = dy_{max} - dy_{min} \tag{5.1}$$

In (5.1), *dymax* and *dymin* could be acquired based on trigonometric function as shown in (5.2) and (5.3), respectively, where ω is depending on the sensor size and focus length, which could be written as (5.4).

$$dy_{max} = \tan(\omega + \varphi_{max}) \cdot H \tag{5.2}$$

$$dy_{min} = \tan(\omega) \cdot H \tag{5.3}$$

$$\omega = \Phi - \arctan(\frac{L_{iy}/2}{f}) \tag{5.4}$$

$$L_{iy} = l_{py} \cdot N \tag{5.5}$$

By substituting (5.4) into (5.2) and (5.3), equations (5.5), (5.6), and (5.7) for *dymax*, *dymin*, and *dyrange* could be derived, respectively.

$$dy_{max} = \tan[\Phi + \arctan(\frac{l_{py} \cdot N}{2f})] \cdot H \tag{5.6}$$

$$dy_{min} = \tan[\Phi - \arctan(\frac{l_{py} \cdot N}{2f})] \cdot H \tag{5.7}$$

$$dy_{range} = \{\tan[\Phi + \arctan(\frac{l_{py} \cdot N}{2f})] - \tan[\Phi - \arctan(\frac{l_{py} \cdot N}{2f})]\} \cdot H \tag{5.8}$$

The resolution in Y_i direction of a pixel at *Row(n)*, $R_{y(n)}$, could be determined by using similar approach, but instead of pixels at bottom and top array, bottom and top edge of each pixel is used. The parameter L_{iy} in (5.4) is total length of a pixel array, which equals each pixel's length times total number of rows for consist pixel size array ($L_{iy}=l_{py}*N$). In (5.2), φ_{max} could be depicted by (5.11) after substituting (5.10) into, (5.8) and (5.9).

$$\frac{\sin(\phi_{max})}{L_{iy}} = \frac{\sin(\Theta)}{\sqrt{(\frac{L_{iy}}{2}-L_{iy})^2 + f^2}} \tag{5.9}$$

$$\sin(\varphi_{max}) = L_{iy} \cdot \frac{\sin(\Theta)}{\sqrt{(\frac{L_{iy}}{2} - L_{iy})^2 + f^2}} \tag{5.10}$$

$$\Theta = \arctan(\frac{f}{2L_{iy}}) \tag{5.11}$$

$$dy_{max} = \tan[\Phi + \arctan(\frac{l_{py} \cdot N}{2f})] \cdot H \tag{5.12}$$

The resolution in Y_i direction of a pixel at *Row(n)*, $R_{y(n)}$, could be achieved using similar approach, but instead of pixels at bottom and top array, bottom and top edge of each pixel is used, which introduce (5.13). In (5.13), $dy_{(n)}$ and $dy_{(n+1)}$ could be achieved by (5.14) and (5.15), respectively.

$$R_{y(n)} = dy_{(n+1)} - dy_{(n)} \tag{5.13}$$

$$dy_{(n)} = \tan\{\omega + \arcsin[\frac{L_{iy(n)}}{\sqrt{(\frac{L_{iy}}{2} - L_{iy(n)})^2 + f^2}} \cdot \sin(\Theta)]\} \cdot H \tag{5.14}$$

$$dy_{(n+1)} = \tan\{\omega + \arcsin[\frac{L_{iy(n+1)}}{\sqrt{(\frac{L_{iy}}{2} - L_{iy(n+1)})^2 + f^2}} \cdot \sin(\Theta)]\} \cdot H \qquad (5.15)$$

In (5.14) and (5.15), $L_{iy(n)}$ and $L_{iy(n+1)}$, are the length from current pixel's top edge to top edge of the pixel array, and length from current pixel's bottom edge to top edge of the pixel array bottom, respectively. For same size pixel array, $L_{iy(n)}=(n)*l_{py}$ and, $L_{iy(n+1)}=(n+1)*l_{py}$. Consequently, the equation for $R_{y(n)}$ could be obtained by substituting (5.14) and (5.15), into (5.13), where $\Theta= arctan(f/0.5L_{iy})$. Therefore, $R_{y(n)}$ would be modeled by (5.16).

$$R_{y(n)} = \{\tan[\Phi - \arctan(\frac{l_{py} \cdot N}{2f}) + \arcsin(\frac{(n+1) \cdot l_{py}}{\sqrt{(\frac{l_{py} \cdot N}{2} - (n+1) \cdot l_{py})^2 + f^2}} \cdot \sin(\arctan(\frac{2f}{l_{py} \cdot N})))]$$

$$- \tan[\Phi - \arctan(\frac{l_{py} \cdot N}{2f}) + \arcsin(\frac{n \cdot l_{py}}{\sqrt{(\frac{l_{py} \cdot N}{2} - n \cdot l_{py})^2 + f^2}} \cdot \sin(\arctan(\frac{2f}{l_{py} \cdot N})))]\} \cdot H$$

$$(n = 0, 1, 2, \ldots, N\text{-}1)$$

$$(5.16)$$

5.2.1.2 *Pixel Resolution in Row Direction, R_x*

Geometry for getting the resolution in X_i direction, R_x, which is considered as $R_{x(n)}$ for the pixels at *Row(n)*, is shown in Figure 5-4. In the figure, there are two triangles, ΔabO_c and ΔABO_c, which are similar triangles. R_x could be determined by these two triangles, which give the basic equation for deriving $R_{x(n)}$ as shown in (5.17).

$$R_{x(n)} = dy'_{(n+\frac{1}{2})} \frac{l_{px}}{f'_{(n+\frac{1}{2})}} \qquad (5.17)$$

In (5.17), $dy'_{(n+1/2)}$ and $f'_{(n+1/2)}$ could be determined by (5.18) and (5.19), respectively.

$$dy_{(n+\frac{1}{2})}{}' = \sqrt{H^2 + (dy_{(n+\frac{1}{2})})^2} \tag{5.18}$$

$$f'_{(n+\frac{1}{2})} = \sqrt{f^2 + (\frac{L_{iy}}{2} - L_{iy(n+\frac{1}{2})})^2} = \sqrt{f^2 + (\frac{L_{iy}}{2} - (n+\frac{1}{2})\cdot l_{py})^2} \tag{5.19}$$

By substituting (5.18) and (5.19) into (5.17), $R_{x(n)}$ could be eventually obtained as shown in (5.20).

$$R_{x(n)} =$$

$$H \cdot \sqrt{1 + \{\tan[\Phi - \arctan(\frac{l_{py}\cdot N}{2f}) + \arcsin(\frac{(n+\frac{1}{2})\cdot l_{py}}{\sqrt{(\frac{l_{py}\cdot N}{2} - (n+\frac{1}{2})\cdot l_{py})^2 + f^2}} \cdot \sin(\arctan(\frac{2f}{l_{py}\cdot N})))]\}^2}$$

$$\cdot \frac{l_{px}}{\sqrt{f^2 + (\frac{L_{iy}}{2} - (n+\frac{1}{2})\cdot l_{py})^2}}$$

$$\tag{5.20}$$

Equations (5.16) and (5.20) give the general expressions of a pixel's (spatial) resolutions referred to ground plane in the 3D world, which could be determined if N, l_{py}, l_{px}, f, ϕ, and H are known. First three parameters, N, l_{py}, and l_{px}, could be obtained from commercial image sensor's datasheet that includes size and number of pixels in the sensor. Focal length, f, could also be determined from specifications of commercial off-the-shelf lens.

For instance, TZOID imager was designed and the camera system was used a lens that has $f=4.42mm$. H and ϕ are depending on mounting situation of the camera, which are not difficult to be measured by laser altimeter and digital clinometer. However, camera systems used for traffic application are typically mounted on a traffic light pole at 8.84 m (29ft) above the ground [95]. Further, the tilt angle or mounting height could be calibrated by P-RM method easily in case $\phi\, or\, H$ is not available.
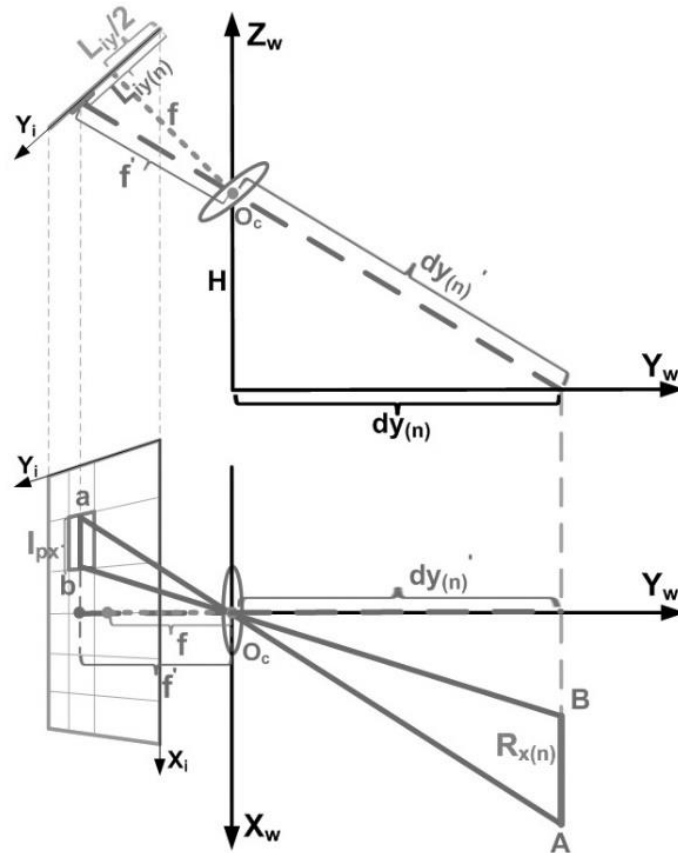
Figure 5-4. Geometry for modeling pixel resolution in X_i direction.

5.3   *Feature Extraction*

The 2D size could be determined straightforward as each pixel's resolution is achieved after edge detection. Consequently, the location, size, and speed of vehicles could be accessed. The height estimation by P-RM method could also achieved with certain assumption.

5.3.1   **2D Length Detection**

The length measuring in X_i and Y_i direction would be straightforward as knowing the pixel's resolutions, R_X and R_Y, after running edge detection algorithm. As shown in Figure 5-5, the lengths of L_1 and L_2 are R_x multiplying number of pixel of L_1, and R_y multiplying by number of pixel of L_2, respectively. The edge length that is not exactly on the X_i or Y_i direction could be calculated by trigonometric function, such as L_3 in Figure 5-5.
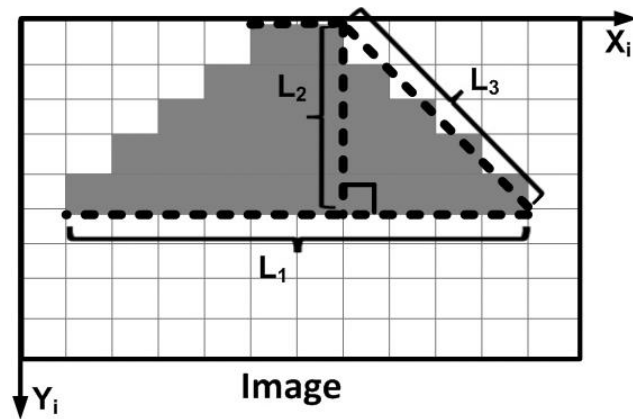
Figure 5-5. Schematic diagram for length measuring.
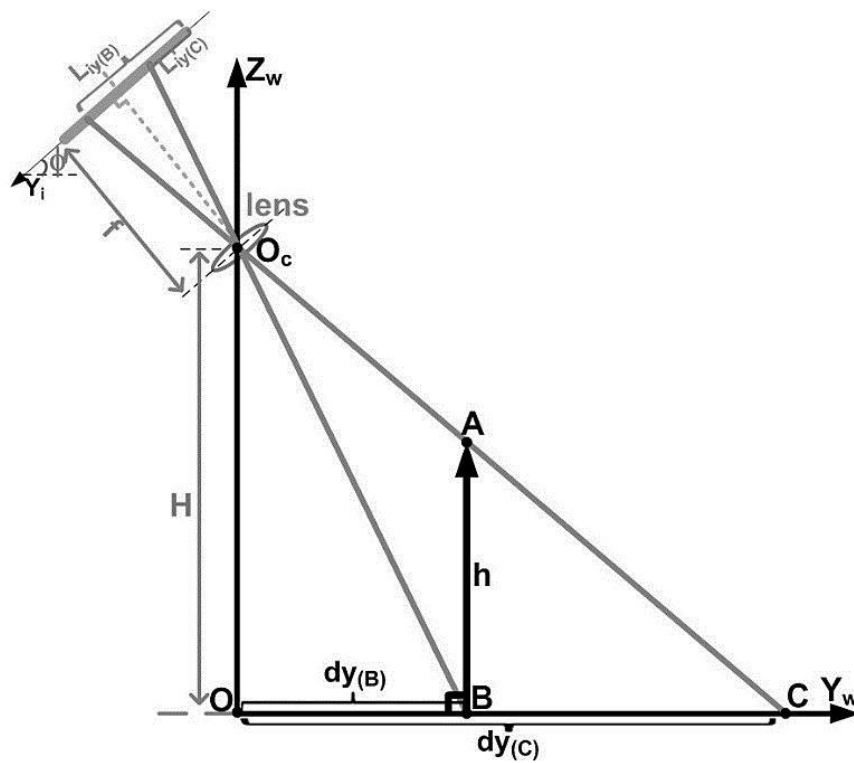
Figure 5-6. Geometry for height estimation, h.

5.3.2 **3D Height Estimation**

In this section, a height estimation method based on PR-M method of single-view image is presented, which is an alternative way of reconstructing the scene without the

disadvantages of the existing methods. It is assuming that the potential target object such as human, is either walking or standing on the ground plane, and is perpendicular to the ground. Additionally, three parameters have to be known for the method to work; size and number of pixels in the image sensor, and the focal length of lens used in the camera system.

The height estimation method is based on pixel's spatial resolution in column direction, R_y. The location of object could be determined first by using (5.14) after detecting object's bottom edge, point B. The top edge of the object, point A, needs be detected as well. As a result of the precondition that assumes object is perpendicular to the ground plane, the geometry for object height analysis is shown in Figure 5-6. Geometry for height estimation, h. including two similar triangles, ΔCOO_c and ΔABO, which introduces following equation, (5.21) on height, h, extraction.

$$
\begin{aligned}
h &= H \cdot (1 - \frac{dy_{(B)}}{dy_{(C)}}) \\
&= H \cdot (1 - \frac{\tan\{\omega + \arcsin[\dfrac{L_{iy(B)}}{\sqrt{(\dfrac{L_{iy}}{2} - L_{iy(B)})^2 + f^2}} \cdot \sin(\Theta)]\}}{\tan\{\omega + \arcsin[\dfrac{L_{iy(C)}}{\sqrt{(\dfrac{L_{iy}}{2} - L_{iy(C)})^2 + f^2}} \cdot \sin(\Theta)]\}})
\end{aligned}
\tag{5.21}
$$

The size of object could be calculated directly using proposed model and equations if N, l_{py}, l_{px}, f, ϕ, and H are known. First three parameters, N, l_{py}, and l_{px}, could be obtained from commercial image sensor's datasheet that includes size and number of pixels in the sensor. Focal length, f, could also be determined from specifications of commercial off-the-shelf lens. H and ϕ are depending on mounting situation of the camera, which are not difficult to be measured by laser altimeter and digital clinometer. Further, the tilt angle and height could be calibrated by P-RM method easily in case ϕ or H is not available.

## *5.4* *Flexible Camera Calibration by P-RM Method*

The camera calibration methods described in Chapter 4 need at least three images for self-calibration, and special or specific patterns in 3D world for conventional camera calibration methods. These techniques are shown to be reliable and time-tested for deriving intrinsic and extrinsic parameters of the camera setup. However, they have the disadvantages of complex algorithm with iterative procedures requiring significant computation resources that limit their application scenarios. They are also inefficient for real-time applications and require high communication bandwidth.

To avoid the problems, a new pixel-resolution (spatial resolution) mapping (P-RM) method is proposed that would work with both standard rectangular and proposed TZOID image sensors. The proposed PR-M is an alternative and efficient way to generate the relationship between 2D image and 3D world which is free of the disadvantages of the existing methods. Table 5-1. Comparison of camera calibration methods. summaries the main features of proposed P-RM method comparing with conventional and self-calibration methods.

Table 5-1. Comparison of camera calibration methods.

	Conventional	**Self-calibration**	**Proposed P-RM**
Specific Patterns	Required	Not Required	Not Required
Moving Camera	Not Required	Required	Not Required
Iteration Procedures	Required	Required	Not Required
Complexity	Medium	High	Low

Three parameters must be known for the P-RM method to work; size and number of pixels in the image sensor, and the focal length of lens used in the camera system. Using these parameters, proposed P-RM method finds the relationship between sensor's 2D pixels and 3D world pixels' resolution directly. It provides straight forward features extraction without using iterative procedures, thus it is computationally less complex. As a result, the

camera system using P-RM method for traffic application does not need large computation capability, or high communication bandwidth, or multiple spatio-temporal image capture, or 3D world patterns to work properly.

5.4.1 **Tilt Angle Calibration**

The tilt angle, ϕ, might be difficult to determine as it could change when the environmental parameters are changed such as time of day, temperature, vibration due to wind, etc. Thus, it is necessary to get the parameter, ϕ, calibrated periodically. The proposed P-RM method achieves this simply by capturing two images without requiring special patterns or specific correction or calibration blocks.

5.4.1.1 *Applying on Vehicle Detection Application*

The best calibration target is the moving vehicle itself in a traffic monitoring application. The length of a moving vehicle will not change as its location is changed. For instance, the white vehicle in Figure 5-7 could be used as a calibration target, the length of which, l, could be measured by finding the distance between the leading and rear edges of the vehicle using (5.22), which is similar to determining the pixel resolution described before. As a result, the expression of l could be written as shown in (5.23)

$$l = dy_{(leadingEdge)} - dy_{(rearEdge)} \tag{5.22}$$

$$
\begin{aligned}
l = \{ &\tan[\Phi - \arctan(\frac{l_{py} \cdot N}{2f}) + \arcsin(\frac{n_{leadingEdge} \cdot l_{py}}{\sqrt{(\frac{l_{py} \cdot N}{2} - n_{leadingEdge} \cdot l_{py})^2 + f^2}} \cdot \sin(\arctan(\frac{2f}{l_{py} \cdot N})))] \\
&- \tan[\Phi - \arctan(\frac{l_{py} \cdot N}{2f}) + \arcsin(\frac{n_{rearEdge} \cdot l_{py}}{\sqrt{(\frac{l_{py} \cdot N}{2} - n_{rearEdge} \cdot l_{py})^2 + f^2}} \cdot \sin(\arctan(\frac{2f}{l_{py} \cdot N})))] \} \cdot H
\end{aligned}
$$

$$\tag{5.23}$$
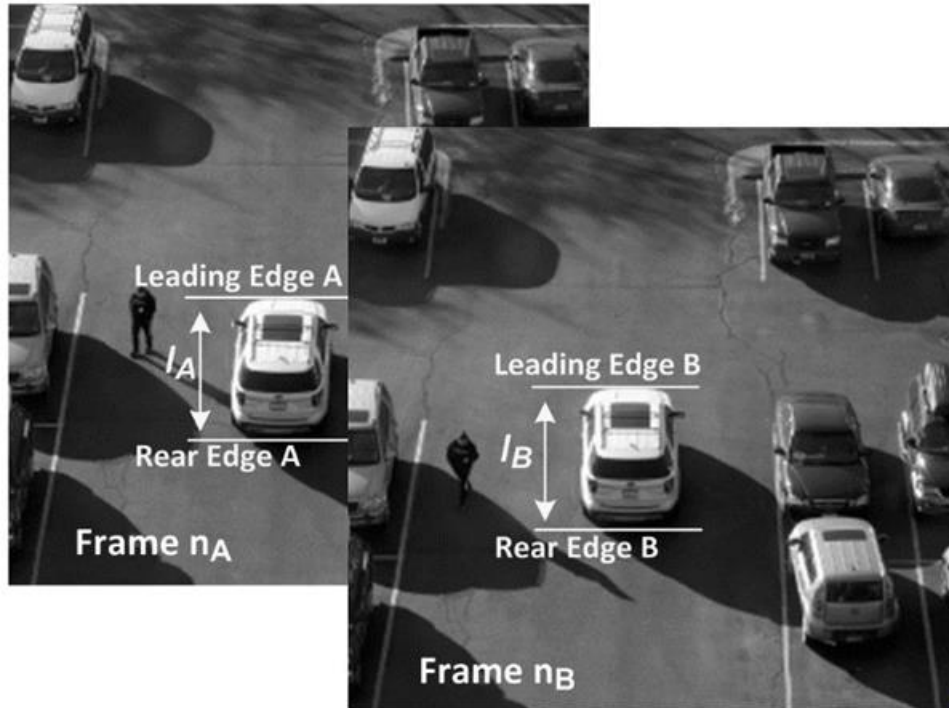
Figure 5-7. Two images capturing a moving vehicle for tilting angle calibration.

Consequently, l_A and l_B, in frame n_A and frame n_B could be determined by (5.23), respectively. Then, an equation with one unknown variable, $\phi\ (0^o<\phi<90^o)$, could be solved after setting l_A equaling to l_B so that the tilt angle could be computed.

5.4.1.2 *Applying on Pedestrian Detection Application*

The best calibration target is a person himself or herself in pedestrian monitoring application. The height of a moving person will not change as its location is changed. For instance, a person in white T-shirt shown in Figure 5-8 could be used as a calibration target, the height of him, h, could be measured by (5.24) based on (5.21).
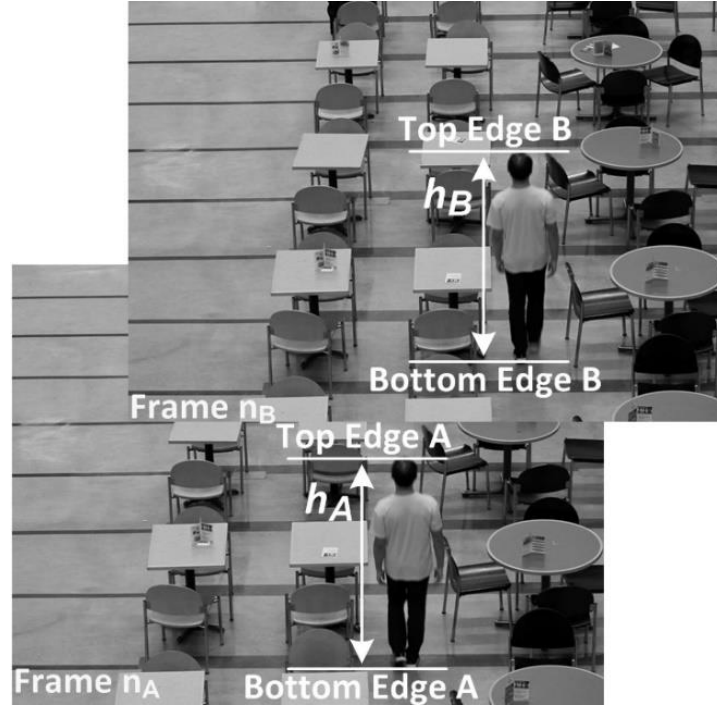
Figure 5-8. Two images capturing a walking person for tilting angle calibration.

$$h = H \cdot (1 - \frac{dy_{(BottomEdge)}}{dy_{(TopEdge)}}) \qquad (5.24)$$

In (5.24), $dy_{(BottomEdge)}$ and $dy_{(TopEdge)}$ could be achieved by (5.14), then h_A and h_B in terms of H and ϕ, in frame n_A and frame n_B could be depicted by (5.24), respectively. Then, a trigonometric equation with one unknown variable, ϕ, could be solved with a solution set after setting l_A equaling to l_B eliminating H so that the tilt angle could be calibrated, which is the one in the range between 0 and 90 degree.

5.4.2 Mounting Height Calibration

The mounting height could be achieved easily by a known length or height object as reference in the field of view. For instance, a human whose height is known could be used as reference target for mounting height calibration, which is quite common in pedestrian height estimation application. The mounting height, H, could be derived by (5.25), in which h_{ref} is the height of reference object.

$$H = h_{ref} \cdot (\frac{dy_{(TopEdge)}}{dy_{(TopEdge)} - dy_{(BottomEdge)}}) , \qquad (5.25)$$

5.5   Summary

In this chapter, the modified camera model was presented which was the fundamental for P-RM method. The relationship between pixel's physical size and pixel's actual spatial resolution was illustrated with details on derivations. Typical image processing algorithms including background subtraction, shadow correction and edge detection were introduced followed by feature extraction algorithms including 2D length detection and 3D height estimation based on P-R model. Finally, flexible camera calibration method focusing on tilt angle and mounting height calibration without specific pattern was discussed.

CHAPTER 6　– TRAPEZOID (TZOID) IMAGE SENSOR

This chapter presents design details of the trapezoid (TZOID) image sensor. First, the introduction and background of TZOID imager is presented. Second, pixel design equations based on P-R model and details of TZOID pixel array development are discussed. Third, the image sensor architecture and details of analog and digital building blocks are illustrated. Finally, the actual TZOID chip with a custom TZOID imager-based camera system is shown.

6.1　Introduction and Background of TZOID Image Sensor

If the cameras are used in real-time traffic detection, they have no or few feature extraction capabilities, resulting in producing large amount of unimportant information, which will increase both communication bandwidth and system power consumption. As a result, more powerful computation capability is required that necessitates high performance platforms. Consequently, it will be difficult to meet low-power, low-cost and light-weight requirements of next generation traffic monitoring systems.

Several methods have been proposed and utilized for image sensors to eliminate irrelevant image data and alleviate the burden of image processing and calculation power in traffic monitoring platforms. The methods include; providing direct frame difference output, utilizing multi-resolution image sensors, and on-chip clustering algorithms, [15]–[18].

A CMOS active pixel sensor (APS) with direct frame difference output was reported in [15]. This image sensor could just output the difference between current frame and previous one as shown in Figure 6-1. The photon sensitive element is a photo diode formed by N-diffusion and P-substrate. The proposed APS pixel circuit has two sample and hold circuits and two signal readout paths for direct frame difference output. A sample and hold circuit consists of a sampling switch and a holding node. The sampling switches (M2, M3) act as electronic shutters to control the exposure time of the pixel. Two MOS capacitors (M8, M9) are added to increase the capacitances at the holding nodes of the source follower transistors (M4, M6). The chip is suitable for motion detection and only sending out moving
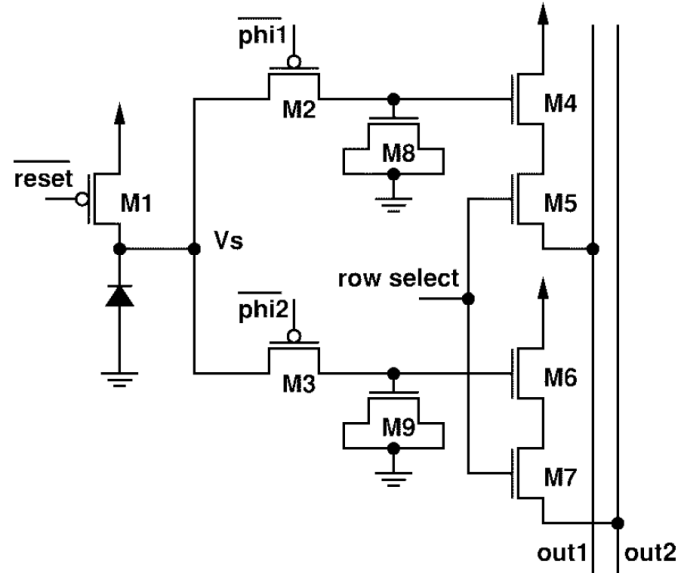
Figure 6-1. APS cell circuit with direct frame difference output [15].

objects, but it still needs traditional complex mapping algorithm for speed and location detection in traffic application.

A spatio-temporal multi-resolution CMOS image sensor was reported in [17]. Block diagram of the imager is shown in Figure 6-2. This image sensor can simultaneously generate two outputs: one at a low frame rate with maximum spatial resolution for stationary backgrounds and the other at a high frame rate with reduced spatial resolution for moving
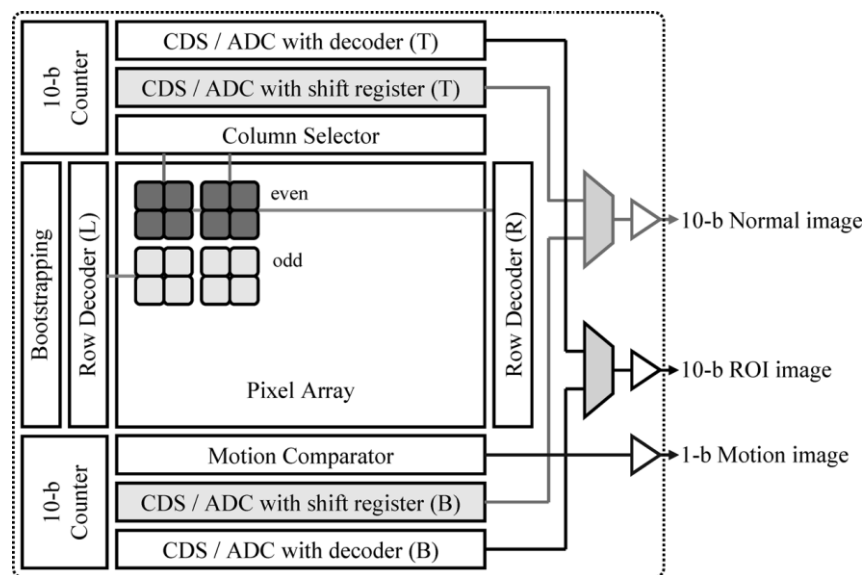
Figure 6-2. Block diagram of spatial-temporal multi-resolution image sensor [17].
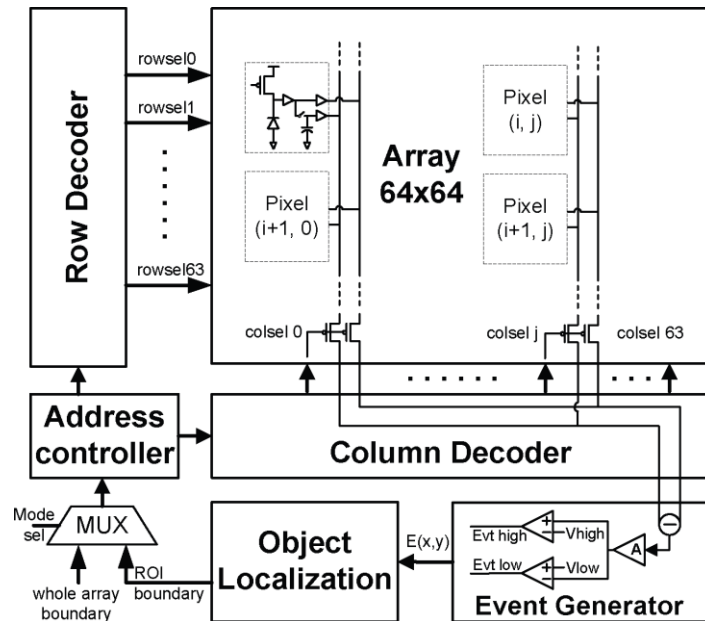
Figure 6-3. Block diagram of the proposed CMOS Image Sensor with on-chip event generator and object localization [18]

objects in the region-of-interest (ROI). However, the overall system requires two external controllers. Although it can automatically switch between modes quickly and captures the image of object in high or low resolution, complex coordinate mapping and calibration is still needed for traffic detection since the pixels of the sensor have the same physical size.

In [18], a CMOS image sensor with motion generator circuit used for on-chip moving object detection and localization is presented. Figure 6-3 shows the system architecture of the proposed image sensor. Each pixel is equipped with an analog memory (capacitor) and can output both the new integration voltage on its photodiode and the previous voltage stored on its capacitor. The event generator computes the difference between the two voltages and compares it to a positive and negative threshold. A motion event is generated if this difference exceeds the thresholds. If the scene illumination and object reflectance are constant, the changes in scene reflectance only result from object movements or camera motion. The motion events are processed on the fly to build clusters based on a distance criterion. At the end of the frame, the position and size of the active object is obtained. The algorithm was implemented on-chip without the need for any external computation and storage. This image sensor performs frame differencing to

generate binary motion events, which are processed on the fly to build clusters based on a distance criterion. After the position and size of the active object is obtained, the sensor switches to ROI intensity mode and reports a picture of the object. This approach eliminates the background and static object information, generating outputs from pixel values that are the difference between two frames that are above a threshold value.

As a result, although these custom image sensors and image pre-processing methods used by them reduces the generated image data, they still require camera calibration algorithms for traffic monitoring applications.

To avoid the aforementioned problems, a custom designed multi-resolution CMOS image sensor with trapezoid array (TZOID) for extracting "only" the dynamic properties of incoming traffic and the facile features of vehicles is proposed. Designed custom TZOID imager with proposed P-RM method does not need large computation capability, or high communication bandwidth, or multiple spatio-temporal image capture, or 3D world patterns to work properly while the whole system could be built smaller and consumes significantly less power for traffic monitoring application nodes.

6.2   TZOID Image Sensor

Typically, the camera is placed high above the ground (higher than 12m) having a bird's eye view of the road in video-based traffic monitoring systems, [96]. Proposed Trapezoid (TZOID) CMOS image sensor with varying pixel sizes uses this fact to provide straightforward camera calibration. Pixel sizes in the array was designed assuming that it will be mounted on a traffic light pole (typically 29ft high and 24ft to 48ft away from the stop line) looking directly toward two (or four) lanes (24-48ft wide) of incoming traffic. The TZOID is mainly designed for traffic detection, for extracting "only" the dynamic properties of incoming traffic and the facile features of vehicles using straightforward mapping and extractions methods, [97].

Gathering information about vehicles at the far end of the FOV (vehicle C in Figure 6-4 (a)) requires high resolution pixels which are not needed for vehicles in the near region (vehicle A in Figure 6-4 (a)). Therefore, a pixel array with variable pixel sizes (Figure 6-4 (b)) covering trapezoidal FOV is proposed to obtain the location and speed of an incoming vehicle accurately and in real time with P-RM method. In this proposed TZOID image sensor, the

pixel sizes are diminishing gradually toward to the far end of the FOV. Such a structure provides higher resolution only for those parts of the FOV.
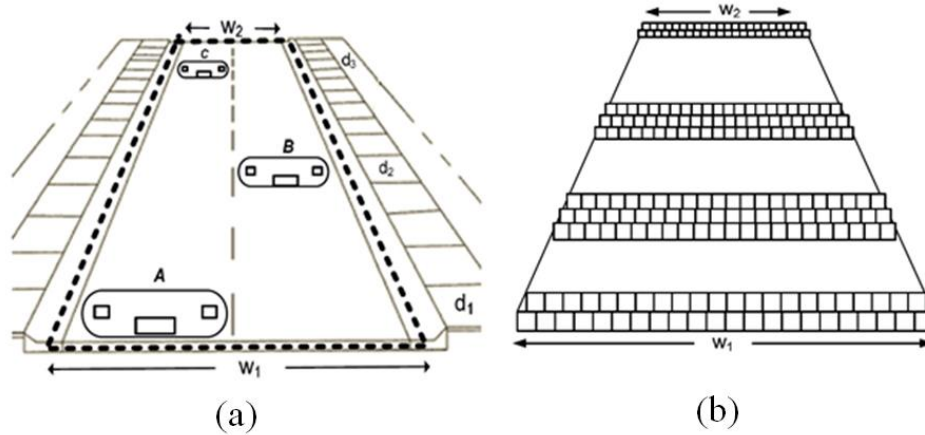
Figure 6-4. Schematic diagram of FOV and pixel array from TZOID imager: (a) two-lane incoming road to an intersection, b) trapezoid pixel array with scaled pixel sizes from large at the near end to small at the far end.

6.2.1 **TZOID Pixel (Array) Design**

The modified camera model, P-M model, in Figure 5-1 is also used for designing TZOID pixel array. The TZOID based camera is placed on a traffic pole of height H. The optical center is at coordinates *(0, H)* and has an effective focal length of f, with tilt angle of ϕ. The center of proposed sensor's pixel array is located at optical axis of the camera lens. In TZOID imager, pixel size depends on its desired pixel resolutions, R_x, according to assumed mounting situation and selected lens. Consequently, these parameters, H, ϕ and f are initially assigned (*H=8.84m*, ϕ *=24 degree,* and *f=4.42mm*) for TZOID pixel array design. The geometry for designing pixel size is the same as the geometry shown in Figure 5-3 and Figure 5-4 As a result, pixel size in column direction, l_{py}, and pixel size in row direction, l_{px}, can be calculated by (6.1) and (6.2) hich are derived from (5.16) and (5.20), respectively, where $dy_{(0)}$ = $dy_{(min)}$ as $n = 1$.

$$l_{py(n)} = \frac{f \cdot \sin\left[\tan^{-1}\left(\dfrac{H}{dy_{(n-1)}}\right) - \tan^{-1}\left(\dfrac{H}{dy_{(n-1)} + R_{y(n)}}\right)\right]}{\sin\left[\Phi + \tan^{-1}\left(\dfrac{H}{dy_{(n-1)} + R_{y(n)}}\right)\right] \cdot \sin\left[\Phi + \tan^{-1}\left(\dfrac{H}{dy_{(n-1)}}\right)\right]} \tag{6.1}$$

$$(n = 1, 2, \ldots, N\text{-}1)$$

$$l_{px(n)} = R_{x(n)} \cdot \sqrt{\frac{f^2 + (\frac{L_{iy}}{2} - (n + \frac{1}{2}) \cdot l_{py})^2}{H^2 + (dy_{(n+\frac{1}{2})})^2}} \tag{6.2}$$

$$(n = 1, 2, ? \quad N\text{-}1)$$

TZOID design composes of 152 rows of pixels with 6 different resolution sections (segment I to VI). Pixels in these sections have 0.25m, 0.5m, 1m, 2m, 5m, and 10m spatial resolutions, respectively. Figure 6-5 shows the pixel size, l_{py}, and desired R_y of each pixels from *Row(0)* to *Row(151)* as TZOID mounts at initially assumed conditions. The minimum pixel size is 3.4µm which is limited by selected CMOS process technology.
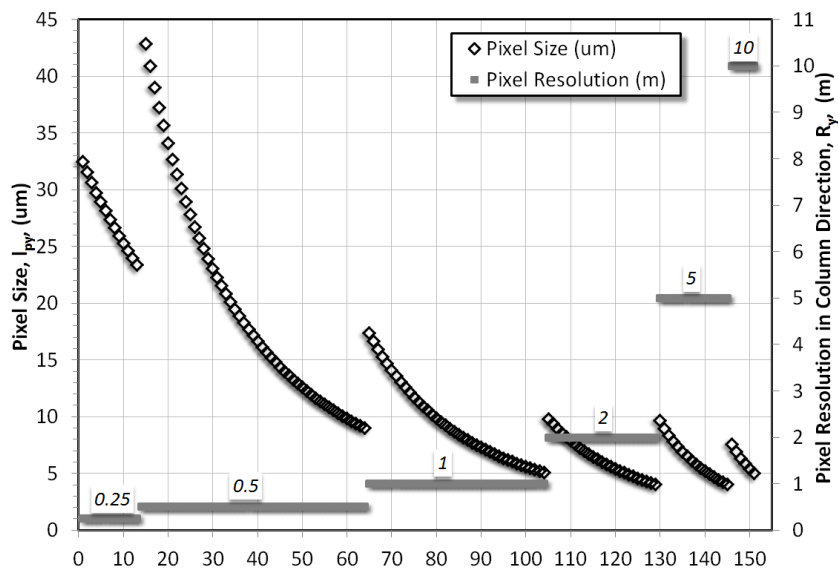
Figure 6-5. Desired pixel size and corresponding pixel resolution at initially assumed mounting situation (*H=8.84m*, ϕ *=24 degree,* and *f=4.42mm*).

Pixel sizes in column direction, l_{py}, are adjusted to fit into the layout design grid of the fabrication process. For example, design equation (6.1) results in a pixel height of 3.413µm but process grid allows only 3.40µm or 3.45µm pixel size causing measurement error due to rounding. After a manufacturable pixel size design and adjustment, distance measurement errors are calculated as shown in Figure 6-6.

Expected ideal measurement error is less than ±1.0% of the absolute distance resolution on the road. The design results in measurement error (on x-direction) of ±1cm for far end of the observation field where per pixel measurement distance is 10m. At closer end of the observation field (the stopping lane) this is 1mm with 0.25m resolution.
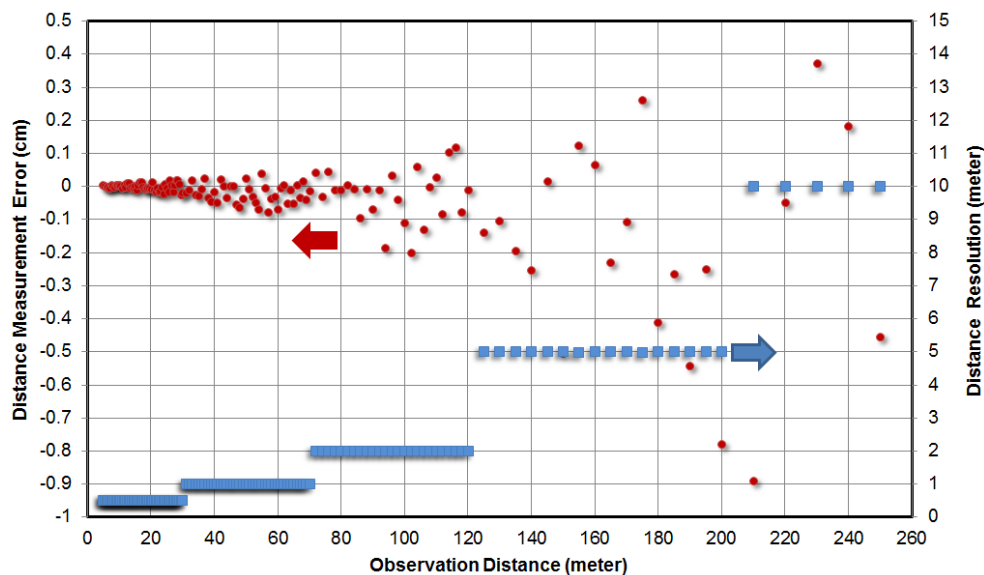
Figure 6-6. Expected ideal relative distance measurement error.

Pixel size in row direction, lpx, can be different than the pixel size in column direction, lpy , that would result in rectangular pixels instead of general square size pixel, which could be achieved by (6.2).

Figure 6-7 shows the pixel size in row direction, lpx, and desired Rx of each pixels from Row(0) to Row(151) in addition to the total width of each row. The pixel width and height in different rows are different, but the sizes of pixels in the same row are the same.

The spatial resolution per pixel in x-direction is set 0.21m for the design. This resulted in a maximum pixel width (l_{pxmax}) of 48.7µm. This limits the number of columns for

the imager. Based on the maximum pixel width, and assuming 10 columns per lane and maximum four lanes to be monitored, the number of imaging pixel columns is set to 40. Number of rows on the TZOID imager is set to 152. Of these pixels the first five (0-4) and last three rows (149-151) are covered for dark level measurement and correction.
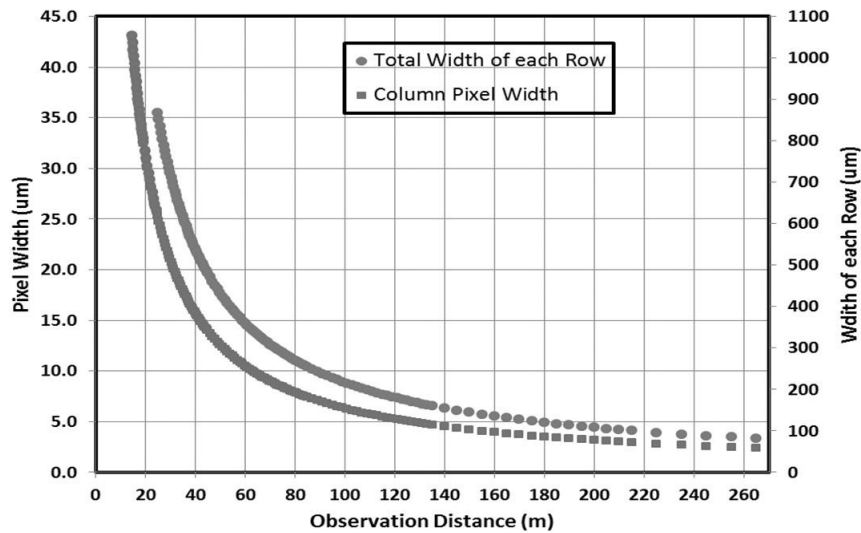
Figure 6-7. The pixel size in row direction and the total width of each row versus observation distance assumed mounting situation
($H=8.84m$, $\phi=24$ degree, and $f=4.42mm$).
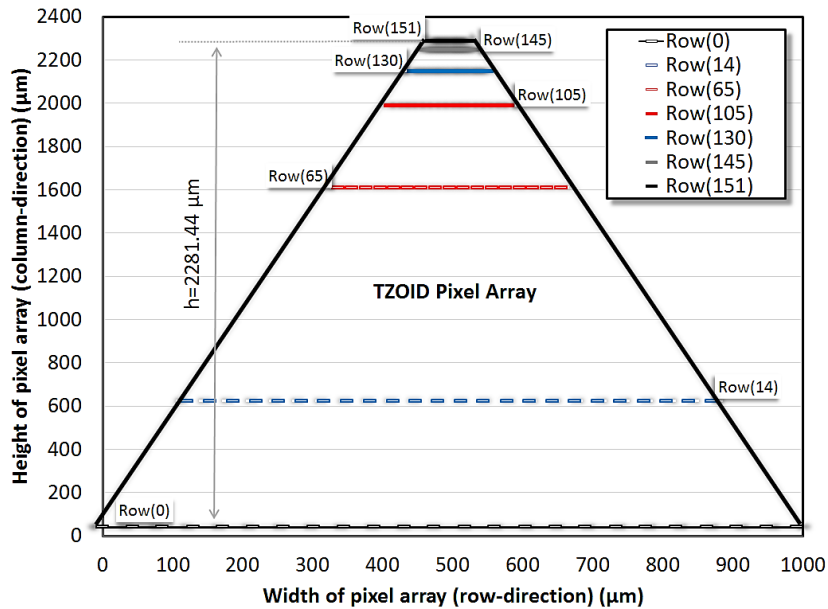
Figure 6-8. Geometric dimension of proposed TZOID pixel array.

The dimension of whole TZOID pixel array is shown in Figure 6-8, which is isosceles trapezoid shape with total 119.29mm^2 area.

6.3   TZOID Architecture

Figure 6-9 shows the block diagram of the TZOID imager comprising the APS pixel, column readout circuits and peripheral circuits.
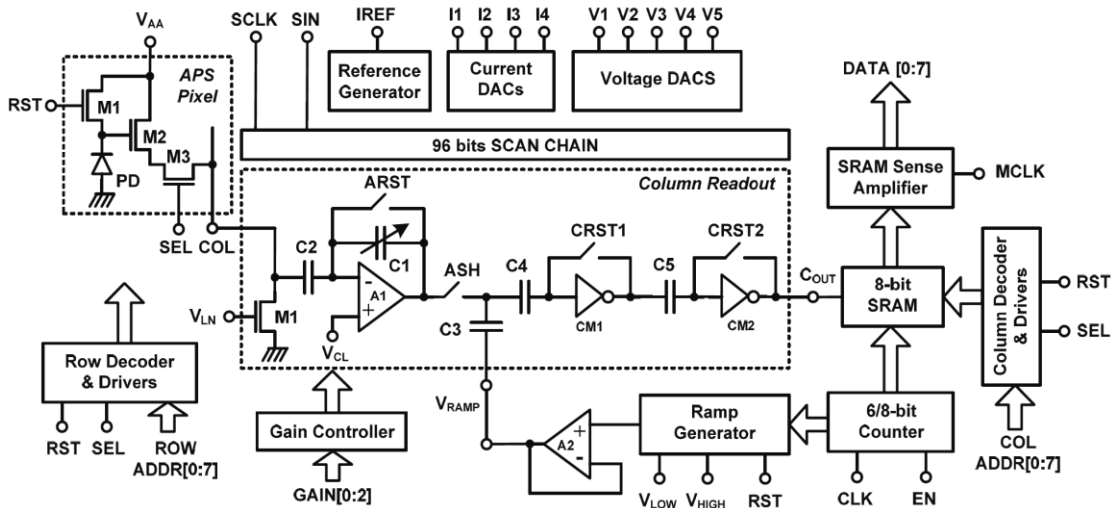
Figure 6-9. Circuit block diagram of TZOID imager.

Photodiode type, 3 transistor (3T) APS CMOS pixel was used in the design. The chip mainly comprises of the trapezoid pixel array, column readout circuits and peripheral circuits, which include pixel source follower load transistor, programmable gain charge amplifier, and auto-zero column comparators. Column readout circuit generates a latch signal to 8-bit transparent SRAM memories on each column[97].

6.3.1.1   Pixel Circuit

Standard CMOS APS pixel containing three metal-oxide-semiconductor field effect transistors (MOSFET) (M1-M3) in each pixel was used in the design as shown in Figure 28. Pixel size varies in the imager as shown in Figure 3-5, however, photosensitive element, photodiode (PD), sizes are kept constant in three regions. Associated photodiode capacitances for different pixel array rows are given in Figure 6-10. PD capacitance is 75fF

for rows 0 to 62, 25fF for rows 63 to 135, and 12.5fF for rows 136 to 151. They are binary weighted for easy application of gain during readout. As a result of scaled PD capacitance, higher sensitivity is achieved by pixels that monitor vehicles far away from the stop light while vehicles close by are monitored with less sensitivity.
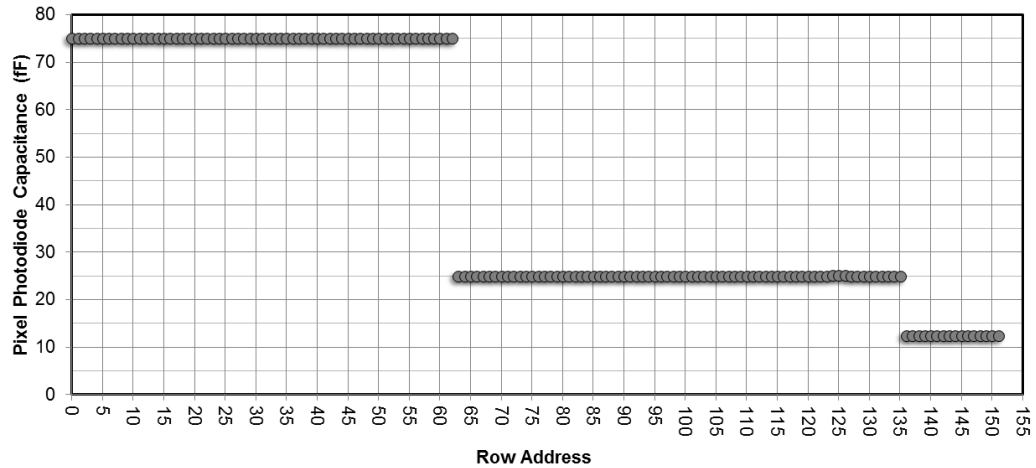
Figure 6-10. Pixel photodiode capacitances of the TZOID imager versus row address.

6.3.1.2   *Column Readout*

Column readout block composes three sub-blocks including programmable column charge amplifier, column comparators, and column latches and memory as shown in Figure 6-11.
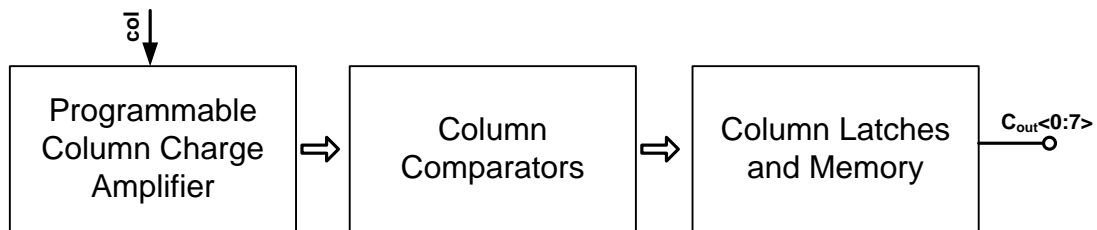
Figure 6-11. Block diagram of column readout.

Figure 6-9 show the schematic of this block which is mainly composed by pixel source follower load transistor (M1), programmable gain (C2/C1) charge amplifier (A1), and auto-zero column comparators (CM1, CM2).

Pixel photodiode signals are read through the pixel source follower with a gain of 0.75 V/V. The pixel signal and reset levels are subtracted performing correlated double sampling (CDS) and amplified by the charge amplifier. Gain of the charge amplifier is set with 3-bit binary control. Charge amplifier gain could be set between 0.67V/V and 32 V/V achieving pixel photodiode to chip output overall gain of between 0.5 V/V and 24 V/V. Charge amplifier output is sampled on a sample-and-hold capacitor (C3). Bottom plate of this capacitor is connected to global ramp generator output for column level ADC operation. Auto-zero two-stage comparator was used for the design. Offset voltages of the comparators are stored on two capacitors (C4, C5) and subtracted/cancelled during analog-to-digital conversion operation. Column readout circuit generates a latch signal (C_{out}) to 8-bit transparent SRAM memories also located on each column.

6.3.1.3   *Analog-to-Digital Converter (ADC) and Memory Blocks*

A single slope ramp signal is applied during digitization of the pixel signals sampled on each column. Triggering occurs when the ramp signal is equal to the sampled pixel signal allowing digital quantization of them. Global programmable (6-bit/8-bit) synchronous counter, binary weighted charge redistribution digital to analog converter (DAC) with output buffer is used in the ADC block. Output of the synchronous counter also drives transparent SRAM cells placed on each column. A global sense amplifier was designed to digitize the column SRAM signals and drive to the digital pad of the imager IC.

6.3.1.4   *Reference Generator and DACs*

A supply independent, programmable current reference circuit was designed and integrated in the TZOID imager as shown in Figure 39. Reference currents could set to 5µA, 10µA or 20µA, using 2-bit control. Default current setting is 5µA for the design. Current DACs use reference current by binary scale mirroring and have 6-bit programming control through scan chain. Similarly voltage DACs were also designed that could be controlled with 6-bit resolution. Four current and five voltage DACS were designed and integrated in the design.
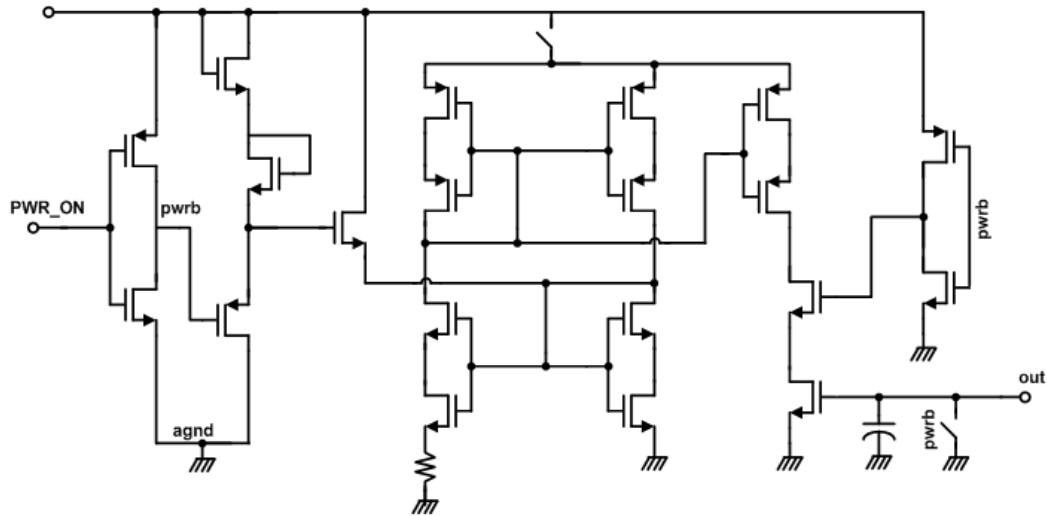
Figure 6-12. Circuit diagram of the programmable reference current generator.

6.4   Actual TZOID Imager

The TZOID imager blocks were simulated and their functionalities were verified before physical design and integration. Designed physical layout of the imager blocks were checked against the schematic and integrated into IC as shown in Figure 6-13. A 0.18μm CMOS process was used for the design and fabrication. Core supply voltage of the TZOID imager was set to 1.8V. Physical size of the imager die is 3.23mm x 2.9mm with 68 pads. A 68 pin Ceramic Leadless Chip Carrier (CLCC) package were chosen to package the fabricated imager die. Packaged TZOID imager is shown in Figure 6-14.

The packaged TZOID imager was soldered on custom designed camera board that holds FPGA development board, and various voltage regulators and digital buffers being configured as a camera system as shown in Figure 42 . The FPGA board on the camera board handles controlling communication with the PC, controlling the TZOID imager, and reading frames of images from the imager. Custom software that a user could set TZOID control registers and communicates with FPGA board was also written.

The proposed trapezoid pixel array is composed of 6080 (152 x 40) total pixels. The minimum pixel size is 3.4μm. Thus, if a rectangular array image sensor is chosen, pixel size has to be 3.4μm x 3.4μm to accommodate minimum pixel size in the trapezoid image sensor. Consequently, a rectangular image sensor with 335,205 (585x573) pixels array size

is required to function as the designed trapezoid sensor with 6080 pixels, which is 55 times larger compared to the trapezoid imager. Rectangular imager with 585x573 array size running at 120 FPS would generate 40.225Mbytes/sec video. However, it is only 0.730Mbytes/sec for TZOID imager which also requires 55 times lower communication bandwidth.
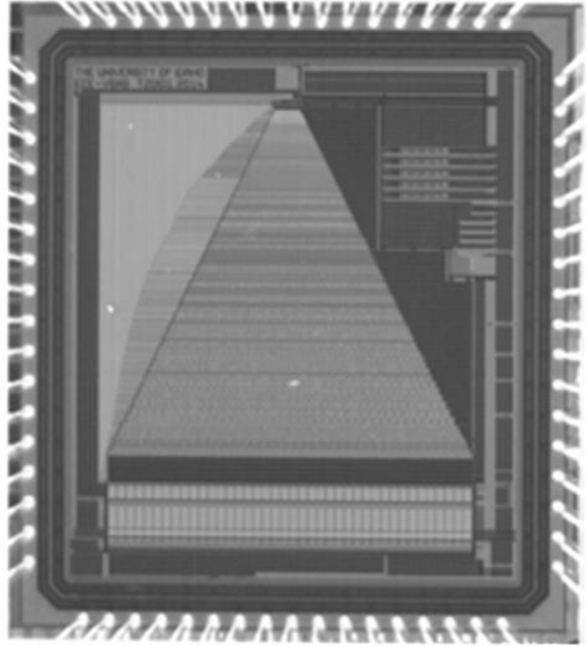
Figure 6-13. Micrograph of the fabricated and packaged TZOID imager.
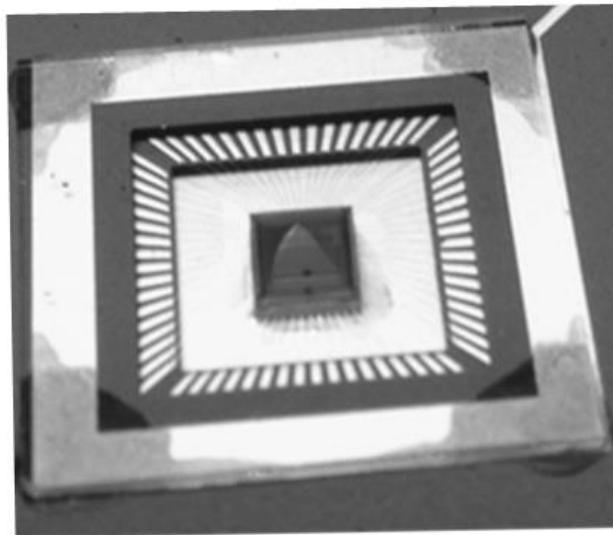
Figure 6-14. Pictures of the packaged TZOID imager.
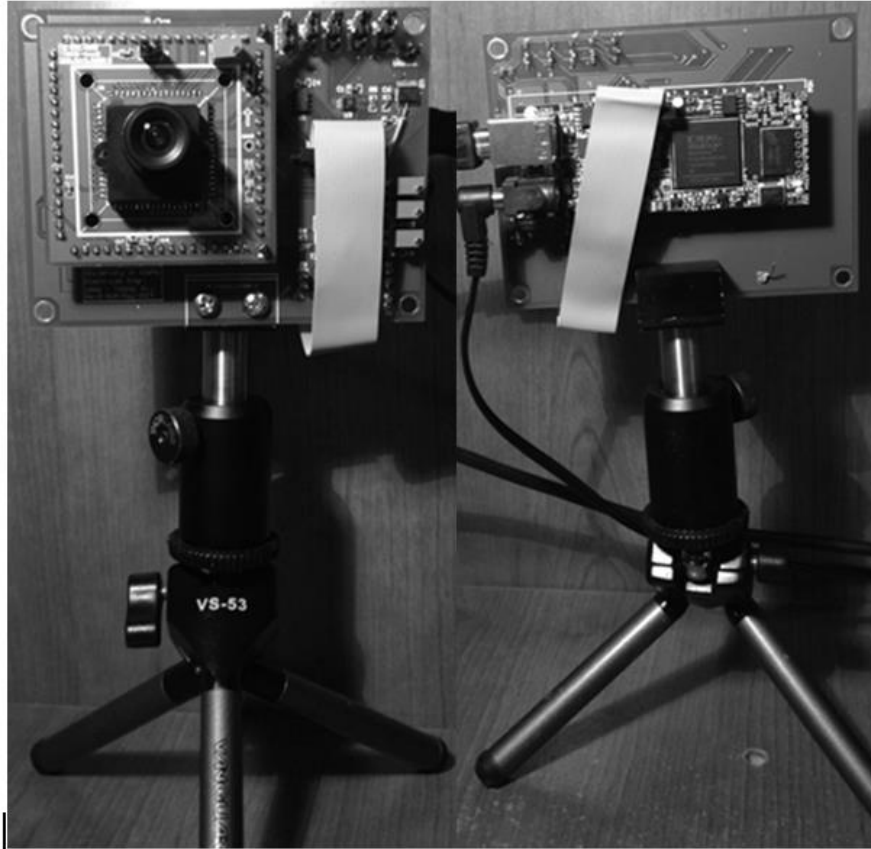
Figure 6-15. Custom TZOID camera system.

6.5   Summary

In this chapter, the details of TZOID image sensor were presented. The introduction and background of TZOID imager was depicted including current techniques that used to reduce irrelevant data in image sensor. The pixel design equations based on P-R model and details of TZOID pixel array development are discussed followed by TZOID image sensor architecture and illustration on analog and digital building blocks of the chip. Finally, the actual TZOID chip was displayed with a custom TZOID imager-based camera system.

CHAPTER 7 –TESTING AND MEASUREMENT RESULTS

Testing was carried out in both controlled laboratory environment and outdoor settings. Proposed P-RM method was tested first by using standard rectangular image sensors before being tested on TZOID imager.

In this chapter, testing and measurement results for P-RM method and TZOID image sensor are presented. First, the testing results from P-RM method implementing on commercial rectangular imager are introduced, which includes indoor lab testing, outdoor vehicle detection and outdoor pedestrian height estimation. Second, TZOID imager testing results including comparison results with standard rectangular imager are presented.

7.1   Testing on PR-M Method

There are two parts of testing for PR-M method because of existing two ways (5.23) and (5.24) to estimate tilting angle by P-RM method. It depends on the actual application as described in Chapter 5, section 5.5.1.

Part I includes series of experiments for vehicle detection applications, while Part II is for pedestrian detection.

A Cannon EOS 100D Rebel SL1 camera equipped with APS-C format (22.3mm x 14.9mm) CMOS image sensor was used in the experiments, which has 4.31µm pixel size. A Canon EF 40mm f/2.8 lens was selected for the camera body to acquire images. Both indoor laboratory and outdoor field tests were performed.

7.1.1   **Part I**

This section presents a series of experiments conducted for vehicle detection applications by using PR-M method.

7.1.1.1   *Indoor Laboratory Tests*

First, tilt angle calibration was performed determining errors under indoor laboratory environment settings. Subsequently, dimensions of a standard chessboard pattern were determined by reconstructing images by P-RM method.

7.1.1.1.1   Tilt Angle Calibration

The tilt angle could be calibrated by capturing a moving vehicle and using the equation (5.23). The camera was mounted at 32.7cm high (*H=0.327m*) using 0.35M pixels mode (720x480) as reference. The value of tilt angel measured by clinometer was used as truth value, which is 42.55 degree.
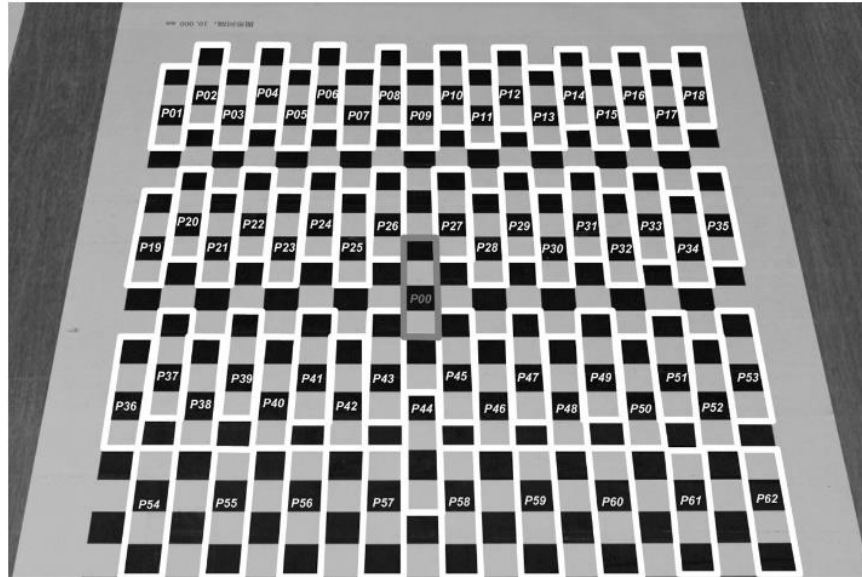
Figure 7-1. Different testing position for tilt angle calibration using length of the object.

A printed chessboard pattern with 10x10mm black and white unite lattice was used to verify the tilt calibration. Sixty-three positions are arranged as labeled *Pxx*, which have the same dimensions as shown in Figure 7-1. Assuming starting position of vehicle is *P00* and moving to the other position, such as *P01*, which gives one (the first) set of test condition for calibration, so that total of 62 testing positions/conditions could be accessed. RMS value of calibrated tilt angle was determined after repeating measurements for each test condition hundred times. The experiment results are shown in Figure 7-2 and Figure 7-3. Standard deviation for whole values of calibrated tilt angle is 0.29 $^\circ$ with 42.51° average value, which is 0.04° (0.09% error) less than the truth value, 42.55°, though the maximum tested error is 1.91%. The RMS value for total acquired number is 42.52° that is -0.08% from truth value.
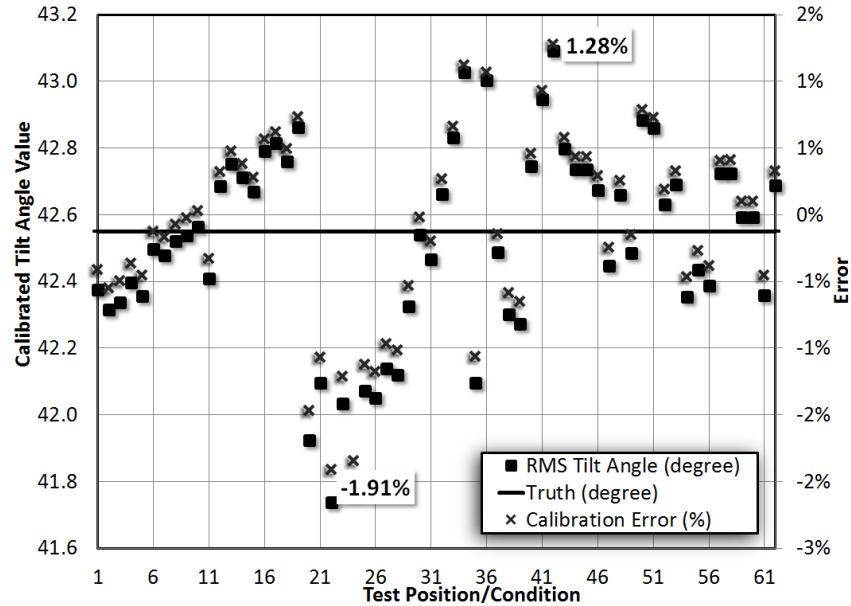
Figure 7-2. Achieved tilt angle with related error in tilt angle calibration testing.
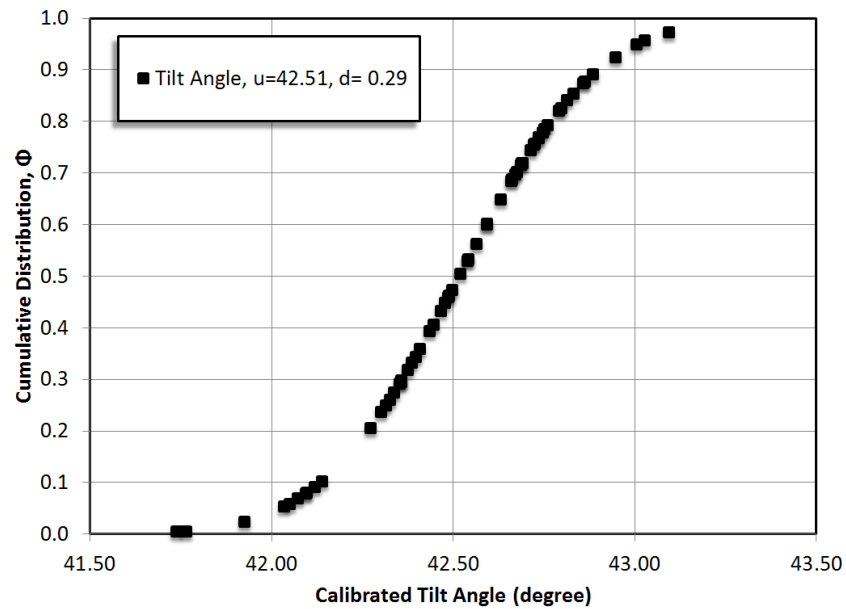
Figure 7-3. Cumulative distribution plot according to test results in tilt angle calibration testing.

7.1.1.1.2 *Two-Dimensional Measurement*

Each pixel's resolution could be obtained after acquiring tilt angle value using the equations (5.16) and (5.20). The taken images could be rebuilt by P-RM method. Examples of reconstructed images are shown in Figure 7-4(c) and Figure 7-4(d) relative to their original taken images, Figure 7-4(a) and Figure 7-4(b), respectively.

The dimension of each black and white unites/lattices (10x10mm as truth value) were measured by including both static and dynamic repeatability tests. The results from 100 times static repeatability tests is shown in Figure 7-5, where each point is the root mean square (RMS) value from 100 measuring data at each lattice. The largest error of width and length detection is 1.34% and -2.5%, respectively, while the largest maximum minimum deviation is 0.25mm for width detection and 0.45mm for length detection. 100 times dynamic repeatability testing result is shown in Figure 7-6, which is illustrating the largest error is -2.2% for width detection and -3.33% for length detection with largest maximum minimum deviation 0.40mm and 0.52mm, respectively. Amount of 99.73% of test errors would be smaller than 3.27%.
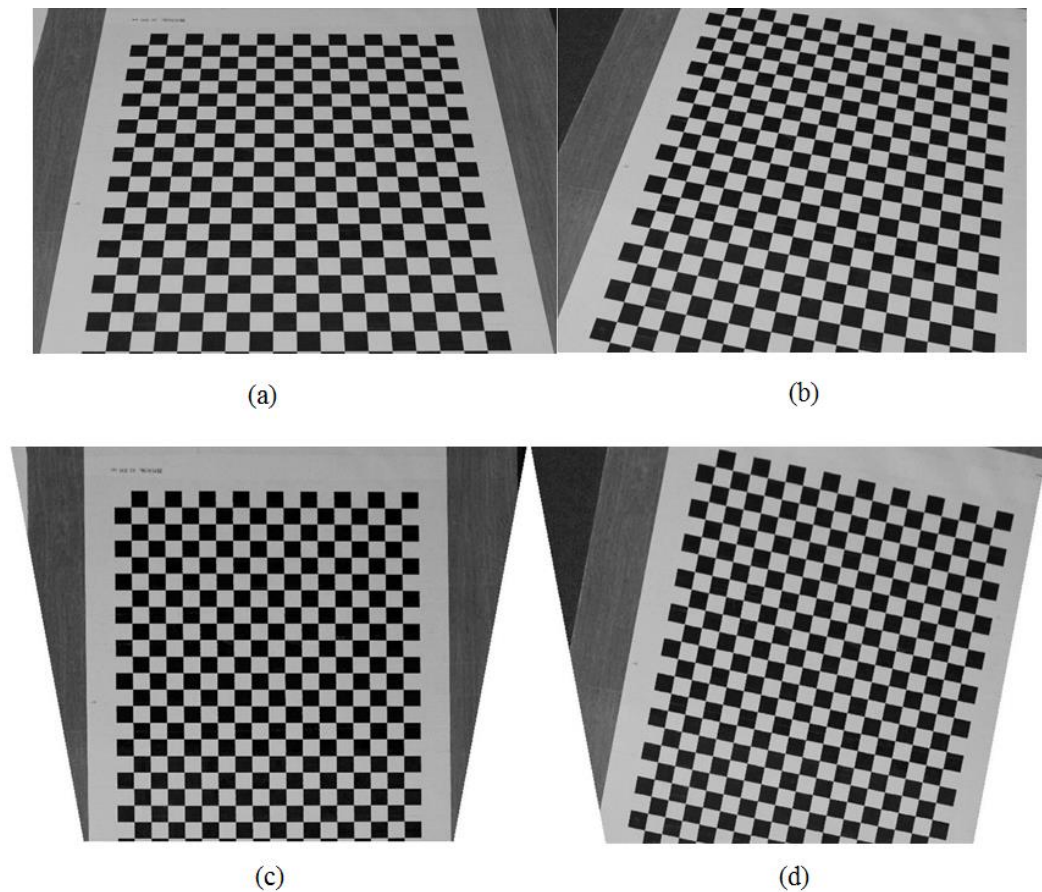
Figure 7-4. Sample pictures from P-RM method realization: (a) and (b) taken image, (c) and (d) relative reconstructed image.
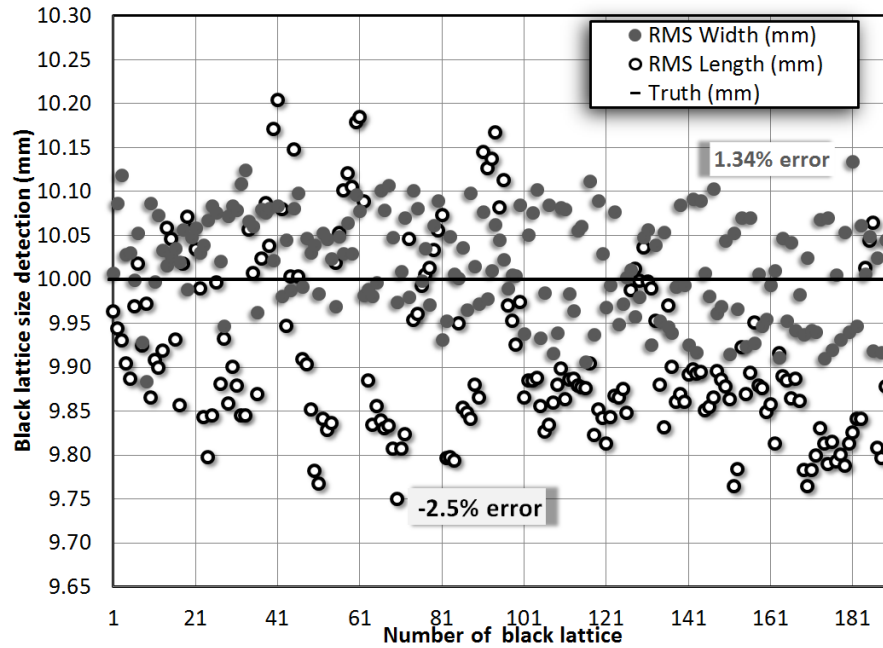
Figure 7-5. 100 times static repeatability test results on dimension measurement on each black and white lattice.
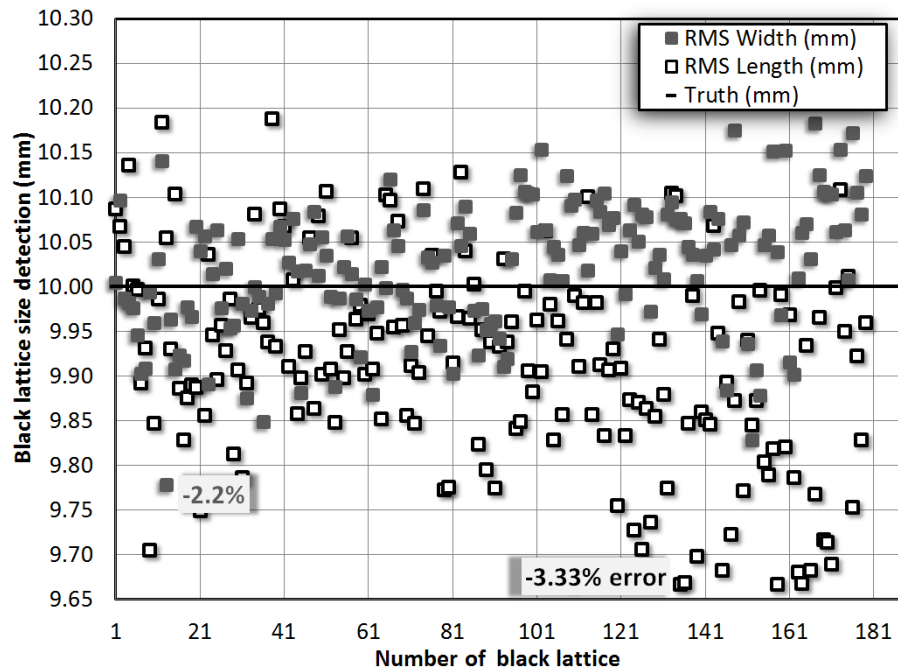
Figure 7-6. 100 times dynamic repeatability test results on dimension measurement on each black and white lattice.

Gaussian noise, the standard deviation of which is from 0 to 4 pixels, was considered for the edge detection measurements. This was used to investigate the performance of the P-

RM method in ill-conditions, such as blurry image, soft focus areas, and unstable edge detection algorithm. Figure 7-7 and Figure 7-8 reflect the noise testing results. The error of tilt angles, $\Delta\phi$, versus added noises is plotted in Figure 7-7. Two-dimension (2D) measurement error versus tilt angle calibrating error given in Figure 7-8.
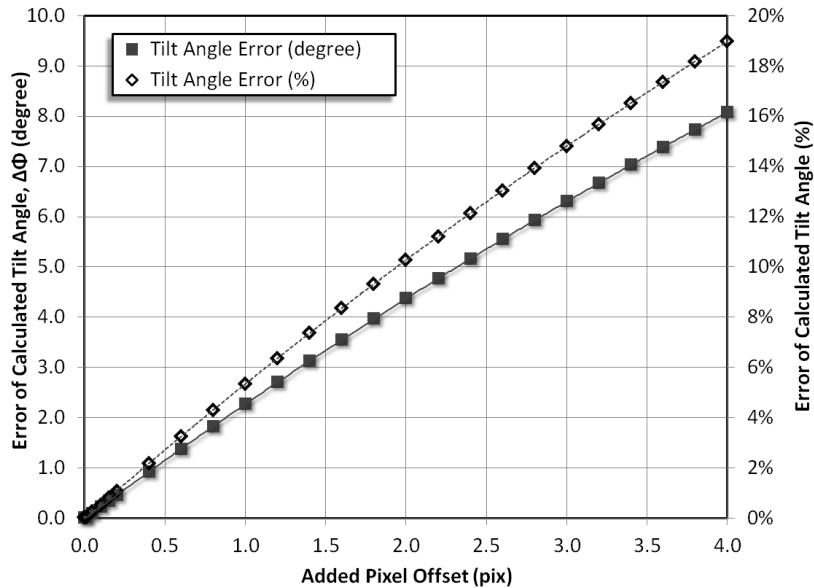
Figure 7-7. Tilt angle error versus added pixel offset
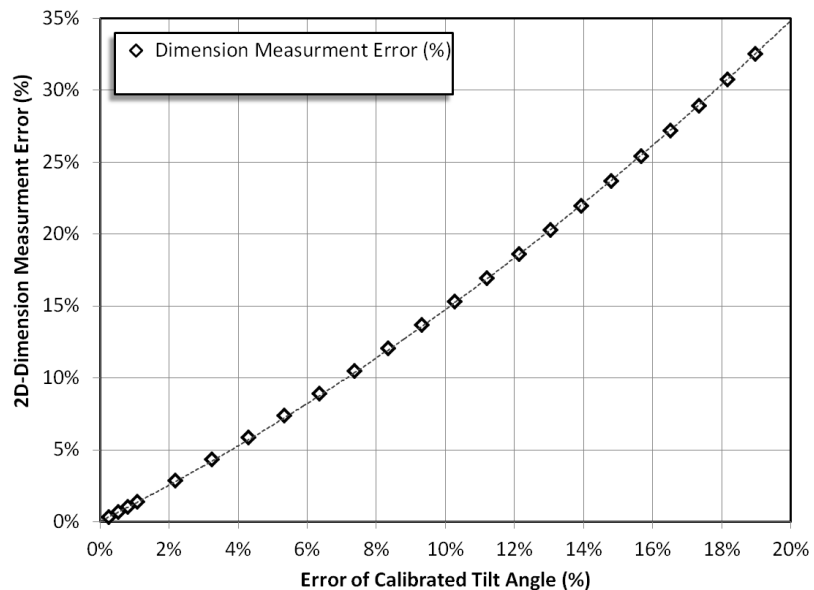
Figure 7-8. 2D dimension measurement error versus tilt angle error.

7.1.1.2   *Outdoor Field Tests*

The Cannon camera equipped with 40mm f/2.8 lens using 2M pixels mode (1920x1080) was used in this outdoor experiment. The camera was mounted at 28.71m high above the ground with 61.30 degree (truth value) tilt angle viewing a parking lot. Dimension mensuration (length test) and vehicle speed detection by using P-RM method were both applied, in addition to tilt angle calibration test.

(a)   (b)

(c)
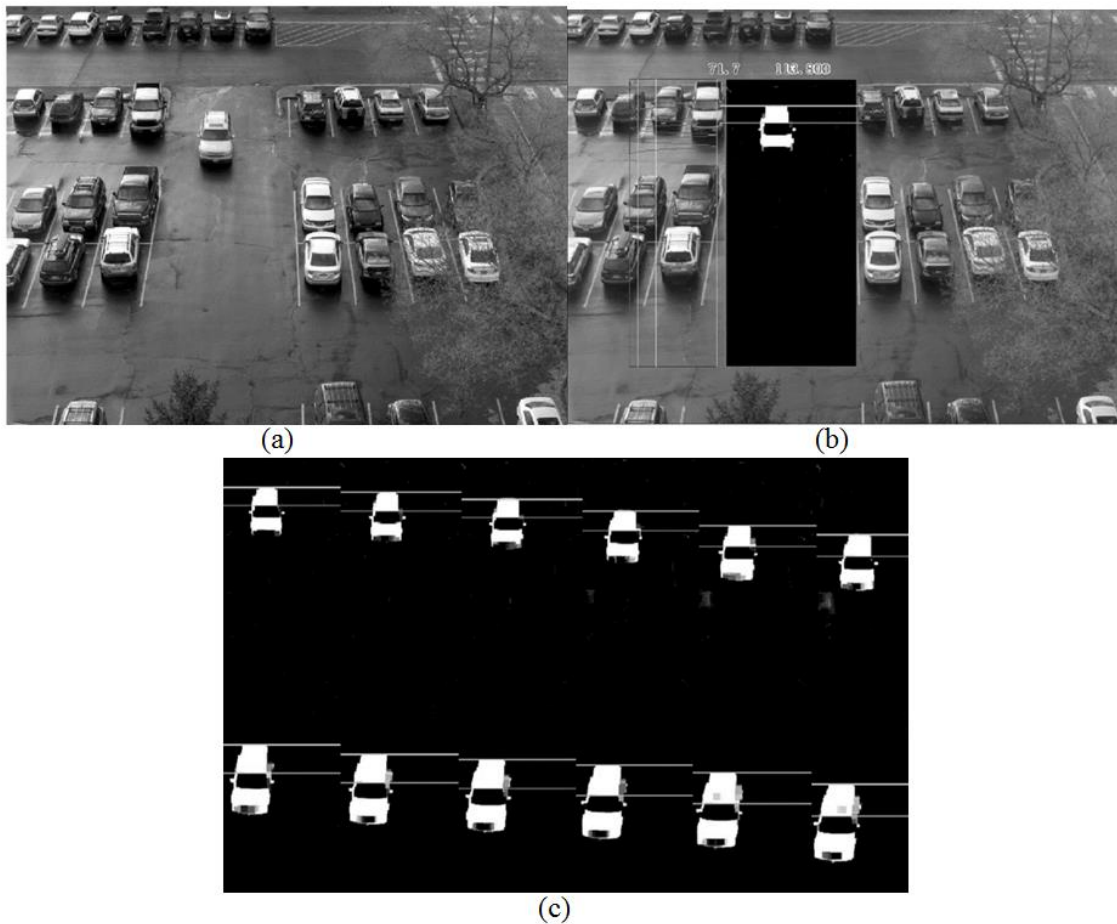
Figure 7-9. Outdoor field test on tilt angel calibration: (a) taken image (b) image processed by background subtraction and edge detection (c) sequences of detected vehicle's roof.

7.1.1.2.1   *Tilt Angle Calibration*

A set of images contained a moving vehicle captured in a parking lot area is shown in Figure 7-9(a). The moving vehicle was the target for tilt angle calibration, which was first

detected by background subtraction using kernel density estimation (KDE) method [51] as show in Figure 7-9(b). Subsequently, the edge detection algorithm based on pixel intensity gradient was implemented to obtain the edges of moving vehicle's roof. In Figure 7-9(c), the examples of moving vehicle extracted from a sequence of frames with detected roof edges are illustrated, which could be used for tilt angle calibration by (5.23). The calibration results from 120 groups of frames are shown in Figure 7-10(d). The average value of calibrated tilt angle is 61.291 degree, which has 0.01% error from the truth value with 0.525% standard deviation, though the maximum error of 2.61% occurred at frame# 52.
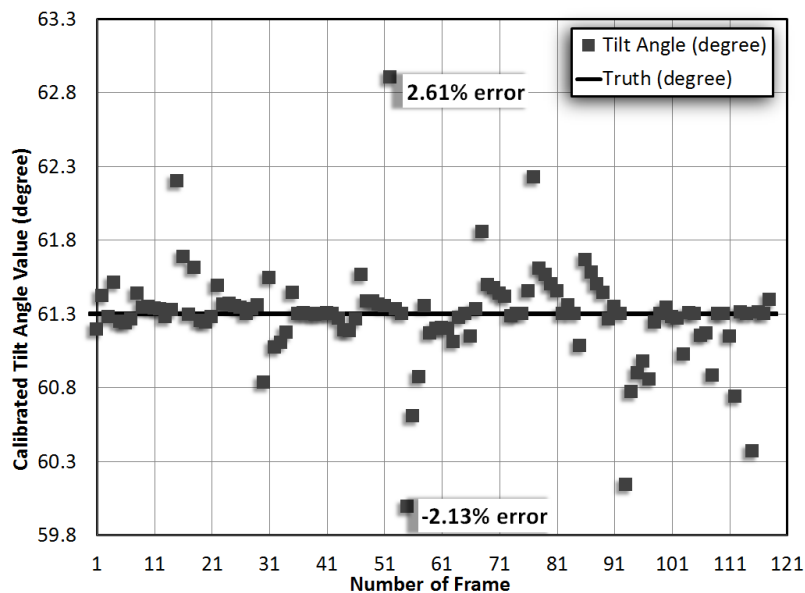
Figure 7-10. Computed tilt angle results.

7.1.1.2.2   *Feature Extraction*

To evaluate the accuracy, in Figure 7-11, twenty-two test lines (L1 to L22) were chosen for evaluation. The experimental lengths were computed by Euclidean distance method after obtaining each pixel's resolution. The mean computed length and computed error of each test line are illustrated in Table 7-1. The proposed P-RM method produced highly accurate length estimation with mean accuracy of around 99.15% and overall accuracy better than 98.6%.
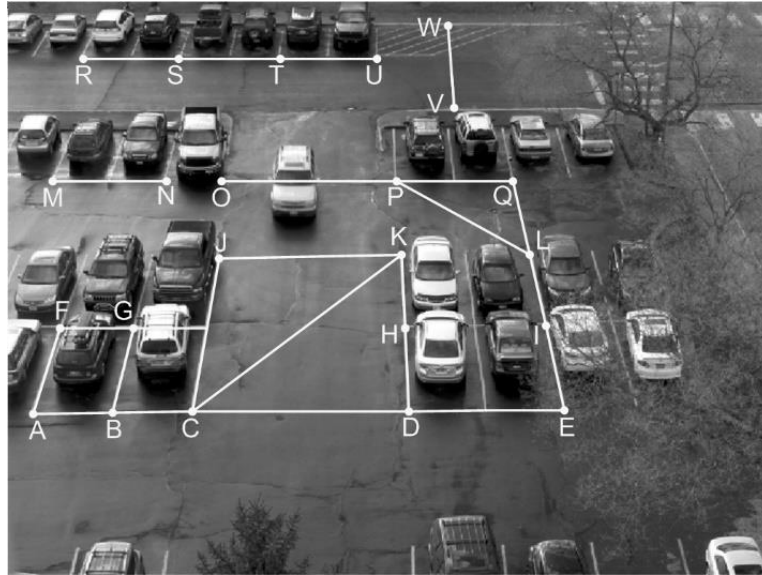
Figure 7-11. Labeled end points and lines for length test.

Table 7-1. Extracted lengths for outdoor field test.

Trial	Line	End points	Actual length (m)	Computed length (m)	Computed error	Accuracy
1	L_1	AB	2.885	2.900	0.536%	99.464%
2	L_2	BC	2.835	2.862	0.952%	99.048%
3	L_3	CD	7.690	7.771	1.049%	98.951%
4	L_4	DE	5.750	5.787	0.645%	99.355%
5	L_5	FG	2.875	2.876	0.035%	99.965%
6	L_6	JK	7.700	7.773	0.952%	99.048%
7	L_7	MN	5.290	5.338	0.913%	99.087%
8	L_8	OP	8.000	8.103	1.285%	98.715%
9	L_9	PQ	5.610	5.666	1.000%	99.000%
10	L_{10}	RS	5.530	5.572	0.767%	99.233%
11	L_{11}	ST	5.620	5.666	0.820%	99.180%
12	L_{12}	TU	5.560	5.619	1.066%	98.934%
13	L_{13}	AF	6.100	6.150	0.814%	99.186%
14	L_{14}	BG	6.050	5.979	-1.174%	98.826%
15	L_{15}	DH	5.950	5.900	-0.840%	99.160%
16	L_{16}	EI	5.900	5.845	-0.936%	99.064%
17	L_{17}	HK	5.860	5.803	-0.976%	99.024%
18	L_{18}	IL	5.900	5.925	0.428%	99.572%
19	L_{19}	LQ	7.190	7.251	0.846%	99.154%
20	L_{20}	VW	12.670	12.775	0.827%	99.173%
21	L_{21}	PL	10.210	10.330	1.175%	98.825%
22	L_{22}	CK	14.670	14.781	0.757%	99.243%

The speed of the moving vehicle in this experiment was also computed. The camera was running at 24 frames per second (FPS) so that the time difference, Δt_{AB}, between *Frame* n_A and *Frame* n_B is $(n_B-n_A)/24$ second. The leading edge of the moving vehicle was detected as shown in Figure 7-12, which was used to determine the moving distance, S, between two frames. The position, P, of the leading edge could be achieved by (5.14) so that the moving distance from *Frame* n_A to *Frame* n_B, ΔS_{AB}, equals to P_A minus P_B. Consequently, the speed of the vehicle, $V_{vehicle}$, is obtained by (7.1).

$$V_{vehicle} = \frac{\Delta S_{AB}}{\Delta t_{AB}} \tag{7.1}$$
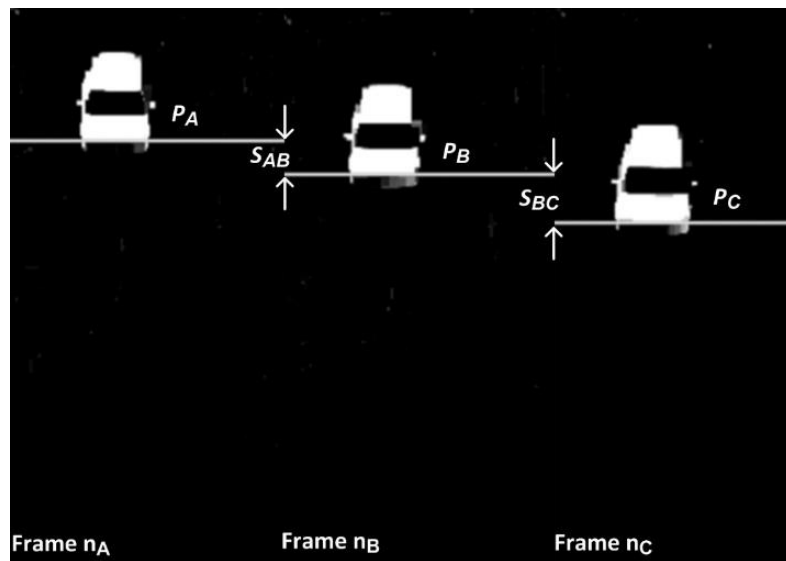
Figure 7-12. Example frames for moving vehicle speed detection by leading edge detection.

The vehicle speed was calculated at every ten frames, the result of which is shown in Table 7-2. The mean value of speed detection accuracy is 98.52% with maximum error of -2.244%. The overall speed detection accuracy is better than 99.7%.

Table 7-2 Extracted speed of moving vehicle for outdoor field test.

Trial	Frame #	Leading edge (pix)	Leading edge (m)	Actual Speed (m/s)	Computed speed (m/s)	Computed error	Accuracy
1	1	101.473	69.162	2.235	/	/	/
2	11	107.585	68.246	2.235	2.198	-1.676%	98.324%
3	21	113.800	67.335	2.235	2.187	-2.144%	97.856%
4	31	121.159	66.422	2.235	2.190	-2.025%	97.975%
5	41	128.208	65.512	2.235	2.185	-2.244%	97.756%
6	51	133.346	64.601	2.235	2.185	-2.241%	97.759%
7	61	139.852	63.684	2.235	2.200	-1.559%	98.441%
8	71	146.736	62.756	2.235	2.229	-0.264%	99.736%
9	81	155.220	61.826	2.235	2.230	-0.237%	99.763%
10	91	161.879	60.902	2.235	2.218	-0.763%	99.237%
11	101	170.751	59.985	2.235	2.201	-1.521%	98.479%
12	111	178.192	59.070	2.235	2.196	-1.754%	98.246%
13	121	186.975	58.150	2.235	2.208	-1.217%	98.783%
14	131	194.570	57.233	2.235	2.202	-1.485%	98.515%
15	141	202.551	56.316	2.235	2.201	-1.539%	98.461%
16	151	211.207	55.398	2.235	2.203	-1.423%	98.577%

7.1.2 **Part II**

Part II is a series of experiments implemented for pedestrian detection application by using PR-M method.

7.1.2.1 *Indoor Laboratory Tests*

First, tilt angle and mounting height calibration was performed determining errors under indoor laboratory environment settings. Subsequently, dimensions of a standard chessboard pattern which was perpendicular to the ground were determined by P-RM method.

7.1.2.1.1 *Tilt Angle Calibration*

The tilt angle could be calibrated by capturing a moving object's top and bottom edges with using (5.24). The camera was mounted at 296.00mm high (H=0.2960m) using 2.5M pixels mode (1920x1280) as reference. The value of tilt angel measured by clinometer was used as truth value, which is 40.10°.

A wooden cube with height of 53.88mm was used as a moving object for tilt angle calibration test, which was moving from *P0* to *P17* as shown in Figure 7-13 including total 18 positions. Any two images from those two positions could be set as a pair for tilt angle calibration, so that total 153 testing pairs could be arranged, which means 153 groups of tests. RMS value of calibrated tilt angle was determined after repeating measurements for each test pair hundred times.

The experiment results are shown in Figure 7-14. Standard deviation for whole values of calibrated tilt angle is $0.232°$ with $40.141°$ RMS value, which is $0.041°$ (0.11% error) larger than the truth value, $40.10°$, though the maximum tested error is -1.47%.
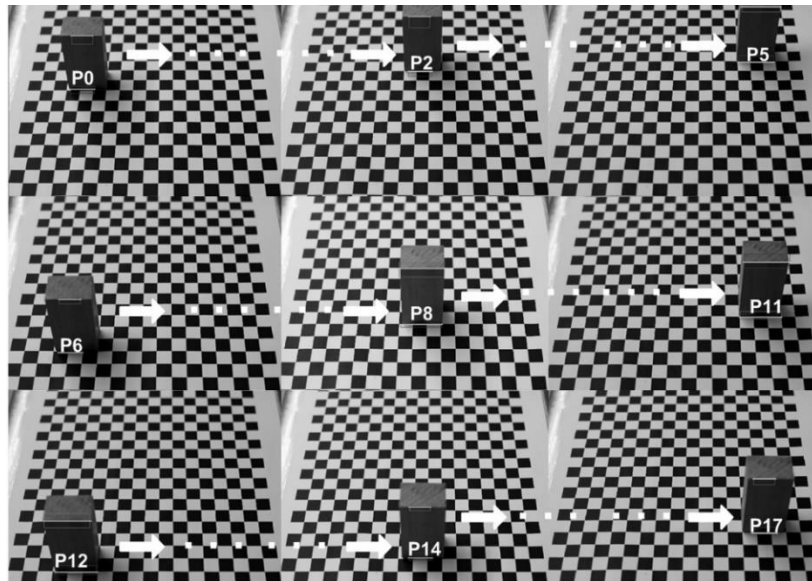
Figure 7-13. Different testing position for tilt angle calibration indoor experiment.
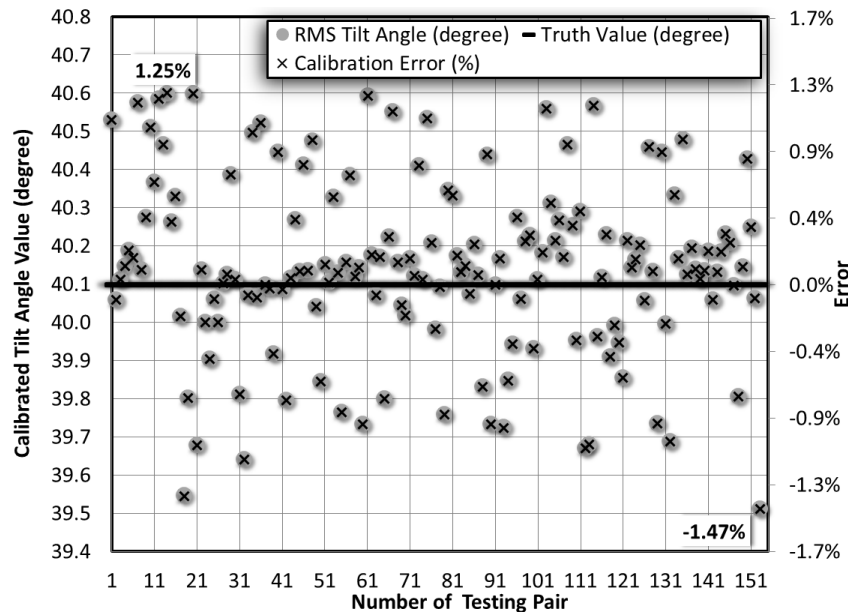
Figure 7-14. Achieved tilt angle with related error in tilt angle calibration testing.

7.1.2.1.2   *Mounting Height Calibration*

The height of wooden cube is 55.00mm, which could be used as reference height, h_{ref}, for mounting height, H, calculation by using (5.21) derived in Chapter 5, Section 5.4.2. Eighteen test positions used in previous section were also imitated for mounting height calibration test. To keep the same manner as tilt angle calibration experiment, one hundred images captured at each position were used for gathering RMS values of mounting height achieved at each testing position. The testing result is showing in Figure 7-15, which illustrates that the mounting height calibration accuracy could achieve to better than 99.6% with maximum -0.38% error at *P0* in the experiment. The standard deviation from total 1800 values of calibrated height, H, is 0.729mm with 296.08mm RMS value, which is 0.08mm (0.024% error) larger than the truth value, 296.00mm.

The calibration results for indoor testing setup are listed in Table 7-3.
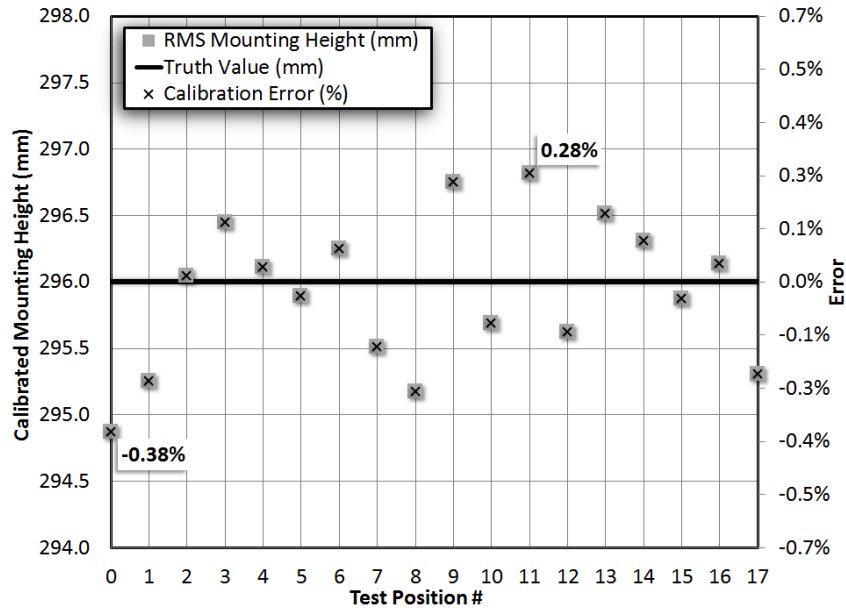
Figure 7-15. Achieved mounting height with related error in mounting height calibration testing.

Table 7-3 Calibration Results for Indoor Laboratory Tests.

Result	Computed tilt angle	Computed mounting height
Average value	40.14º	296.08mm
Average error	0.11%	0.024%

7.1.2.1.3   *Three-Dimensional Feature Extraction*

Two wooden cubes were placed on the ground plane which was covered by a printed chessboard pattern sheet as shown in Figure 7-16(a). The PR-M method was launched for feature extraction by substituting tilt angle value and mounting height value calibrated in Table 7-3, as a result, the pixels' spatial resolutions in both column and row direction, R_y, and, R_x, at ground plane could be achieved by (5.16) and (5.20), which are plotted in Figure 7-17. In Figure 7-16(b), total eighteen end points with twenty-five lengths were detected and labeled by using edge detection for verifying proposed mensuration accuracy. The RMS value of computed length and computed error of each test line from one hundred times static repeat images are listed in Table 7-4. The results show that the proposed method produced both accurate 2D length and 3D height estimation with mean accuracy of 99.47% and 99.30%, respectively, in addition to the overall accuracy is better than 98.7%.
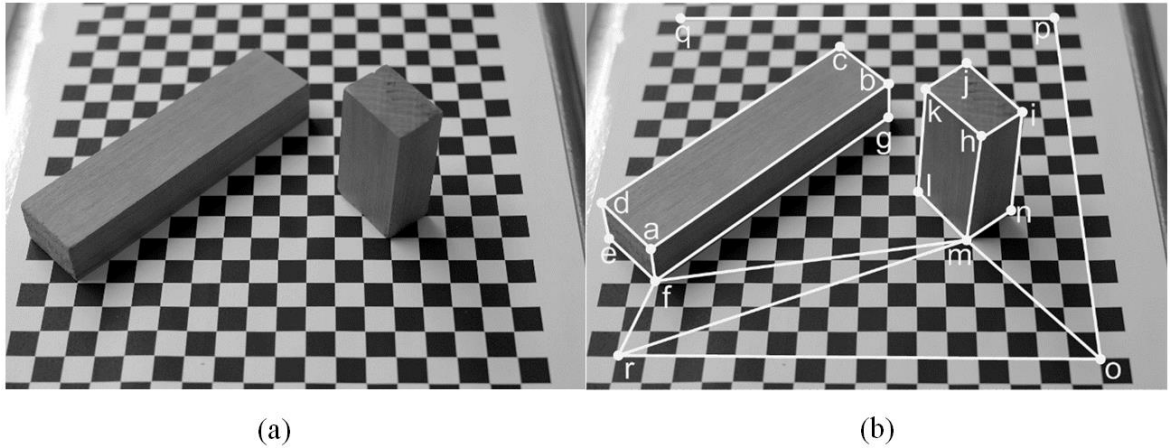
Figure 7-16. Indoor experiment on feature extraction (a) raw image (b) image labeled end points and lines for length estimation.
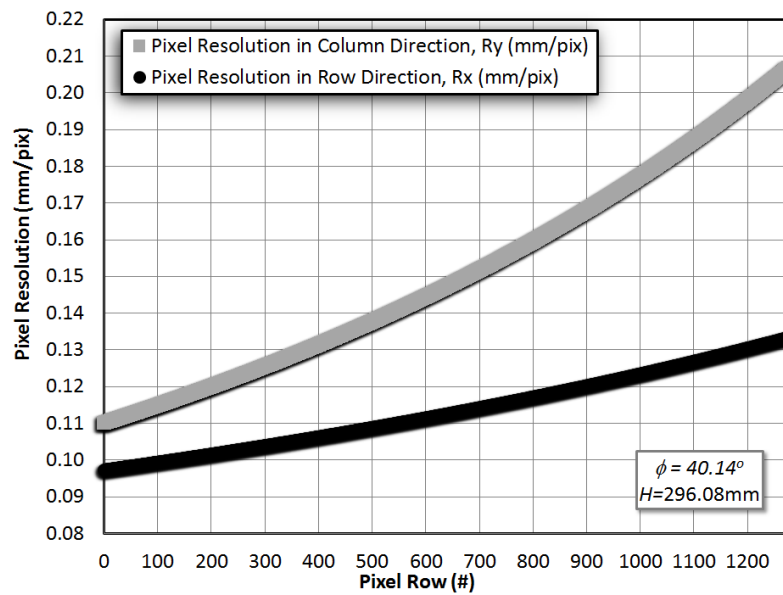
Figure 7-17. Achieved pixels' (spatial) resolutions by P-RM method.

The bird's eye view of two cubes' edges in Figure 7-16(a) is reconstructed in Figure 7-18, which includes reconstructed edges points with corner points' coordinates and a restored ground plane/background image with correct dimensions. Additionally, three-dimensional views of reconstructed scene are generated as shown in Figure 7-19 and Figure 7-20 containing 3D reconstruction of the camera location, two wooden cubes positions, and ground plane with actual scale.

Table 7-4. Extracted lengths and heights for indoor test.

Trial	Line	End points	Actual length (m)	Computed length (m)	Computed error	Accuracy
1	l_1	ab	113.5	113.81	0.27%	99.73%
2	l_2	bc	28.1	27.95	-0.53%	99.47%
3	l_3	cd	113.0	113.45	0.40%	99.60%
4	l_4	da	27.9	27.96	0.22%	99.78%
5	l_5	fe	28.0	28.35	0.89%	99.11%
6	l_6	fg	114.5	114.50	0.00%	100.00%
7	l_7	hi	18.9	19.11	1.10%	98.90%
8	l_8	ij	28.6	28.35	-0.86%	99.14%
9	l_9	jk	18.9	18.67	-1.35%	98.76%
10	l_{10}	kh	28.6	28.60	0.02%	99.98%
11	l_{11}	fm	111.8	111.27	-0.47%	99.53%
12	l_{12}	rm	130.0	129.16	-0.65%	99.35%
13	l_{13}	mo	64.0	63.85	-0.23%	99.77%
14	l_{14}	op	170.0	170.69	0.41%	99.59%
15	l_{15}	pq	160.0	161.44	0.90%	99.10%
16	l_{16}	qr	170	169.51	-0.29%	99.71%
17	l_{17}	ro	160.0	159.36	-0.40%	99.60%
18	l_{18}	oq	233.45	232.17	-0.55%	99.45%
19	l_{19}	rp	233.45	234.71	0.54%	99.46%
20	h_1	af	18.8	18.61	-1.02%	98.98%
21	h_2	de	18.9	18.65	-1.30%	98.70%
22	h_3	bg	18.9	18.61	-1.55%	98.45%
23	h_4	hm	53.8	53.81	0.02%	99.98%
24	h_5	kl	53.9	53.84	-0.12%	99.88%
25	h_6	in	53.7	53.80	0.19%	99.81%
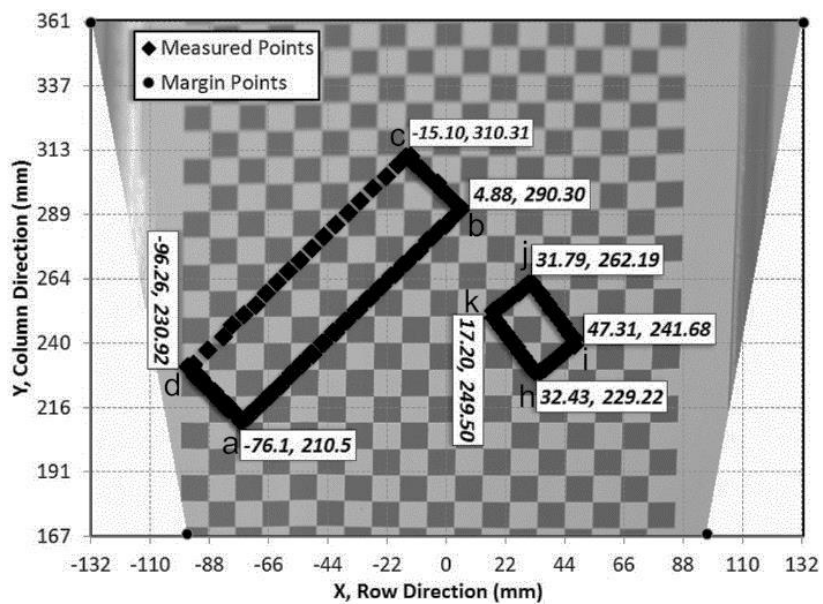
Figure 7-18. Overhead view of two detected cube's edges with measured corner points.
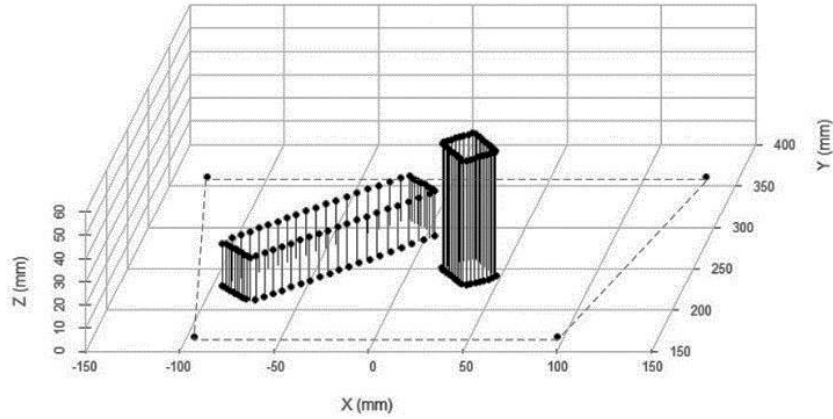
Figure 7-19. 3D view of detected cubes in FOV.
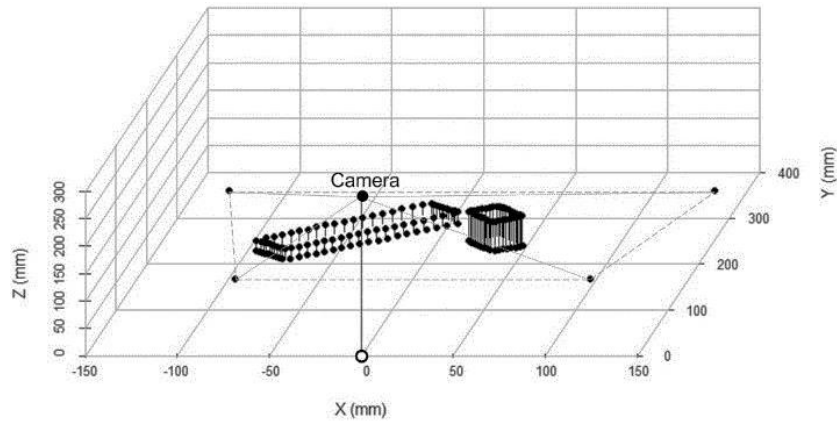
Figure 7-20. 3D view of reconstructed scene including camera's mounting position.

Gaussian noise, the standard deviation of which is from 0 to 4 pixels, was considered for the edge detection measurements. This was used to investigate the performance of the P-RM method in ill-conditions, such as blurry image, soft focus areas, and unstable edge detection algorithm.
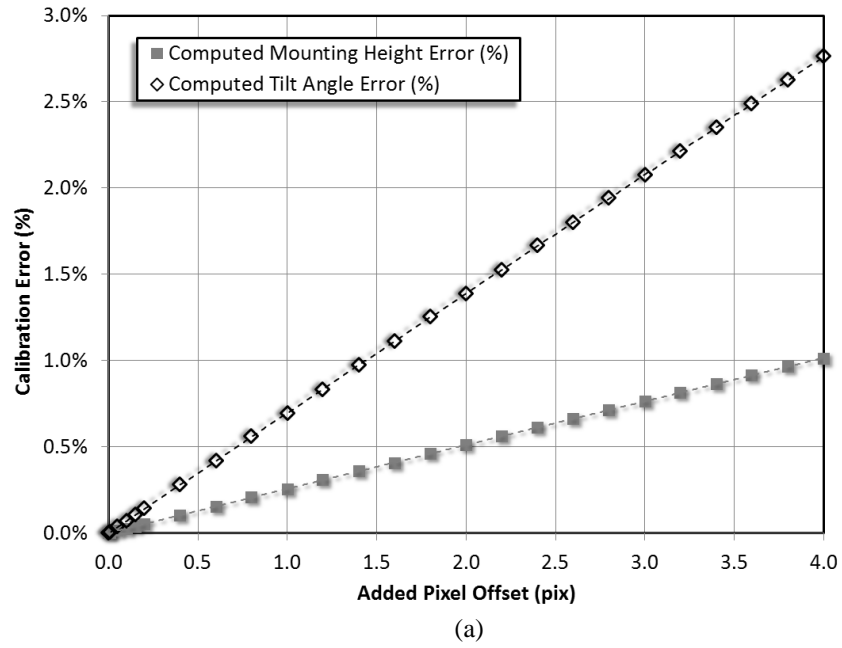
(a)
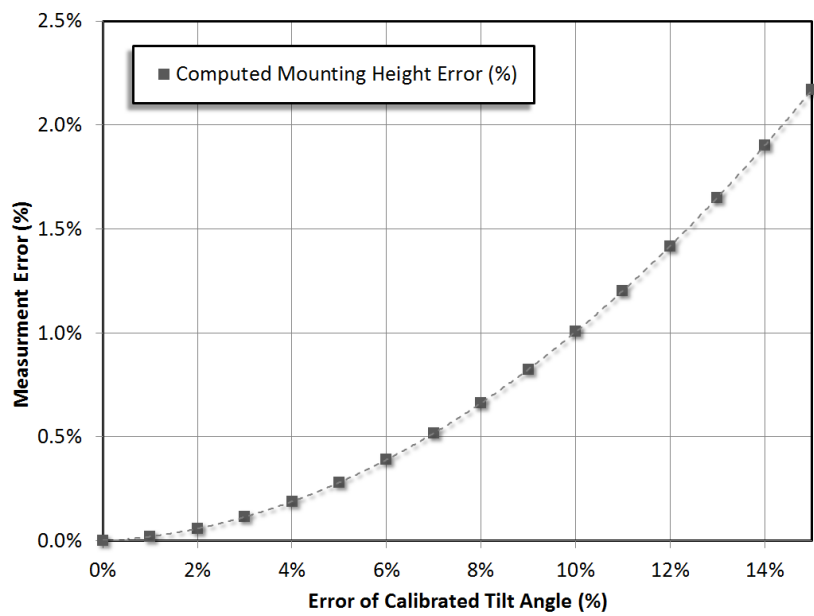
(b)
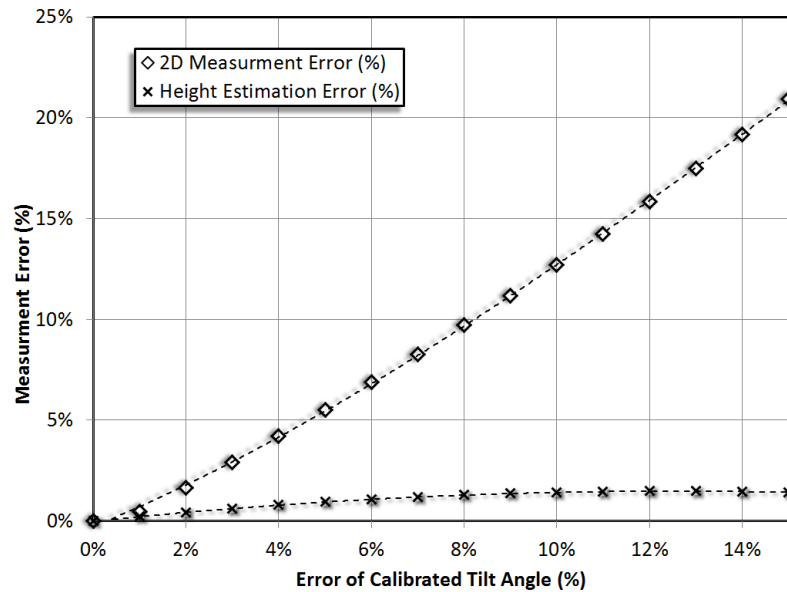
(c)
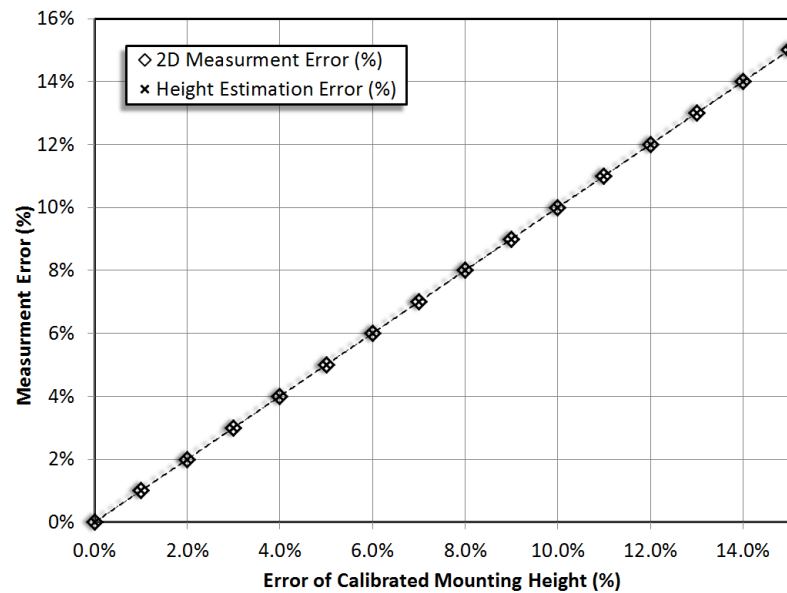
(d)

Figure 7-21. Noise testing results: (a) tilt angle error and mounting height error versus added pixel offset (b) mounting height error versus tilt angle error (c) three-dimension measurements error versus tilt angle error (d) three dimension measurements error versus mounting height error.

Four images in Figure 7-21 reflect the noise testing results. The error of computed tilt angles and error of computed mounting height versus added noises is plotted in Figure 7-21(a), which means that four 4 pixels uncertainty will introduce maximum 2.8% error to tilt angle and 1% error to mounting height. Computed mounting height error versus error of computed

tilt angle is shown in Figure 7-21(b). Two-dimension (2D) measurement error and height estimation error versus tilt angle calibrating error and mounting height calibrating error are given in Figure 7-21(c) and Figure 7-21(d), respectively. The error of measuring is proportional to the error of mounting height since the estimated object height is proportional to the camera's mounting height as shown in (5.21). The accuracy of 2D mensuration depends on tilt angle heavily. On the other hand, the height estimation takes less affection on the accuracy of tilt angle calibration.

7.1.2.2   *Outdoor Field Tests*

The Cannon camera equipped with 40mm f/2.8 lens using 2M pixels mode (1920x1080) was used in this outdoor experiment. The camera was mounted at 8.378m (truth value) high above the ground with 52.50^o (truth value) tilt angle viewing a lobby hall. Tilt angle and mounting height calibration test and dimension mensuration (pedestrian height estimation) test were both applied.

Three student volunteers labeled as PA (1.590m), PB (1.840m) and PC (1.745m) were walking or standing in the FOV of the camera as shown in example frames in Figure 7-22 (a) and (b). The volunteers were playing roles of the reference target for calibration and detection targets for height estimation. The feet edge (bottom edge) and head edge (top edge) of the volunteers were acquired by background subtraction using kernel density estimation method [51] with Sobel edge detection operator [98] as show in Figure 7-22 (c) and (d), which were essential for calibration and dimension information extraction.
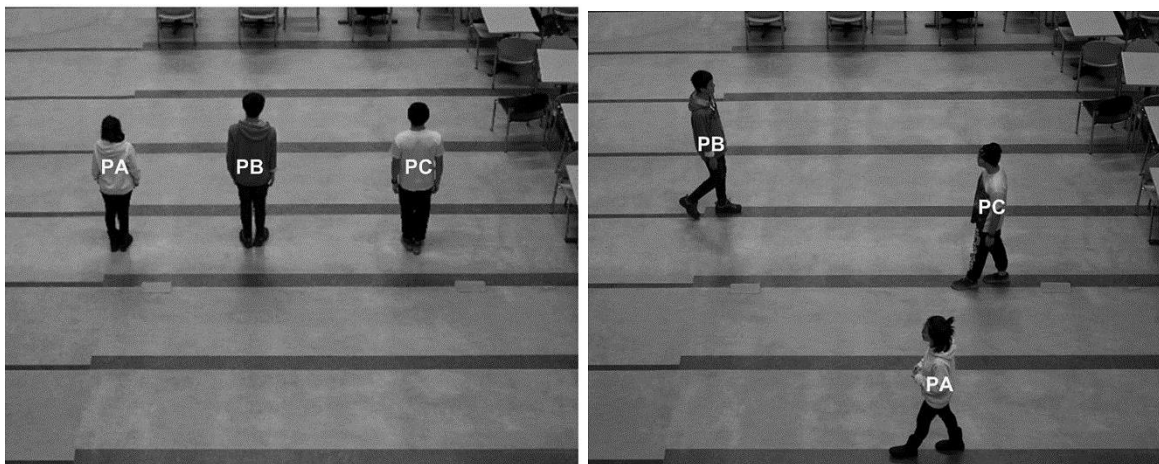
(a)   (b)
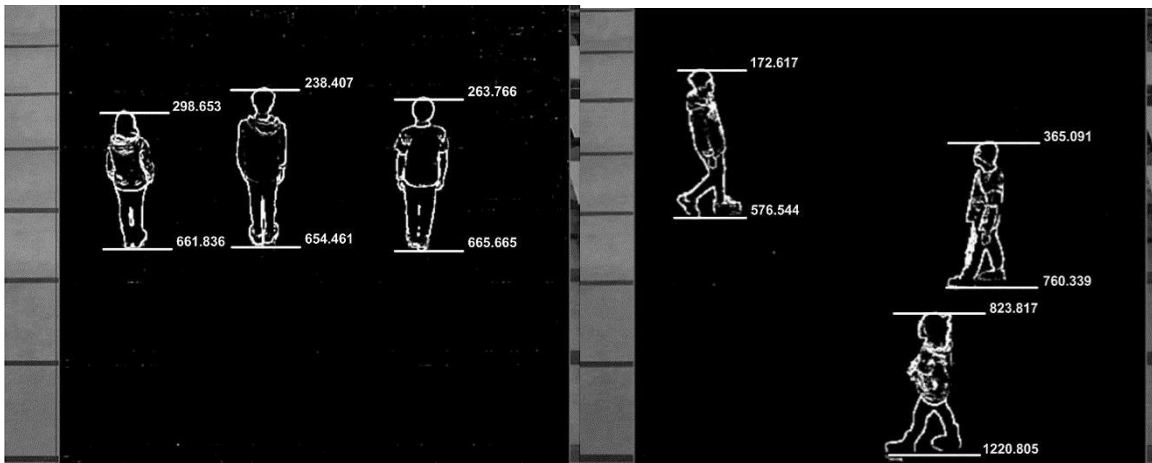
(c)

(d)
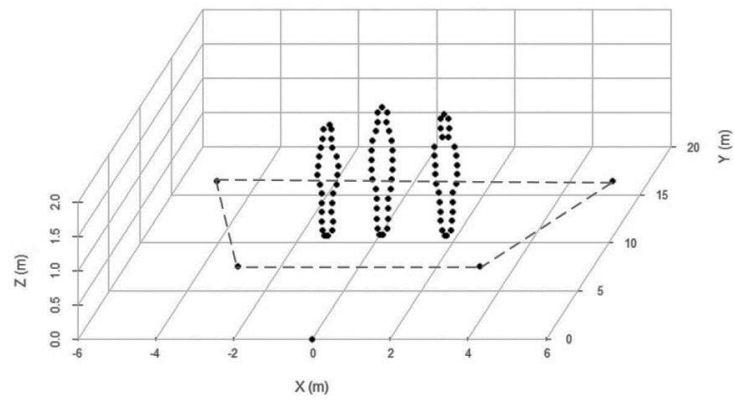
(e)
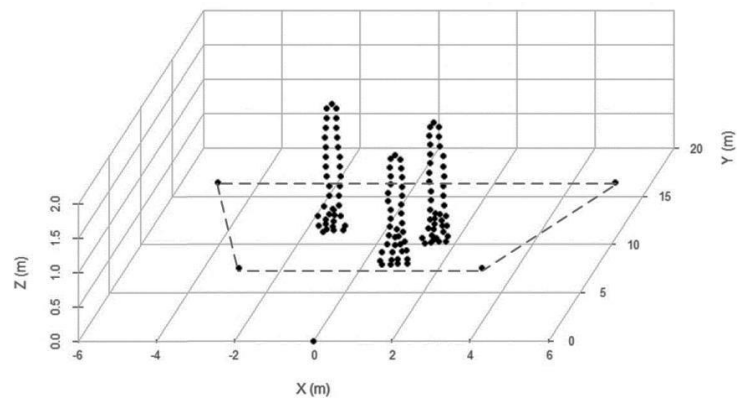
(f)

Figure 7-22. Sample frames of outdoor test: (a) and (b) taken images (c) and (d) images processed by background subtraction and edge detection for obtaining head and feet edges of a person (e) and (f) 3D reconstructed images of (a) and (b).

7.1.2.2.1   Tilt Angle and Mounting Height Calibration

The three volunteers were regarded as calibration target and reference height object successively for tilt angle and mounting height calibration. The calibration steps are the same as the procedures illustrated in section 7.1.2.1.1 and section 7.1.2.1.2.

Twenty pairs of frames were randomly selected for computing the tilt angle value depending on reference target of PA, PB, and PC, respectively. The average values of computed tilt angles depending on PA, PB, and PC, which were respectively used in mounting height calculation test, are listed in Table 7-5. The initial testing results on tilt angles were shown in Figure 7-23 indicating that the maximum computed error for whole test is 2.36% with average value of 52.83º (0.63% error), the standard deviation of which is 0.46º.

Twenty frames including three volunteers were processed for mounting height calibration test after tilt angle calibrating. The testing results are exhibited in Figure 7-24. The average value of calibrated mounting height is 8.395m, which has 0.21% error from the truth value with 0.079m standard deviation, though the maximum error of 2.26% at frame# 321.
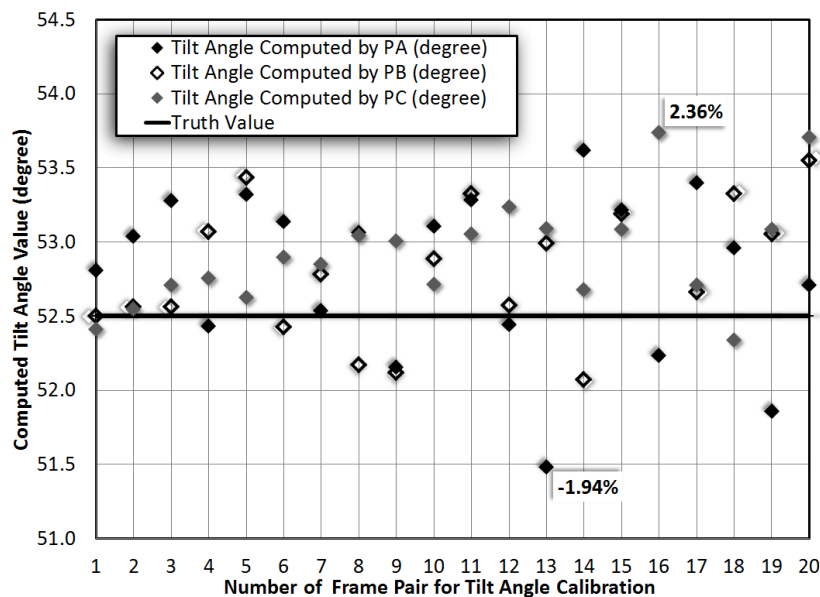
Figure 7-23. Tilt angle calibration results.
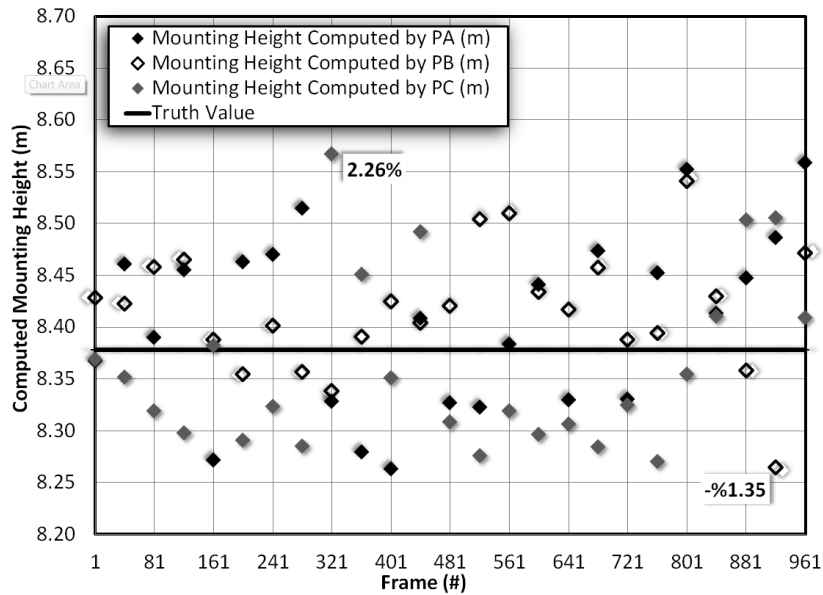
Figure 7-24. Mounting height calibration results.

Table 7-5. Computed tilt angle, ϕ, and mounting height, H, for outdoor test.

	Cali. target/ Ref. height	Computed ϕ (º) mean	Actual ϕ (º)	Accuracy	Computed H (m) mean	Actual H (m)	Accuracy
1	PA	52.768	52.50	99.49%	8.407	8.378	99.65%
2	PB	52.805	52.50	99.42%	8.417	8.378	99.53%
3	PC	52.913	52.50	99.21%	8.362	8.378	99.81%
Over All	/	52.827	52.50	99.38%	8.395	8.378	99.79%

The tilt angle and mounting height calibration results are summarized in Table 7-5, which gives the mean computed values and computed accuracies of each volunteers as calibration target. The proposed method produced accurate calibration with mean accuracy of around 99.38% and overall accuracy better than 97.64%.

7.1.2.2.2   Pedestrian Height Estimation

The previous computed values, 52.83º and 8.395m, were used as values of tilt angle and mounting height in (5.21), respectively, for pedestrian height estimation. The three volunteers were randomly walking in the FOV as the camera was capturing for height information extraction. As shown in Figure 7-22(c) and (d), the profiles of the volunteers were extracted with top/head and bottom/foot edges detection of the volunteers. Consequently, the height of the person could be estimated by proposed PR-M method as described in section

5.4.2. The examples of 3D reconstructed images for Figure 7-22 (a) and (b) are exhibited in Figure 7-22 (e) and (f), in addition to Figure 7-25, which demonstrates the experimental results of height estimation on three volunteers in one hundred different frames. The summarized estimation results are listed in Table 7-6, which shows that the overall accuracy on height estimation could achieve to better than 96%.
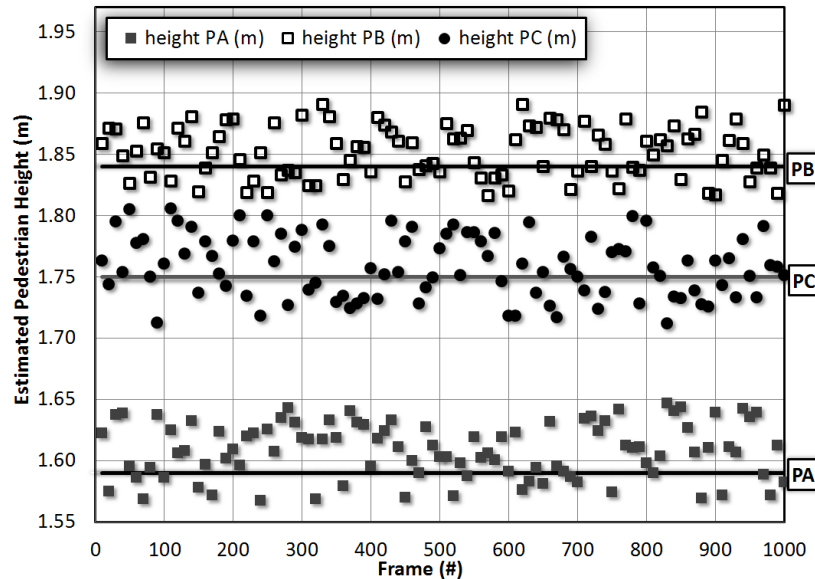
Figure 7-25. Pedestrian height estimation results on three volunteers from one hundred different frames.

Table 7-6. Height estimation results for outdoor test.

	Volunteer	Actual height (m)	Estimated height mean (m)	Standard Deviation (m)	Error mean	Error max	Accuracy
1	PA	1.590	1.608	0.022	1.15%	3.31%	96.69%
2	PB	1.840	1.852	0.021	0.63%	2.98%	97.02%
3	PC	1.750	1.763	0.025	0.77%	3.48%	96.52%

The proposed PR-M method calibrated mounting information correctly and recovers the 3D information from a 2D image accurately with 96% height detection accuracy. Note that smaller pixel size, increasing number of total pixels in the array, and more elegant image processing algorithms could generate more accuracy detection results and yield more robust 3D reconstructions.

7.2   Testing on TZOID Image Sensor with PR-M Method

The TZOID camera setup was mounted at 0.28m above ground plane tilting at 24 degree with a fixed focus-lengths lens (*f=4.42mm*). The distribution of pixels' resolutions is shown in Figure 7-26, which keeps 6 different resolution sections detecting from 0.52m to 6.98m away, though the mounting height was decreased by 8.56m. The percentage (96.833%) of mounting height reduced is the percentage (96.83%) of pixel resolution enhanced.
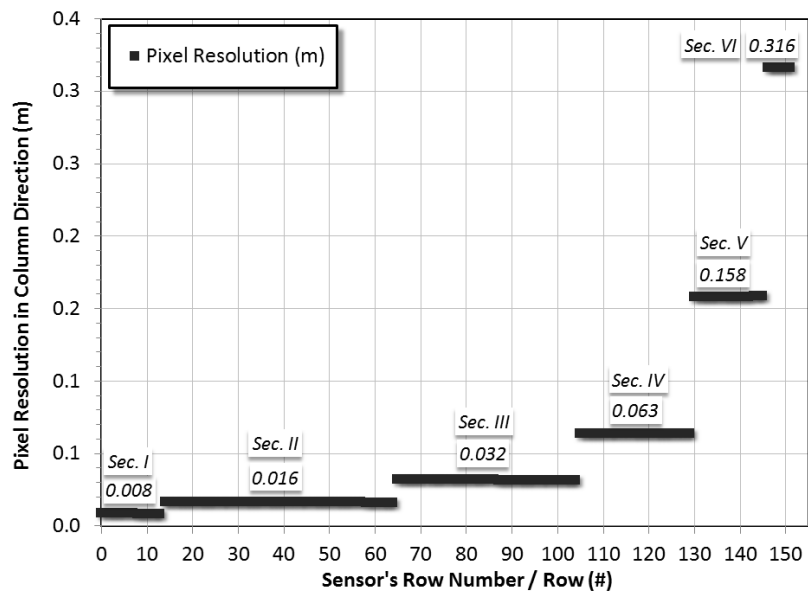
Figure 7-26. Pixels' resolutions of TZOID sensor as mounting at 1.4m high and tilting at 24 degree with 4.2mm focus-length lens.

Figure 7-27. Two-lane straight road model with model vehicle for TZOID image sensor testing**.**

A two-lane straight road model was used for TZOID image sensor testing as shown in Figure 7-27. The model vehicle on the model road was pulled by a step motor at certain speed controlled by a microcontroller, which makes test setup more controllable and reliable.

The tilt angle calibration was tested as described in Sec.5.5.1. Figure 7-28(a) shows where both leading and rear edge of the vehicle were detected for computation. The computed tilt angle and percentage of accuracy is listed in Figure 7-28(b).
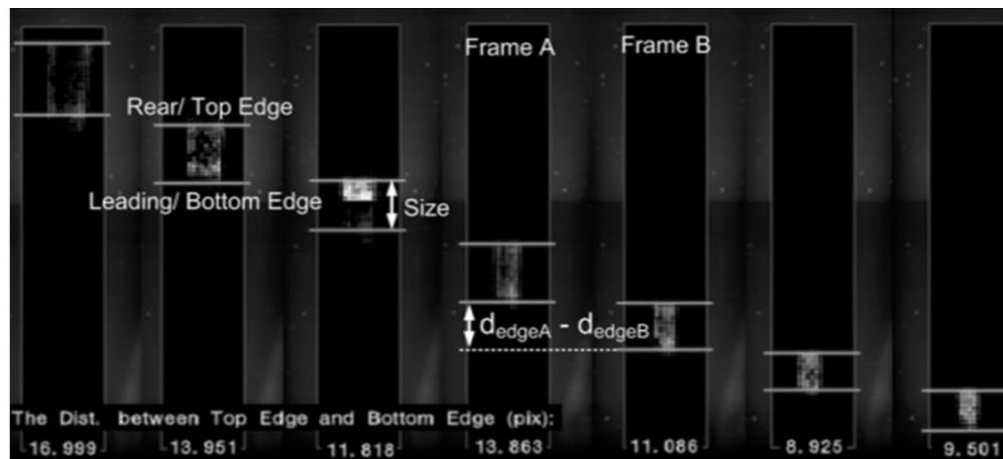
(a)

Computed Tilt Angle (degree)	23.74
Truth Value (degree)	24.00
Error (%)	1.08
Calibration Accuracy (%)	98.92

(b)

Figure 7-28. Two-lane straight road model test: (a) Example frames for leading edge and rear edge detection; (b) computed tilt angle.

An off-the-shelf image sensor (OV7670) was used, which has 1/6" optical size including 640x480 numbers of pixels, to compare the performance of TZOID image sensor using road model. Both OV7670 and TZOID cameras were mounted at 0.6m high with 17.5 degree of tilt angle, running at 30fps with the same lens, to achieve a detection region which is 1.28m to 4.29m away from camera setups. The speed detection procedures implemented in Section 7.1.1.2.2 were used on both sensors to obtain speed test result respectively. The comparison is started from the number of pixel rows and pixels' spatial resolution in the same

detection territory, which is shown in Figure 7-29 including three resolution sections of TZOID. Totally 480 pixel rows of OV7670 were focused on the detection region, whereas only 79 pixel rows of TZOID were for the same area, which was 6 times less. The average numbers of pixel resolution in this region of OV7670 and TZOID are 0.0063 meter per pixel and 0.038 meter per pixel, respectively.
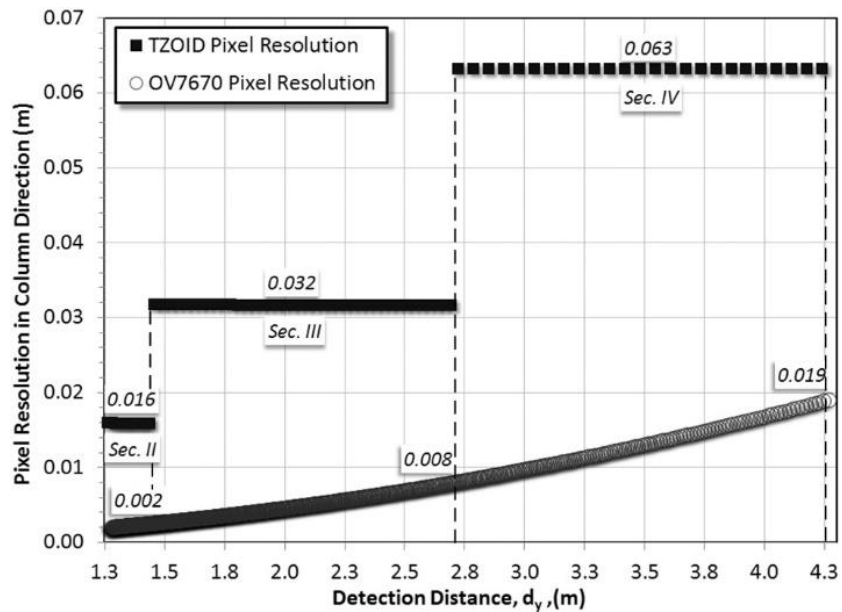
Figure 7-29. Pixel resolutions of TZOID and OV7670 sensor versus detection distance (1.28m-4.29m).

Testing results for speed detection by both sensors are listed in Table 7-7 and Table 7-8. The average detecting accuracy of TZOID is 97.37%, which is 1.79% lower than OV7670. The comparison result between two sensors are listed in Table 7-9, which illustrates that OV7670 does have around 2% better detection accuracy than TZOID does during the same detection territory by loading 6 times more pixel rows. On the other hand, TZOID trades off 2% accuracy for total 98.6% less image data, which means 98.6% less communication bandwidth and 80 times faster processing speed.

Table 7-7. Extracted speed of vehicle by TZOID sensor using P-RM method.

Trial	Frame #	Leading edge (pix)	Leading edge (m)	Actual Speed (m/s)	Computed speed (m/s)	Computed error	Accuracy
1-1	1219	24.81	4.181	22.26	/	/	/
1-2	1819	30.50	3.722	22.26	22.97	3.09%	96.91%
1-3	2419	37.39	3.261	22.26	23.04	3.39%	96.61%
1-4	3019	45.51	2.802	22.26	22.96	3.05%	96.95%
1-5	3619	57.53	2.346	22.26	22.79	2.33%	97.67%
1-6	4219	70.94	1.888	22.26	22.89	2.75%	97.25%
1-7	4819	86.30	1.434	22.26	22.71	1.98%	98.02%
2-1	607	19.62	4.619	31.03	/	/	/
2-2	1207	26.68	3.976	31.03	32.14	3.45%	96.55%
2-3	1807	36.05	3.334	31.03	32.11	3.36%	96.64%
2-4	2407	46.51	2.694	31.03	32.01	3.06%	96.94%
2-5	3007	66.44	2.058	31.03	31.80	2.42%	97.58%
2-6	3607	86.78	1.425	31.03	31.65	1.96%	98.04%
3-1	611	20.07	4.447	40.96	/	/	/
3-2	1211	31.52	3.646	40.96	40.02	-2.35%	97.65%
3-3	1811	44.35	2.842	40.96	40.23	-1.81%	98.19%
3-4	2411	67.98	2.038	40.96	40.17	-1.97%	98.03%
3-5	23	20.52	4.461	45.20	/	/	/
4-1	623	33.51	3.527	45.20	46.69	3.19%	96.81%
4-2	1223	47.86	2.597	45.20	46.48	2.75%	97.25%
4-3	1823	77.33	1.678	45.20	45.96	1.65%	98.35%

Table 7-8. Extracted speed of vehicle by OV7670 sensor using P-RM method.

Trial	Frame #	Leading edge (pix)	Leading edge (m)	Actual Speed (m/s)	Computed speed (m/s)	Computed error	Accuracy
1-1	33	16.10	3.9670	22.26	/	/	/
1-2	633	47.60	3.5150	22.26	22.60	1.53%	98.47%
1-3	1233	87.04	3.0636	22.26	22.57	1.39%	98.61%
1-4	1833	138.74	2.6130	22.26	22.53	1.21%	98.79%
1-5	2433	210.92	2.1643	22.26	22.44	0.79%	99.21%
1-6	3033	315.96	1.7170	22.26	22.37	0.47%	99.53%
2-1	5	18.02	3.9344	31.03	/	/	/
2-2	605	65.08	3.3067	31.03	31.3835001	1.14%	98.86%
2-3	1205	130.54	2.6788	31.03	31.395	1.18%	98.82%
2-4	1805	232.26	2.0538	31.03	31.25	0.71%	99.29%
2-5	2405	414.44	1.4315	31.03	31.115	0.27%	99.73%
3-1	13	14.26	4.0002	40.96	/	/	/
3-2	613	76.7	3.1678	40.96	41.62	1.61%	98.39%
3-3	1213	178.14	2.3442	40.96	41.18	0.54%	99.46%
3-4	1813	378.8	1.5231	40.96	41.05	0.23%	99.77%
4-1	14	24.00	3.8395	45.2	/	/	/
4-2	614	100.14	2.9306	45.2	45.44	0.53%	99.47%
4-3	1214	238.28	2.0249	45.2	45.29	0.19%	99.81%

Table 7-9. Comparison results between TZOID and OV7670.

Sensor	OV7670	TZOID	TZOID *vs* OV7670
Accuracy (Sec. IV) (%)	98.58	96.84	-1.77%
Accuracy (Sec. III) (%)	99.45	97.68	-1.78%
Pixel Rows (Sec. IV)	117	19	-83.76%
Pixel Rows (Sec. III)	291	33	-88.66%
Average Accuracy (%)	99.16	97.37	-1.81%
Total Pixel Rows	480	79	-83.54%

7.3   Summary

In this chapter, testing and measurement results for P-RM method and TZOID image sensor were presented. First, the testing results from P-RM method implementing on commercial camera are introduced, which includes indoor lab testing, outdoor vehicle detection and outdoor pedestrian height estimation. Second, TZOID imager testing results including comparison results with standard rectangular imager, OV7670, were presented.

CHAPTER 8   - CONCLUSION

This research work develops a multi-resolution CMOS active pixel sensor (APS) imager with a straightforward calibration method and feature extraction capability for traffic detection application. The prototype of a custom CMOS image sensor with specifically designed pixel array for traffic detection applications with innovative mapping algorithms and intelligent feature extraction methods are accomplished.

The research is divided into five research objectives. The first objective is the investigation of fundamental principles, limits, and problems associated with vision-based traffic detection. The second objective is the investigation and development of camera model to achieve a specific model (pixel-resolution model) for pixel's physical size and related pixel's spatial resolution. The third objective is to design and fabricate a TZOID CMOS imager based on pixel-resolution model for traffic applications. The fourth objective is to develop feature extraction algorithms for designed image sensor with straightforward camera calibration algorithms based on pixel-resolution model. The fifth objective is to develop a TZOID CMOS image sensor based vision system which is suitable for traffic detection with high accuracy and low data rate.

The fundamental principles, limits, and problems associated with vision-based traffic detection are investigated resulting in a design strategy including both sensor design and algorithm design. The background and development of CMOS image sensor are analyzed followed by a literature review on current camera calibration methods used in traffic detection. The main drawbacks for those current methods are impractical and consuming high computation because of iteration procedures, requiring specific patterns, and requiring multiple cameras.

A modified camera model is developed to achieve a specific model (pixel-resolution model) for pixel's physical size and related pixel's spatial resolution. This model gives a new way to generate the relationship between 2D image and related 3D scene based on spatial resolution pixels in CMOS image sensor arrays. The proposed PR-M method and related feature extraction algorithms including height estimation are based on this pixel-resolution model. The PR-M method achieves height estimation without using specific

patterns in real world or multiple (or moving) cameras with straightforward camera calibration algorithms. A commercial camera is used to verify the efficiency of the PR-M method. Both tilt angle and mounting height of the camera were calibrated correctly in lab and outdoor experiments. Three-dimensional mensuration tests are applied, which show better than 98.7% accuracy in lab tests and better than 96% accuracy in actual pedestrian height estimation tests. Additionally, reconstructed 3D image of detected points is successively obtained by in this research work.

A TZOID CMOS image sensor is designed and fabricated based on pixel-resolution model for traffic applications. Each pixel's size was determined and designed according to desired spatial resolution to achieve less unimportant data with enough detection accuracy. The abundant data could be alleviated by reducing the number of redundant pixels without losing unacceptable detecting accuracy. A TZOID CMOS image sensor based vision system applying on traffic detection is generated which was tested in the laboratory. After comparing with a commercial counterpart running at the same frame rate, TZOID generates 98% less data but only 2% detection accuracy lost.

The overall outcome of this research is the design and implementation of the world's first trapezoid pixel array multi-resolution CMOS image sensor with flexible camera calibration technique and accurate feature extraction algorithms based on straightforward P-RM method with pixel-resolution model.

8.1   Future Directions

There are third directions of future work for TZOID image sensor design.

The first is integrating the motion detection and edge detection ability in the image sensor chip to achieve less data, less communication bandwidth and faster speed.

The second research direction is implanting pixel merging function to the next generation of TZOID pixel array, the pixel size of which could be adjusted according to actual mounting situation to keep pixels' spatial resolutions in the desired resolution zones.

The third research direction is designing a 3D camera system by using single view camera that can generate 3D reconstruct scene, especially for vehicle, and pedestrian, in real time.

REFERENCES

[1] R. M. (Mitron S. C. Tyburski, "A Review of Road Sensor Technology for Monitoring Vehicle Traffic," *ITE Inst. Transp. Eng. J. USA*, vol. 59:8, Aug. 1988.

[2] S. E. Underwood, "A REVIEW AND CLASSIFICATION OF SENSORS FOR INTELLIGENT VEHICLE- HIGHWAY SYSTEMS," *IVHS Tech. Rep.*, 1990.

[3] B. Edde, *Radar : principles, technology, applications /.* Englewood Cliffs, N.J. : Prentice Hall, c1993.

[4] E. R. Fossum, "Active Pixel Sensors -- Are CCDs Dinosaurs?," in *SPIE*, 1993, vol. 1900, pp. 2–14.

[5] E. R. Fossum, "CMOS image sensors: electronic camera on a chip," presented at the Electron Devices Meeting, 1995. IEDM '95., International, 1995, pp. 17–25.

[6] C. Setchell and E. L. Dagless, "Vision-based road-traffic monitoring sensor," *IEE Proc. - Vis. Image Signal Process.*, vol. 148, no. 1, pp. 78–84, Feb. 2001.

[7] C. C. C. Pang, S. S. Xie, S. C. Wong, and K. Choi, "Generalized camera calibration model for trapezoidal patterns on the road," *Opt. Eng.*, vol. 52, no. 1, pp. 017006–017006, 2013.

[8] G. Wang, D. Xiao, and J. Gu, "Review on vehicle detection based on video for traffic surveillance," in *2008 IEEE International Conference on Automation and Logistics*, 2008, pp. 2961–2966.

[9] L. A. Klein, M. K. Mills, and D. R. P. Gibson, "Traffic Detector Handbook: Third Edition—Volume II," FHWA-HRT-06-139, Oct. 2006.

[10] J. C. T. Diaz, Q. Houben, J. Czyz, and O. Debeir, "A camera auto-calibration algorithm for real time road traffic analysis," presented at the Conference on Computer Vision Theory and Applications (VISAPP), Lisboa, Portugal, 2009, vol. 1, pp. 626–631.

[11] J. Wu, Z. Liu, J. Li, C. Gu, M. Si, and F. Tan, "An algorithm for automatic vehicle speed detection using video camera," in *2009 4th International Conference on Computer Science Education*, 2009, pp. 193–196.

[12] N. K. Kanhere and S. T. Birchfield, "Real-Time Incremental Segmentation and Tracking of Vehicles at Low Camera Angles Using Stable Features," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 148–160, Mar. 2008.

[13] G. S. K. Fung, N. H. C. Yung, and G. K. H. Pang, "Camera calibration from road lane markings," *Opt. Eng.*, vol. 42, no. 10, pp. 2967–2977, 2003.

[14] X. C. He and N. H. C. Yung, "New method for overcoming ill-conditioning in vanishing-point-based camera calibration.pdf," *Opt. Eng.*, vol. 46, 2007.

[15] S.-Y. Ma and L.-G. Chen, "A single-chip CMOS APS camera with direct frame difference output," *IEEE J. Solid-State Circuits*, vol. 34, no. 10, pp. 1415–1418, Oct. 1999.

[16] E. Artyomov and O. Yadid-Pecht, "Adaptive Multiple-Resolution CMOS Active Pixel Sensor," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 53, no. 10, pp. 2178–2186, Oct. 2006.

[17] J. Choi, S. W. Han, S. J. Kim, S. I. Chang, and E. Yoon, "A Spatial-Temporal Multiresolution CMOS Image Sensor With Adaptive Frame Rates for Tracking the Moving Objects in Region-of-Interest and Suppressing Motion Blur," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2978–2989, Dec. 2007.

[18]   B. Zhao, X. Zhang, S. Chen, K. S. Low, and H. Zhuang, "A 64 64 CMOS Image Sensor With On-Chip Moving Object Detection and Localization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 4, pp. 581–588, Apr. 2012.

[19]   I. Fukui, "TV image processing to determine the position of a robot vehicle," *Pattern Recognit.*, vol. 14, no. 1, pp. 101–109, Jan. 1981.

[20]   R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robot. Autom.*, vol. 3, no. 4, pp. 323–344, Aug. 1987.

[21]   Z. Zhang, "A Flexible New Technique for Camera Calibration," MSR-TR-98-71, Dec. 1998.

[22]   Y. Zhao, X. Li, and W. Li, "Binocular vision system calibration based on a one-dimensional target," *Appl. Opt.*, vol. 51, no. 16, pp. 3338–3345, Jun. 2012.

[23]   P. Gurdjos, J.-S. Kim, and I.-S. Kweon, "Euclidean Structure from Confocal Conics: Theory and Application to Camera Calibration," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006, vol. 1, pp. 1214–12221.

[24]   E. K. Bas and J. D. Crisman, "An easy to install camera calibration for traffic monitoring," in *Proceedings of Conference on Intelligent Transportation Systems*, 1997, pp. 362–366.

[25]   Y. Li, J. Zhang, W. Hu, and J. Tian, "Method for pan-tilt camera calibration using single control point," *JOSA A*, vol. 32, no. 1, pp. 156–163, Jan. 2015.

[26]   R. I. Hartley, "An algorithm for self calibration from several views," in *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 908–912.

[27]   S. D. Ma, "A self-calibration technique for active vision systems," *IEEE Trans. Robot. Autom.*, vol. 12, no. 1, pp. 114–120, Feb. 1996.

[28]   F. Rameau, A. Habed, C. Demonceaux, D. Sidibé, and D. Fofi, "Self-calibration of a PTZ Camera Using New LMI Constraints," in *Computer Vision – ACCV 2012*, 2012, pp. 297–308.

[29]   O. D. Faugeras, Q.-T. Luong, and S. J. Maybank, "Camera self-calibration: Theory and experiments," in *European conference on computer vision*, 1992, pp. 321–334.

[30]   S. J. Maybank and O. D. Faugeras, "A Theory of Self-calibration of a Moving Camera," *Int J Comput Vis.*, vol. 8, no. 2, pp. 123–151, Aug. 1992.

[31]   B. Wu, A. Abdel-Rahim, and S. U. Ay, "A trapezoid multiresolution CMOS image sensor for traffic monitoring," in *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2016, pp. 1–4.

[32]   B. Wu, A. Abdel-Rahim, and S. U. Ay, "A trapezoid CMOS image sensor with 2% detection accuracy for traffic monitoring," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2017, pp. 1154–1158.

[33]   D. Middleton, R. Parker, and R. Longmire, "Investigation of Vehicle Detector Performance and ATMS Interface," Texas Transportation Institute - The Texas A&M University, College Station, Texas, FHWA/TX-07/0-4750-2, Mar. 2007.

[34]   R. M. (Mitron S. C. Tyburski, "A Review of Road Sensor Technology for Monitoring Vehicle Traffic," *ITE Inst. Transp. Eng. J. USA*, vol. 59:8, Aug. 1988.

[35]   E. Rocah and R. Palacios, "Image-processing algorithms for detecting and counting vehicles waiting at a traffic light.pdf," *J. Electron. Imaging*, vol. 19, Dec. 2010.

[36]   S. Hockaday, "Evaluation of image processing technology for applications in highway operations: final report," no. TR 91-2, Jun. 1991.

[37]   R. C. Gonzalez and R. E. Woods, *Digital image processing*. Upper Saddle River, N.J. : Prentice Hall, 2012.

[38]   N. Ghosh and B. Bhanu, "Incremental Vehicle 3-D Modeling from Video," in *18th International Conference on Pattern Recognition (ICPR'06)*, 2006, vol. 3, pp. 272–275.

[39]   W. Ouyang, X. Zeng, and X. Wang, "Single-Pedestrian Detection Aided by Two-Pedestrian Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1875–1889, Sep. 2015.

[40]   L. Xie, G. Zhu, Y. Wang, H. Xu, and Z. Zhang, "Robust vehicles extraction in a video-based intelligent transportation systems," in *Proceedings. 2005 International Conference on Communications, Circuits and Systems, 2005.*, 2005, vol. 2, p. 890.

[41]   Y.-K. Wang and S.-H. Chen, "A robust vehicle detection approach," in *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005.*, 2005, pp. 117–122.

[42]   J. Heikkila, "Geometric camera calibration using circular control points," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1066–1077, Oct. 2000.

[43]   M. Piccardi, "Background subtraction techniques: a review," in *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, 2004, vol. 4, pp. 3099–3104 vol.4.

[44]   C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, Jul. 1997.

[45]   I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 809–830, Aug. 2000.

[46]   Y. Soh, Y. Hae, and I. Kim, "Spatio-temporal Gaussian Mixture Model for Background Modeling," in *2012 IEEE International Symposium on Multimedia*, 2012, pp. 360–363.

[47]   M. Boninsegna and A. Bozzoli, "A tunable algorithm to update a reference image," *Signal Process. Image Commun.*, vol. 16, no. 4, pp. 353–365, Nov. 2000.

[48]   "Detecting moving objects, ghosts, and shadows in video streams - IEEE Journals & Magazine." [Online]. Available: https://ieeexplore.ieee.org/document/1233909/. [Accessed: 10-Apr-2018].

[49]   A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *European conference on computer vision*, 2000, pp. 751–767.

[50]   D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[51]   A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proc. IEEE*, vol. 90, no. 7, pp. 1151–1163, Jul. 2002.

[52]   Y. Boykov and O. Veksler, "Graph Cuts in Vision and Graphics: Theories and Applications," 2017, pp. 100–118.

[53]   D.-W. Lim, S.-H. Choi, and J.-S. Jun, "Automated detection of all kinds of violations at a street intersection using real time individual vehicle tracking," in *Proceedings Fifth IEEE Southwest Symposium on Image Analysis and Interpretation*, 2002, pp. 126–129.

[54] L. W. Tsai, J. W. Hsieh, and K. C. Fan, "Vehicle Detection Using Normalized Color and Edge Map," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 850–864, Mar. 2007.

[55] H. Xua, X. Xia, L. Guo, W. Chen, and G. Huang, "A Novel Algorithm of Moving Cast Shadow Suppression," in *2006 8th international Conference on Signal Processing*, 2006, vol. 2.

[56] J. L. Barron, D. J. Fleet, S. S. Beauchemin, and T. A. Burkitt, "Performance of optical flow techniques," in *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1992, pp. 236–242.

[57] J. Nakamura, *Image Sensors and Signal Processing for Digital Still Cameras.pdf*. Taylor & Francis Group, 2006.

[58] G. P. Weckler, "Operation of p-n Junction Photodetectors in a Photon Flux Integrating Mode," *IEEE J. Solid-State Circuits*, vol. 2, no. 3, pp. 65–73, Sep. 1967.

[59] R. H. Dyck and G. P. Weckler, "Integrated arrays of silicon photodetectors for image sensing," *IEEE Trans. Electron Devices*, vol. 15, no. 4, pp. 196–201, Apr. 1968.

[60] P. J. W. Noble, "Self-scanned silicon image detector arrays," *IEEE Trans. Electron Devices*, vol. 15, no. 4, pp. 202–209, Apr. 1968.

[61] S. G. Chamberlain, "Photosensitivity and Scanning of Silicon Image Detector Arrays," *IEEE J. Solid-State Circuits*, vol. 4, no. 6, pp. 333–342, Dec. 1969.

[62] P. W. Fry, P. J. W. Noble, and R. J. Rycroft, "Fixed-pattern noise in photomatrices," *IEEE J. Solid-State Circuits*, vol. 5, no. 5, pp. 250–254, Oct. 1970.

[63] W. S. Boyle and G. E. Smith, "Charge Coupled Semiconductor Devices," *Bell Syst. Tech. J.*, vol. 49, no. 4, pp. 587–593, 1970.

[64] A. Krymski, D. Van Blerkom, A. Andersson, N. Bock, B. Mansoorian, and E. R. Fossum, "A high speed, 500 frames/s, 1024x1024 CMOS active pixel sensor," in *1999 Symposium on VLSI Circuits, 1999. Digest of Technical Papers*, 1999, pp. 137–138.

[65] E. R. Fossum, "Ultra low power imaging systems using CMOS image sensor technology," *Adv. Microdevices Space Sci. Sens. Proc SPIE*, vol. 2267, pp. 107–111, 1994.

[66] R. H. Nixon, S. E. Kemeny, C. O. Staller, and E. R. Fossum, "128x128 CMOS photodiode-type active pixel sensor with on-chip timing, control and signal chain electronics," in *Proc. SPIE Charge-Coupled Devices and Solid-State Optical Sensors V*, 1995, vol. 2415, pp. 117–123.

[67] S. K. Mendis, S. E. Kemeny, and E. R. Fossum, "A 128×128 CMOS active pixel image sensor for highly integrated imaging systems," in *IEDM '93. Technical Digest., International*, 1993, pp. 583–586.

[68] S. Mendis, S. E. Kemeny, and E. R. Fossum, "CMOS active pixel image sensor," *IEEE Trans. Electron Devices*, vol. 41, no. 3, pp. 452–453, Mar. 1994.

[69] S. K. Mendis *et al.*, "CMOS active pixel image sensors for highly integrated imaging systems," *IEEE J. Solid-State Circuits*, vol. 32, no. 2, pp. 187–197, Feb. 1997.

[70] S. Yoshihara *et al.*, "A 1/1.8-inch 6.4 MPixel 60 frames/s CMOS Image Sensor With Seamless Mode Change," *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2998–3006, Dec. 2006.

[71] A. I. Krymski, N. E. Bock, N. Tu, D. V. Blerkom, and E. R. Fossum, "A high-speed, 240-frames/s, 4.1-Mpixel CMOS sensor," *IEEE Trans. Electron Devices*, vol. 50, no. 1, pp. 130–135, Jan. 2003.

[72]   M. S. Shin, J. B. Kim, M. K. Kim, Y. R. Jo, and O. K. Kwon, "A 1.92-Megapixel CMOS Image Sensor With Column-Parallel Low-Power and Area-Efficient SA-ADCs," *IEEE Trans. Electron Devices*, vol. 59, no. 6, pp. 1693–1700, Jun. 2012.

[73]   D. Johns and K. W. Martin, *Analog integrated circuit design*. John Wiley & Sons, 1997.

[74]   D. C. Brown, "Close-range camera calibration," *Photogramm. Eng.*, vol. 37, no. 8, pp. 855–866, 1971.

[75]   W. Faig, "CALIBRATION OF CLOSE-RANGE PHOTOGRAMMETRIC SYSTEMS: MATHEMATICAL FORMULATION," *Photogramm. Eng. Remote Sens.*, vol. 41, no. 12, Dec. 1975.

[76]   G. Q. Wei and S. D. Ma, "A complete two-plane camera calibration method and experimental comparisons," in *1993 (4th) International Conference on Computer Vision*, 1993, pp. 439–446.

[77]   Y. Zhang, L. Zhou, H. Liu, and Y. Shang, "A Flexible Online Camera Calibration Using Line Segments," *J. Sens.*, p. 16, 2016.

[78]   J. Jung, I. Yoon, S. Lee, and J. Paik, "Object Detection and Tracking-Based Camera Calibration for Normalized Human Height Estimation," *J. Sens.*, 2016.

[79]   P. K. Sinha, *Image acquisition and preprocessing for machine vision systems*. Society of Photo-Optical Instrumentation Engineers, 2012.

[80]   W. James and E. K. Suckiel, *Some Problems of Philosophy: A Beginning of an Introduction to Philosophy*. Lincoln: University of Nebraska Press, 1996.

[81]   R. M. Haralick, "Using Perspective Transformations in Scene Analysis," *Comput. Graph. Image Process.*, vol. 13, pp. 191–221, 1980.

[82]   J. Xue, X. Su, L. Xiang, and W. Chen, "Using concentric circles and wedge grating for camera calibration," *Appl. Opt.*, vol. 51, no. 17, pp. 3811–3816, Jun. 2012.

[83]   A. Arfaoui and S. Thibault, "Fisheye lens calibration using virtual grid," *Appl. Opt.*, vol. 52, no. 12, pp. 2577–2583, Apr. 2013.

[84]   H.-L. Chou and W.-H. Tsai, "A new approach to robot location by house corners," *Pattern Recognit.*, vol. 19, no. 6, pp. 439–451, Jan. 1986.

[85]   L. L. Wang and W.-H. Tsai, "Camera calibration by vanishing lines for 3-D computer vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 4, pp. 370–376, Apr. 1991.

[86]   E. K. Bas and J. D. Crisman, "An easy to install camera calibration for traffic monitoring," in *Proceedings of Conference on Intelligent Transportation Systems*, 1997, pp. 362–366.

[87]   S. W. Park, T. E. Kim, and J. S. Choi, "Real-Time Estimation of Trajectories and Heights of Pedestrians," in *2011 International Conference on Information Science and Applications*, 2011, pp. 1–8.

[88]   T. N. Schoepflin and D. J. Dailey, "Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 2, pp. 90–98, Jun. 2003.

[89]   L. Wang, S. B. Kang, H.-Y. Shum, and G. Xu, "Error analysis of pure rotation-based self-calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 275–280, Feb. 2004.

[90]   B. Zhao and Z. Hu, "Camera self-calibration from translation by referring to a known camera," *Appl. Opt.*, vol. 54, no. 25, pp. 7789–7798, Sep. 2015.

[91]   M. Pollefeys, L. V. Gool, and A. Oosterlinck, "The modulus constraint: a new constraint self-calibration," in *Proceedings of 13th International Conference on Pattern Recognition*, 1996, vol. 1, pp. 349–353 vol.1.

[92]   M. Pollefeys and L. van Gool, "Stratified self-calibration with the modulus constraint," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 707–724, Aug. 1999.

[93]   L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: establishing a common coordinate frame," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 758–767, Aug. 2000.

[94]   S. Khan and M. Shah, "Consistent labeling of tracked objects in multiple cameras with overlapping fields of view," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1355–1360, Oct. 2003.

[95]   "Traffic Detector Handbook: Third Edition—Volume I," Oct. 2006.

[96]   L. A. Klein, M. K. Mills, and D. R. P. Gibson, "Traffic Detector Handbook," USDOT, 3rd.Ed. FHWA-HRT-06-139, Oct. 2006.

[97]   B. Wu, A. Abdel-Rahim, and S. U. Ay, "A trapezoid multiresolution CMOS image sensor for traffic monitoring," in *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2016, pp. 1–4.

[98]   N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an image edge detection filter using the Sobel operator," *IEEE J. Solid-State Circuits*, vol. 23, no. 2, pp. 358–367, Apr. 1988.