VISION BASED CALIBRATION AND LOCALIZATION TECHNIQUE FOR VIDEO SENSOR NETWORKS

SHARIF AMAR MOHAMED SHARIF

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING
UNIVERSITI TEKNOLOGI PETRONAS
MARCH 2009

STATUS OF THESIS

Title of thesis	Vision Based Calibration and Localization Technique for Video Sensor Networks

I, SHARIF AMAR MOHAMED SHARIF,

hereby allow my thesis to be placed at the Information Resources Center (IRC) of Universiti Teknologi PETRONAS (UTP) with the following conditions:

1. The thesis becomes the property of UTP
2. The IRC of UTP may make copies of the thesis for academic purposes only.
3. This thesis is classified as

□ Confidential

☑ Non-confidential

If this thesis is confidential, please state the reason:

____

The contents of the thesis will remain confidential for ____________ years.

Remarks on disclosure:

____

Endorsed by

Signature of Author	Signature of Supervisor
SHARIF AMAR MOHAMED	VARUN JEOTI
Date: 11/32009	Date: 11/3/2009

UNIVERSITI TEKNOLOGI PETRONAS

Approval by Supervisor (s)

The undersigned certify that they have read, and recommend to the Postgraduate Studies Programme for acceptance, a thesis entitled "**Vision Based Calibration and Localization Technique for Video Sensor Networks**" submitted by (**Sharif Amar Mohamed Sharif**) for the fulfillment of the requirements for the degree of Master of Science in Electrical and Electronics Engineering.

10/3/2009

Date

Signature :

Main Supervisor : Varun Jeoti

Date : 10/3/2009

UNIVERSITI TEKNOLOGI PETRONAS

Vision Based Calibration and Localization Technique for Video Sensor Networks

By

Sharif Amar Mohamed Sharif

A THESIS

SUBMITED TO THE POSTGRADUATE STUDIES PROGRAMME AS
A REQUIREMENT FOR THE

DEGREE OF MASTER OF SCIENCE

IN ELECTRICAL AND ELECTRONICS ENGINEERING

BANDAR SERI ISKANDAR,

PERAK

March, 2009

I hereby declare that the thesis is based on my original work except for the quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTP or other institutions.

Signature : ______________________________

Name : SHARIF AMAR MOHAMED SHARIF

Date : 10/3/2009

ABSTRACT

The recent evolutions in embedded systems have now made the video sensor networks a reality. A video sensor network consists of a large number of low cost camera-sensors that are deployed in random manner. It pervades both the civilian and military fields with huge number of applications in various areas like health-care, environmental monitoring, surveillance and tracking. As most of the applications demand the knowledge of the sensor-locations and the network topology before proceeding with their tasks, especially those based on detecting events and reporting, the problem of localization and calibration assumes a significance far greater than most others in video sensor network. The literature is replete with many localization and calibration algorithms that basically rely on some a-priori chosen nodes, called *seeds*, with known coordinates to help determine the network topology. Some of these algorithms require additional hardware, like arrays of antenna, while others require having to regularly reacquire synchronization among the *seeds* so as to calculate the time difference of the received signals. Very few of these localization algorithms use vision based technique.

In this work, a vision based technique is proposed for localizing and configuring the camera nodes in video wireless sensor networks. The camera network is assumed randomly deployed. One a-priori selected node chooses to act as the core of the network and starts to locate some other two reference nodes. These three nodes, in turn, participate in locating the entire network using tri-lateration method with some appropriate vision characteristics. In this work, the vision characteristics that are used the relationship between the height of the image in the image plane and the real distance between the sensor node and the camera. Many experiments have been simulated to demonstrate the feasibility of the proposed technique. Apart from this work, experiments are also carried out to locate any other new object in the video sensor network.

The experimental results showcase the accuracy of building up one-plane network topology in relative coordinate system and also the robustness of the technique against the accumulated error in configuring the whole network.

ABSTRAK

Evolusi terkini dalam sistem terbenam telah menjadikan rangkaian penderia video satu kenyataan. Rangkaian penderia video terdiri daripada sebilangan besar penderia-kamera berkos rendah yang digunakan dalam keadaan rawak. Ianya menyerang kedua-dua bidang ketenteraan dan orang awan dengan bilangan besar aplikasinya dalam pelbagai bidang seperti penjagaan kesihatan, pengawasan persekitaran, pengawasan dan penjejakan. Kebanyakan aplikasi menuntut pengetahuan tentang lokasi-penderia dan topologi rangkaian sebelum meneruskan tugas-tugas mereka, terutama yang berkaitan dengan hal-hal mengesan dan melapor, masalah penempatan dan penentukuran menganggap satu kelebihan jauh lebih bagus berbanding dengan kebanyakan rangkaian penderia video yang lain. Literatur dipenuhi dengan banyak penempatan dan algoritma penentukuran yang pada asasnya bergantung kepada sebilangan priori nod-nod terpilih, yang dipanggil *biji-biji*, dengan koordinat yang diketahui untuk membantu dalam menentukan topologi rangkaian. Sesetengath algoritma ini memerlukan perkakasan tambahan, seperti tatasusunan antena, manakala yang lain perlu secara teratur mendapatkan semula penyegerakan antara biji-biji agar dapat menghitung perbezaan masa isyarat yang diterima. Hanya sebilangan kecil algoritma penempatan menggunakan teknik berasaskan penglihatan.

Dalam kajian ini, teknik berasaskan penglihatan dicadangkan untuk menempat dan mengatur nod kamera dalam rangkaian penderia wayerless video. Rangkaian kamera diandaikan diatur secara rawak. Satu priori nod terpilih memilih untuk bertindak sebagai teras rangkaian dan mula mengatur beberapa dua nod rujukan yang lain. Ketiga- tiga nod ini, bergilir-gilir menyertai dalam menempatkan keseluruhan rangkaian menggunakan kaedah 'tri-lateration' dengan beberapa cirri-ciri penglihatan yang sesuai. Banyak eksperimen telah disimulasi untuk menunjukkan kemungkinan terhadap teknik yang dicadangkan. Sebahagian daripada kajian ini, eksperimen juga turut dijalankan bagi menempatkan mana-mana objek baru yang lain dalam rangkaian penderia video.

Hasil kajian ini menunjukkan kejituan membangunkan topologi rangkaian satu satah dengan merujuk kepada sistem koordinat dan juga keteguhan teknik pada ralat terkumpul dalam mengatur keseluruhan rangkaian.

ACKNOWLEDGMENTS

First of all, I would like to express my greatest thankful to Allah (*Subhanahu Wa Ta'ala*) for his uncountable blessings and for giving me the strength to success and finish my research.

Also, I would like to express my deepest sense of gratitude to my supervisor A.P. Dr. Varun Jeoti for his encouragement, valuable advices and his constant assistant throughout my study. Without his brilliant supervision, this work will be impossible and I will be grateful, forever, his greatest influence in my attitude and the way of understanding the life.

Also, I'm deeply and forever indebted to my beloved parents for their continuous help, consistent support, and encouragement during this period with special thanks goes to my beloved lady Nihad Musbah for her motivation and moral support in assisting my research.

My sincere thanks goes to all my friends and colleagues in Electrical and Electronics department in Universiti Teknologi PETRONAS for sharing experiences and knowledge during the time of study.

Sharif Amar Mohamed Sharif

Universiti Teknologi PETRONAS, March 2009

Table of Contents

CHAPTER THREE: PROPOSED VISION BASED TECHNIQUE

CHAPTER FOUR: METHODOLOGY - EXPERIMENTAL TESTING

CHAPTER FIVE: ANALYSIS AND DISCUSSION

CHAPTER SIX: CONLCUSIONS AND FUTURE WORKS

LIST OF FIGURES

List of Tables

CHAPTER ONE: INTRODUCTION

In this chapter, the wireless sensor network (WSN), leading to video wireless sensor network (VWSN), interchangeably called camera sensor network is introduced. With the background of VWSN and its applications covered will be dealt with challenges of designing and deploying typical VWSN. The node localization and object tracking as the two challenges are identified and hence developed in this thesis.

1.1 Wireless Sensor Network

A WSN is a network that consists of a large number of low cost nodes that are randomly distributed in an ad-hoc manner as shown in Figure 1-1 and cooperatively perform specific functions. Each node has the capability of communicating with other nodes in the network (using short-range transceiver devices), sensing the surrounding physical phenomena, and carrying out a limited processing on the sensed data with the help of a microcontroller and some finite memory.

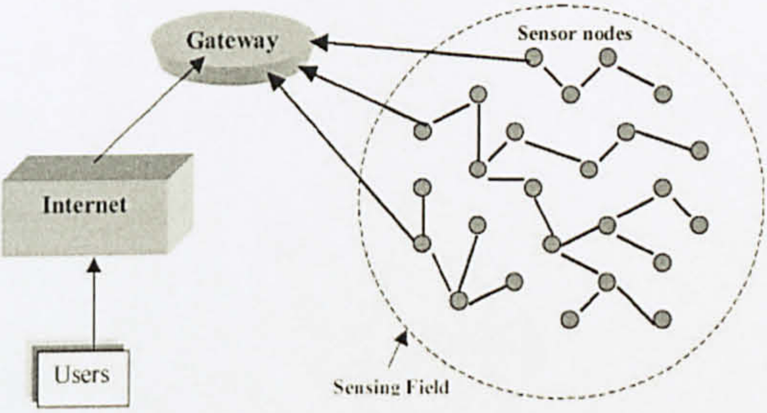


Figure 1-1: Architecture of Wireless Sensor Network [1]

These characteristics and many other unique features have given rise to WSNs being used in variety of applications from civilian to military and from home to industry. The development of WSNs was originally motivated by military applications such as battlefield surveillance and enemy monitoring [2]. However, they are now being used in many civilian applications. In the following section we will briefly mention a few of these applications.

1.2 Wireless Sensor Network Applications

As mentioned above, WSNs have come to control many aspects of our life as in the healthcare, environmental monitoring, surveillance and security. The following are some examples make use of sensor networks:

Habitat and Area Monitoring [3], is the most common application of WSN that is used for monitoring purposes. It represents significant advance over the traditional methods of monitoring where it negates many critical challenges facing researchers/designers, like, the potential impacts of human presence in the field conditions while monitoring plants and animals. Although choosing WSNs in this type of application allows the random deployment of the sensing devices saving on the cost of constructing such networks, it still requires a good localizing technique that makes it more practical.

HealthCare applications [4], another area of application of WSNs, are considered to be of great supplementary values for both medical and sensor networks fields. These kinds of applications require real-time response and accurate data aggregation. Examples of such applications are monitoring chronic patients and assisting elderly persons with full medical care for long period of time. Indeed the significance of these applications is in its ability to inform the medical professionals the latest condition of the patient and the location when emergency occurs.

Home automation and security [5], (also known as smart homes) is one of the WSN applications that are categorized under the automation techniques. Example of these applications are light and climate control, control of doors and window shutters, and monitoring and security systems. Nowadays, securing any residential and industry complex is also based on WSNs.

Surveillance and Tracking [6], is considered to be the most widely used type of WSNs applications. It has been motivated and adapted from the way the soldiers monitor battle-fields. As shown in Figure 1-2 soldiers can collect important data from the battle-field using WSNs.

Figure 1-2: Soldier Collecting Data from the Battle-Field [2]

Currently, there are many other applications of WSNs and there is no doubt that many of them involve some kind of event and data reporting. In a typical application, these data seem useless without the prior knowledge of the location. This requires that the researchers innovate and develop good localization techniques for sensor networks.

1.3 Video Wireless Sensor Networks:

The rapid evolution in the embedded systems technology today has provided the opportunity of using video and image data in the sensor networks. From this emerge various applications in VWSNs. This is a type of WSN that consists of low cost camera nodes. It affects the underlying infrastructure of the normal sensor networks and offers additional challenges, like:

- Power Challenges - due to the large computational requirements in the internal video processing.
- Bandwidth Challenges - due to the relatively big size of the resulting video data.
- Configuring Challenges - due to the extra parameters needed to configure the sensor nodes (*e.g.,* the orientation angle of the camera node and its field of view).

Unlike the normal sensor networks, video sensor networks provide additional visual information comprising of situation awareness, event understanding, and decision making. This makes the use of such networks in the surveillance, monitoring, and tracking applications more efficient. The next section presents a practical scenario that demonstrates these additional advantages of a VWSN.

1.4 Fire Rescue Application

In this section, a brief theoretical study for a practical scenario using video sensor networks namely Fire Rescue Application is presented. Recently, much research effort has been focused on Fire Rescue applications using sensor networks. One such study presented by *Kewei Sha et al* [7] addressed the requirements for Fire Rescue Application using sensor network. Practically and as shown in Figure 1-3, the fire-fighters can be considered as distributed sensor nodes in the fire field.

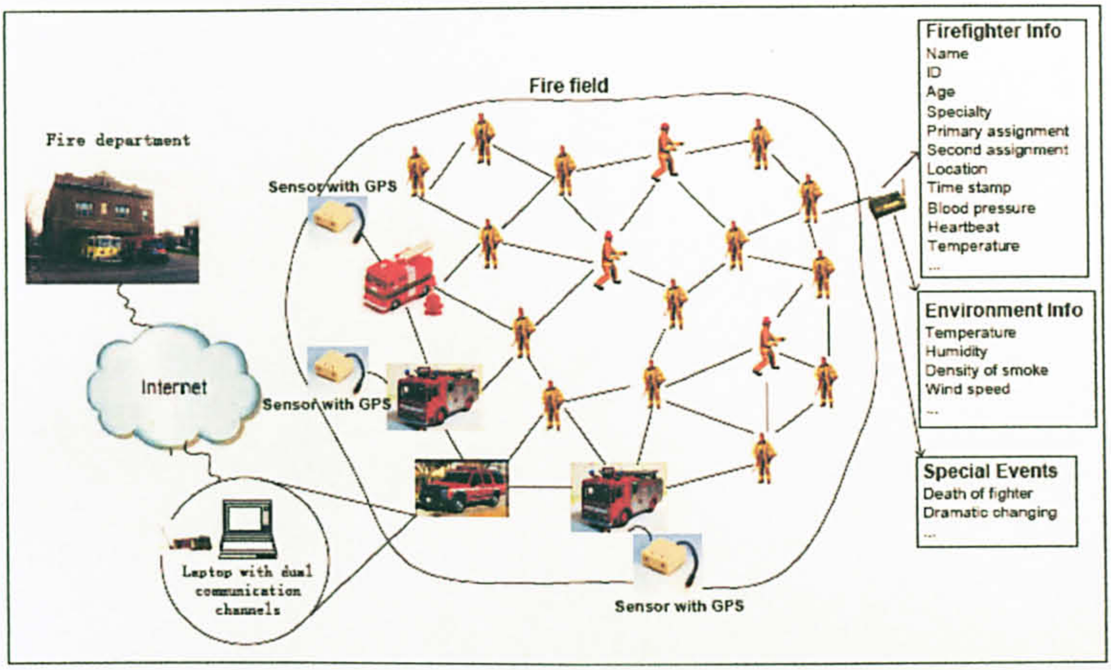


Figure 1-3: Architecture of Fire Net [7]

In some situations the sensing field could be unreachable (*e.g.,* big forests fires, fires in exposed areas where the winds keep changing the fires movement and other safety factors). Such cases can be handled by random deployment of a numbers of camera nodes in the fire fields that can help construct an underlying infrastructure of VWSN and help provide a complete view of what's going on in the fire fields. This network keeps gathering useful information from the fire fields to the fire department, so it can take effective decisions to blaze-out the fires in short time (*e.g.,* start blazing-out the fires from such and such direction, send more fire-fighters to certain location etc.).

Using VWSNs in such applications promises the improvement in its efficiency in handling the *on-the-spot* tasks. This improvement in its efficiency would not be feasible without complementing the visual data with the location data. This shows that, in the practical applications for both normal and video sensor networks, the sensors' location and the network topology are critical issues. The following sections, present the motivation for this work in developing the vision based localization technique and list the thesis contributions.

1.5 Motivation

Various tasks that the sensor networks handle in monitoring and tracking pose serious research problems, like accurately locating the sensor nodes, minimizing the power consumption and efficiently routing the data through the network. As mentioned above, localization and calibration are few of the many important problems in designing and using any sensor network. Their importance can be clearly summarized in these assumptions [8]:

- All the nodes of a WSN are location aware, specially, in the applications that are based on data or event detection and reporting.

- Most of these applications use routing protocols that rely on the knowledge of the geographical locations of the sensor nodes to complete a given task successfully.

- In a VWSN, location awareness together with camera based parameters like its angle of orientation (and, hence its resulting field of view (FOV)) are necessary in reconstructing the visual scenes of the sensing areas for surveillance and tracking purposes.

These fundamental requirements of almost all WSN applications and VWSN motivate this research to come out with a suitable localization method that is powerful and works with more flexibility. The vision based localization method derives its importance from the need to utilize the same equipment that the sensor nodes are equipped with for surveillance and tracking applications without having to seek help from other external devices that may otherwise increase cost of the sensor nodes and affect the underlying infrastructure of the network.

In addition, the localization technique is further enhanced to estimate the angle of orientation and therefore the resulting FOV – a process called calibration in this thesis.

1.6 Objectives

The objective of this research is to develop a novel vision based localization and calibration technique for video sensor networks. This thesis intends to make use of the visual data provided from the nodes and design a novel method to localize the entire network with good accuracy and construct a combined view of the area being sensed. It is based on this fundamental knowledge that the network can be made use of in various applications. In addition, the localization technique should also be able to localize the network with fewer number of reference nodes as compared to other techniques.

1.7 Work Contributions

In this thesis, a novel vision based localization and calibration technique has been proposed that attempts to localize one-plane camera sensor network with random deployment of its nodes. This research on such a technique has made the following contributions to the field of localization in video sensor networks. These contributions are summarized in develop a novel self-complete procedure that helps localize all the nodes and helps construct the scene under observation. The localization technique has following novel features:

- The novel vision based localization technique for static VWSN uses the vision characteristic in estimating the coordinates of the nodes *without requiring* extra hardware. This research also succeeded in decreasing the necessary number of *global reference nodes* to just one as compared to other traditional techniques which require at least two. Unlike others, this technique allowed the three reference nodes to be randomly deployed.

- Furthermore, a calibration procedure has been developed that systematically makes use of the camera node calibration parameters like angle of orientations of the nodes in defining the overlapping areas and reconstructing the visual senses for tracking purposes.

- In addition, a new object has been localized and calibrated and track while its movement in this network that. The traditional way of subtracting the background in the images used to detect the new object and locate it using conventional triangulation method.

1.8 Thesis Outline

The rest of this thesis is organized as follows: Chapter 2 introduces an extensive background study in the localization and calibration techniques that have been proposed in the WSN field and some of the current state-of-the-art vision based localization techniques. It, also, introduces a background study on the vision characteristic that has been used in the vision technique. Chapter 3 formalizes the main steps of the proposed vision based localization and calibration technique and the way it estimates the location of the camera nodes. Moreover, in the same chapter an application scenario of a stationary surveillance network that has been localized and calibrated with the same proposed technique has been presented. The surveillance network attempts to detect and track any moving object in the sensing area using triangulation method. Chapter 4 presents the practical experiments that have been carried out in the laboratory environment and some of the assumptions made in deploying the nodes in the network. Also presented in this chapter the experiments carried out for the application scenario. In chapter 5, the analysis of the results obtained from the previous chapter is elaborated and present statistical study with a comparison against other vision based and wireless signal based localization technique to evaluate the accuracy of the proposed technique. In chapter 6, a conclusion for this research is presented with some suggestions for future work. Finally, 3 appendices are included in this thesis. Appendix-A explains the process of the localization and calibration as used in the localization technique with some flow charts explaining the node participation. Appendix-B explains the mathematical concept behind calculating the coordinates of one sensor node and derives it in the relative coordinate system. This thesis is ended with Appendix-C that lists the MATLAB® codes representing a graphical user interface (GUI) for difference phases of the camera nodes.

CHAPTER TWO: BACKGROUND STUDY

This chapter presents an extensive literature review on various localization and calibration techniques used in WSNs. Examples of both normal and vision based techniques have been included. Section 2.1 introduces the same Fire Rescue Application using VWSN, but from a *point-of-view* that demonstrates the importance of the sensor nodes' coordinates to complete the desired task. Section 2.2 and section 2.3 a background study in localization and calibration is provided. Here, localization implies finding the coordinates of the nodes and calibration means determining the node parameters *e.g,* orientation angle, the camera FOV and other parameters. Section 2.4 encloses current *state-of-the-art* vision based localization and calibration techniques. In section 2.5, the background study of the vision characteristics that are helpful in explaining the proposed technique is provided. And lastly section 2.6 is the conclusion.

2.1 Localization in Practical Scenarios

In this section, a practical scenario of fire rescue that was mentioned in the previous chapter is presented to demonstrate the importance of localization concept and its process. Assume that a forests area catches up fire and an emergency call is sent to the fire department to blaze out the fire in that area. The department uses a new sophisticated safety system that consists of camera nodes forming a VWSN and a few other useful and recommended sensors, *e.g.,* temperature sensors and wind sensors. The department's helicopter flies over the forests area and, randomly, deploys hundreds of camera nodes at various places. After deployment, the nodes start localizing themselves to construct a *Firefighter Camera Network* as shown in Figure 2-1 and share together some useful information about the area on fire.

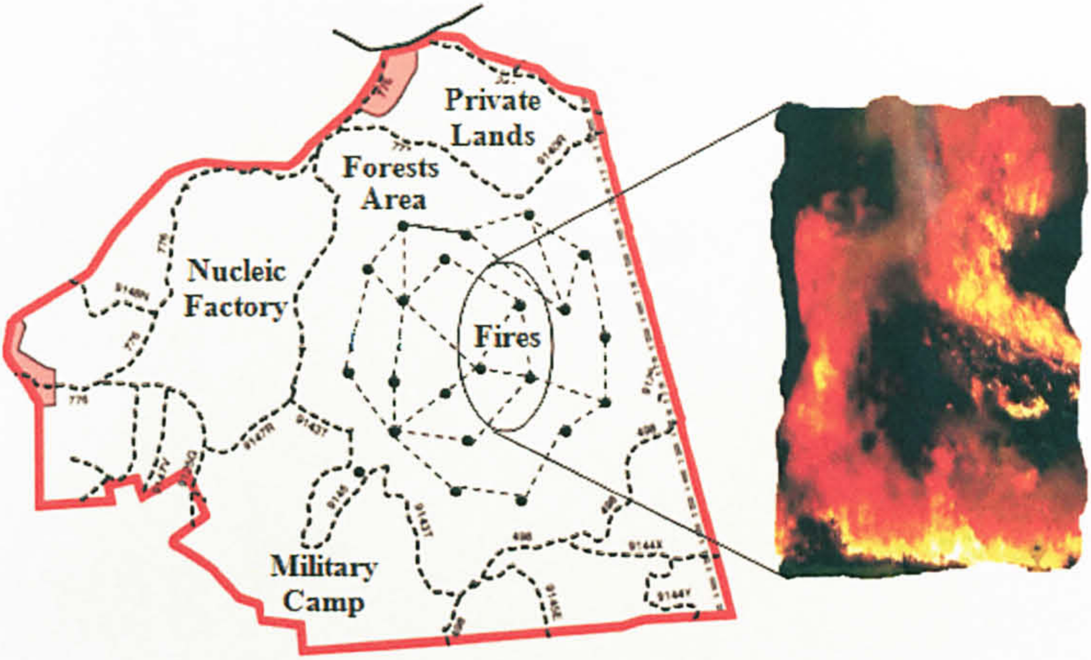


Figure 2-1: Firefighter Camera Network Topology

This type of application requires much care in designing the camera nodes to face the environmental challenges like fire in this case. There are many studies on the requirements of a typical firefighting network - one such study is reported in [7].

Another interesting scenario is that of a video surveillance network in military camp as shown in Figure 2-2. The random deployment of the camera nodes in VWSN helps to discover the area and applying geographical analysis in as short a time as possible before deciding the eventual movement of the army. In addition, VWSN can be used as a surveillance network for monitoring the areas around the camp for more security. This visual information would not be useful with the absence of the location information. The information gathered from the camera sensor network helps in taking appropriate and immediate decision regarding the emergency cases.

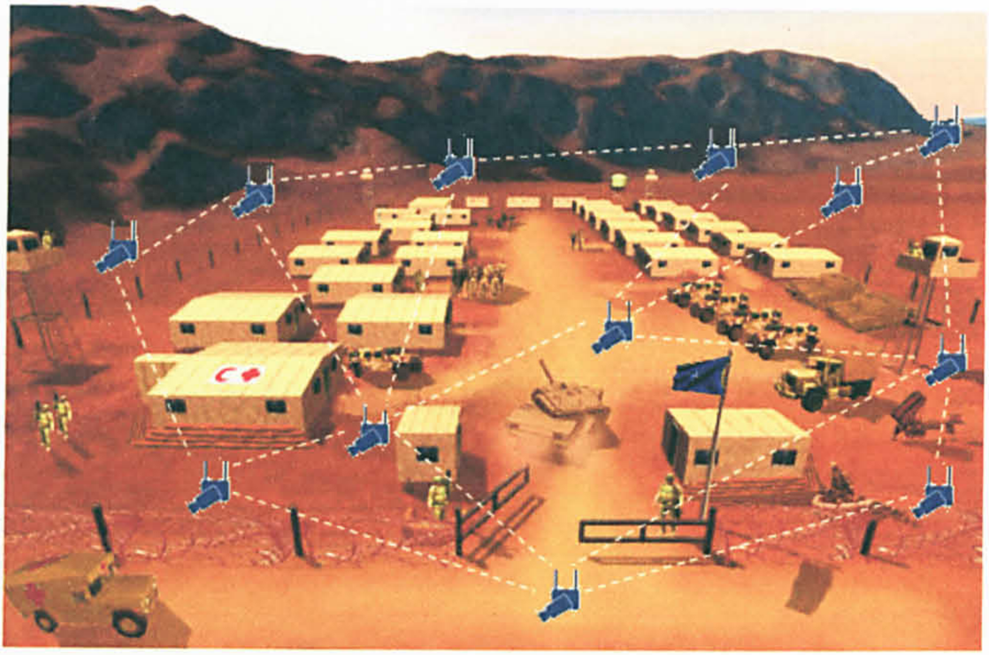


Figure 2-2: Video Surrveillance Network Topology

Recently, many localization techniques have been proposed in the context of random deployment of a sensor network, as is detailed in the next section. Typically, the localization process is meant to estimate the camera nodes' coordinates relative to some a-priori chosen *reference nodes* - those provided with their global coordinates with the help of global positioning system (GPS).

In both scenarios this information helps the control monitoring department to construct a global network topology and explore the nearest areas. Based on the information collected by the sensors the monitoring department will be able to make significant decisions in controlling the situation.

2.2 Localization Background

Localization is a fundamental problem in WSNs. It is considered as a difficult issue to be resolved in real practical applications. Localization is an algorithm that helps in determining and discovering the geographical location of the sensor nodes in a network by estimating their coordinates. Localization is important when we consider the need to locate the nodes in areas otherwise hazardous, inaccessible or disaster-affected. It is also important when some of the nodes fail and the network is required to reconfigure itself. There are many useful survey-studies on the localization algorithms in WSNs [9]. These are classified into either centralized algorithms or distributed algorithms. The distributed algorithms are further classified depending on which ranging method they are based on - whether they are range-free or range based. One good survey study is contained in a handbook [10] written by *Bachrach, and Taylor*, while another handbook in [11] written by *Mohammad Ilyas, and Imad Mahgoub* they capture the current state of sensor networks and deals particularly with technical challenges such as software protocols, data processing, security, and limited power sources for remote sensors. These handbooks cover topics relating to various applications of sensor network such as tracking, and location management which are important topics in this work. The centralized and distributed algorithms are presented in the following sections.

2.2.1 Centralized Algorithms

The main idea behind these algorithms is to collect some ranging and neighborhood information, known as topology information, and send them to the Central Processing Unit (CPU) that has the task of calculating and building up the entire coordinate system of the network with minimum positioning error. These algorithms do not care about the complexity and the computational cost. Instead, they take care to minimize the positioning error for the coordinate system. This error can be minimized progressively by requiring more and more extensive information from the regular sensor nodes in the sensing field participating in the localization task. However, this may, sometime, be impractical because the nodes consume energy and have communication constraints unlike the central unit which has unlimited power and resources.

Doherty, Pister, and Ghaoui [12] have proposed feasible solutions for the localization problem using convex optimization. In their algorithm, they represent the connectivity constraints among the nodes as a linear or semi-definite program which is solved to produce the region of uncertainty in the node location.

Shang Yi [13] has proposed MDS-MAP method for solving the localization problem. This algorithm uses a given connectivity information for the network to roughly estimate the distance between each pair of nodes and then uses a maximization of multidimensional (MDS) a-priori probability technique to drive the location of the nodes.

2.2.2 Distributed Algorithms

In this class of algorithms, the localization problem is distributed among all the sensor nodes to decrease the computational cost of the algorithm with some help from the reference nodes. Each node collects some geographical information and uses them to calculate its relative coordinate with respect to a-priori known coordinates of reference nodes. The next sub-sections classify the distributed algorithm into two categories *Range-based, and Range-free*, based on the ranging technique used in estimating the distance between the nodes.

2.2.2.1 Range-Based

This category of distributed algorithms relies on the fact that each sensor node estimates the distance to the reference nodes by means of some specialized hardware, *e.g.*, array of antennas, infrared sensors, and/or acoustic transceiver. These estimated distances are then used in calculating the coordinates by tri-angulations, tri-lateration, or multi-lateration methods. There are many popular ranging techniques in sensor networks, such as, Time of Arrival (ToA), Time Difference of Arrival (TDoA), Angle of Arrival (AoA), and Radio Signal Strength Indication (RSSI).

- ToA: This ranging technique is based on a sensor node noting the travel time of signals (radio, infrared, ultrasonic, etc) from some other chosen nodes and by knowing the relation between the signal speed and the travel time, the distances between the nodes can be measured. *Xing-Yu* [14] has proposed an assisting localization models for WSNs that is based on different ranging techniques, one such model based on ToA.

- TdoA: This technique measures the time difference of arrival between different types of signals from the reference node or between the signals from some chosen nodes to estimate the distances between the nodes. *Nissanka* [15] has used the first scenario on his localization method where each node is assumed equipped with radio and ultrasonic sources. The concept behind his method is to send ultrasonic signal after a constant delay time from sending radio signal as shown in Figure 2-3.

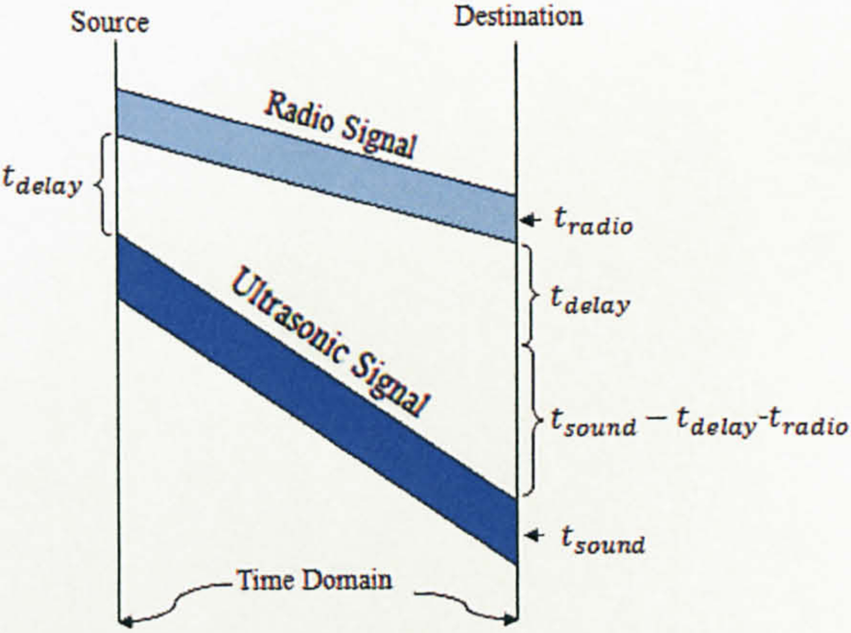


Figure 2- 3: TDoA technique. The source send radio and sound signals and the destination will note the times to calculate the distance [9]

Relying on the fact that there is difference between the radio wave speed and the sound speed and the delay time, is calculated using equation (2.1)

$$d = (v_{radio} - v_{sound}) \times (t_{sound} - t_{delay} - t_{radio}) \qquad 2.1$$

Xing-Yu [14], however, uses the second scenario in his assisted localization method to estimate the distances between nodes. In this method, each sensor node will note the first received signal from the nearest reference node to be considered as the reference point for differential time delays. The time difference between the other arriving signals and the reference point will represent the difference on distance from that node as shown in Figure 2-4.

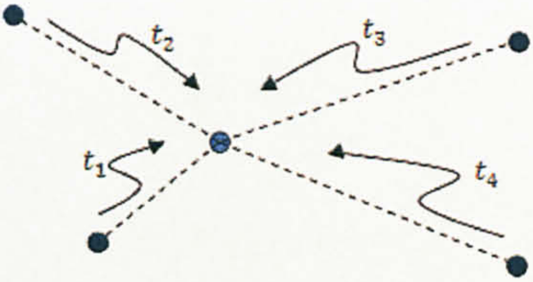


Figure 2- 4: TDoA Technique. The Node will Note the First Received Signal to be the Reference Signal for the Rest

The TDoA for the second signal resource will satisfy the following equation:

$$t_{21} = t_2 - t_1 = (d_2 - d_1)/v \qquad 2.2$$

Here, t_1, t_2 are the time taken by the signal from the 1st and the 2nd reference nodes respectively to reach the sensor node, and, d_1, d_2 are the Euclidean distances, t_{21} is the TDOA, and v is the propagation speed.

- AoA: This technique is a method for determining the direction of propagation of an incident wave and its angular offset from some reference direction. One common approach in this method is to use an antenna array on each sensor node and measure the TDoA and the phase difference of arrival at each element of the array. *Nasipuri,A.et al* [16] have proposed that the sensor node would note the times when it receives the different beacon signals, and then it can use it to evaluate its angular bearings and location with respect to the beacon nodes by simple triangulation.

- RSSI: This ranging technique is built on the knowledge of the transmitter power, the path loss model, and the power of the received signal to estimate the distance between nodes. In this technique, the path loss model is affected by the non-stationary propagation environments. *Alippi, C.* [17] has used RSSI technique to map the network by conveying short packets at different power levels through out the network and storing the average RSSI values of the received packets in memory tables to create nonlinear ranging model to solve the localization problem.

Despite the fact that these ranging techniques give fine localization results, range-based schemes require the nodes to be equipped with special hardware to produce with fine distance estimation.

2.2.2.2 Range-Free

Unlike the range-based techniques, range-free techniques do not rely on special hardware to estimate the distance between the nodes and are only based on the geographical information from the node's neighborhood to estimate these distances. So, they are considered more cost-effective. The idea behind these techniques is that, when certain node receives signal from a reference node, it would mean it can place the reference node somewhere on a circular area with radius equal to the transmitting range of that reference node. When the same node hears again from another reference node in the neighborhood, it will reduce the uncertainty region to the dark or light gray areas of intersection between the two circles as shown in Figure 2-5.

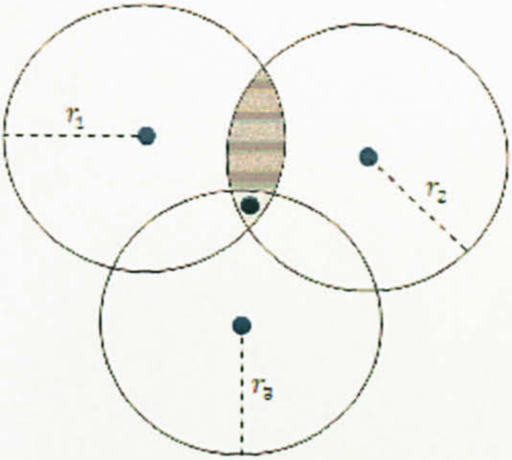


Figure 2-5: The Idea of Range-Free Techniques

When another reference node is added to the previous scenario, the uncertainty area will reduce more to the light gray area only as shown in the figure. There are many localization algorithms that are based on the range-free techniques to estimate the distances. Some of these algorithms are discussed below:

- Centroid [18] is the simplest range-free localization method that assumes that each node will listen and collect all beacon signals from the reference nodes in the neighborhood. By considering the number of receiving and sending packets, it will calculate the connectivity with that reference node and evaluate a chosen threshold to select the set of the reference nodes in the nearby region. This method will place the node at the intersection area of the selected reference nodes which is defined by the *centroid* of these reference nodes. The concept of this method is strongly based on the deployment of the reference nodes in the region which, sometime, can render it impractical, especially in networks with low reference nodes density.

- Distance Vector Hop propagation method (DV-Hop) [19] is a ranging localization technique that employs a classic vector distance exchange between the nodes. At the outset, all the reference nodes send a *beacon* message throughout the network including their locations and a hop-count parameter initialized to one, so each node after this process can maintain the shortest number of hops to all reference nodes. The second step in this technique is to convert these counting hop distances to a real distance basing on the average distance estimation calculated by the reference nodes using this formula:

$$average\ distance = \frac{\sum_{j=1}^{n} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j=1}^{n} h_{ij}} \quad , i \neq j \qquad 2.3$$

where, n is the number of reference nodes in the network, h_{ij} is the number of hops between the reference nodes i and j.

- Amorphous Positioning [20] is a localization algorithm derived directly from the DV-Hop technique. It starts with the same concept by spreading a *beacon* message throughout the network in order to know the locations of all the reference nodes and the shortest number of hops to reach them.

The Amorphous localization algorithm uses a different method from DV-HOP algorithm to estimate the average distance of a single hop by assuming that the density of sensors in the network, n_{local}, is known in advance, so that it calculates the average distance of the hop-counting parameter using the help of *Kleinrock and Slivester* formula [21]:

$$average\ distance = r\left(1 + e^{-n_{local}} - \int_{-1}^{1} e^{-\frac{n_{local}}{\pi}(\arccos\ t - t\sqrt{1-t^2})} dt\right) \quad 2.4$$

- Approximate Point In Triangle (APIT) [22] is another range free localization scheme where the idea is to divide the environment into triangular regions between beaconing nodes as shown in Figure 2-6.

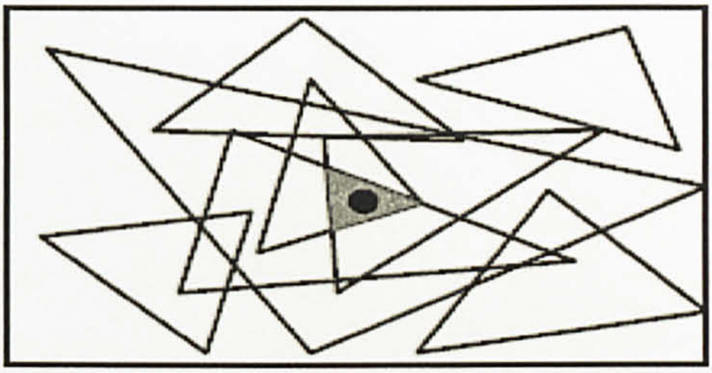


Figure 2-6: Area-based APIT algorithm overview [22]

The node then performs a perfect theoretical test called Point-In-Triangulation (PIT) test by analyzing the signal strength from different audible anchors to decide whether it is inside or outside a given triangle. The scheme chooses the center of gravity (COG) of the intersection of all the triangles in which the node resides as the location estimation.

2.3 Calibration Background

Calibration is the process that identifies the sensor nodes' parameters, *e.g.,* angles of orientation, pan, tilt, zoom, spatial relationships between nearby cameras, and the coverage and overlapping areas in video sensor networks, and modify these parameters by means of adjustment aided by a control unit to increase and optimize the overall effectiveness of the network. The importance of calibration comes from the huge number of sensor nodes that would render the manual calibration and management process infeasible, similar to the localization process for a huge number of nodes in the network. Also, the calibration provides a more precise description to the events sensed. There are many important parameters in the calibration process that will be described in detail;

- Orientation Angle: It is the angle that the camera makes with a reference line as shown in Figure 2-7. The camera is said to look in the direction where an imaginary line bisecting the field of view of the camera looks.

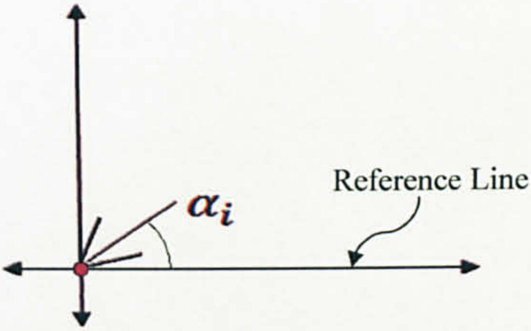


Figure 2-7: Orientation Angle Description.

- Coverage and Overlapped Areas: The coverage area is defined as the area that is seen in the camera's field of view. Conversely, the overlapped area is the area covered by two or more sensor nodes at the same time. As shown in Figure 2-8, the coverage areas of sensor S_1 and sensor S_2 are *Area1* and *Area 2* respectively, whereas the overlapped area is *Area 3.*

These areas basically rely on the camera field of view angle and its sides' length determined by resolvable distance of the camera, the orientation angle, and the coordinate of the sensor nodes, also described in more detail in chapter 3.

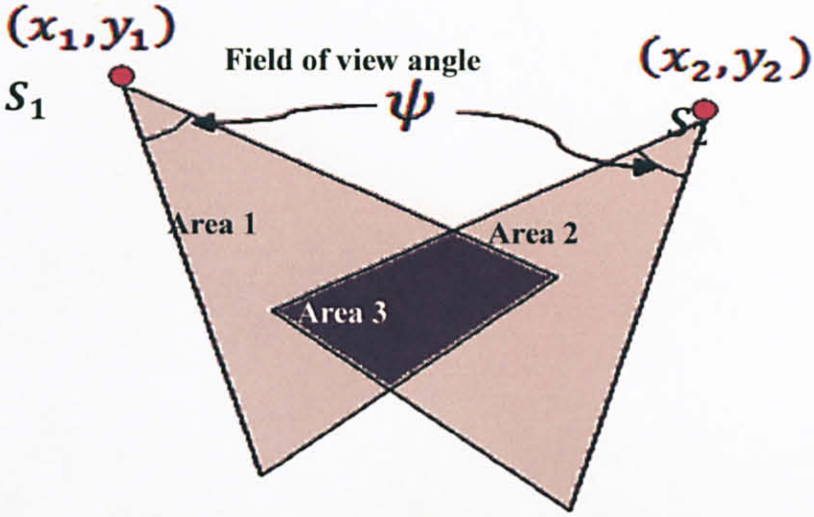


Figure 2-8: Coverage and Overlapped Areas Description

One of the most important calibration technique for camera sensor networks is proposed by *Xiaotao Liu* [23]. They propose an automated calibration protocol that determines the location and orientation of a camera sensor node with the help of only four reference nodes whose real locations are a-priori known. The technique is based on optics, geometry, and computer vision principles.

2.4 Vision Based Localization Method

The studies on the vision based localization techniques of a camera sensor networks are new and limited. One such study is carried out by *Huang Lee, and Hamid Aghajan* [24]. They introduce a technique that localizes the nodes of a surveillance network based on observations of a non-cooperative moving target. This technique assumes that both the reference nodes and the sensor nodes need to be localized. It is further assumed that the nodes to be localized can observe the moving target simultaneously. The nodes participating in joint synchronized observations of the target define a relative coordinate system to the reference nodes. This system is then modeled as a system of equations which can be solved by applying the Gauss-Newton method. They also proposed in [25] a vision-based localization technique that finds the coordinates of the camera nodes with the assistance of a moving robot controlled by the sensor network. The algorithm detects the robot's location in the camera image plane and expresses the relationship between the virtual observed coordinates and the global coordinates in a system of equations that can be solved to give the camera node's coordinates when three observations are available. This technique studies the localization problem where the image planes of the nodes are parallel to the robot's motion plane with the assumption of a constant speed.

The vision based localization methods are different from the traditional methods in the way they estimate the distance between the nodes in the network. This is done by creating some relationships between the actual locations of the nodes and the scenes that comes from the video streams of the camera nodes. This is one of the most useful benefit of using vision localization methods that, instead of using additional hardware (arrays of antenna or special positioning hardware GPSs), it can make use of the collected data to help build up the topology of the network. The vision-based technique should be designed carefully so as to employ lightweight image processing with minimum data exchange between the nodes because of the complexity and cost limitations. Like the normal localization methods, we can classify the vision based techniques, as suggested by Huang Lee [24], into two schemes - *Decentralized* schemes and *Cluster-based* schemes. In the decentralized, the sensor node estimates its own coordinates and calibration parameters from its observations and also the broadcasted information by the reference nodes. The cluster-based schemes, on the other hand, transfer the job of solving the coordinates and the calibration parameters of all the sensor nodes to the reference node (cluster-head) after receiving their joint observations. Figure 2-9-(a) and (b) show the information data flow in the network for both the schemes [24].

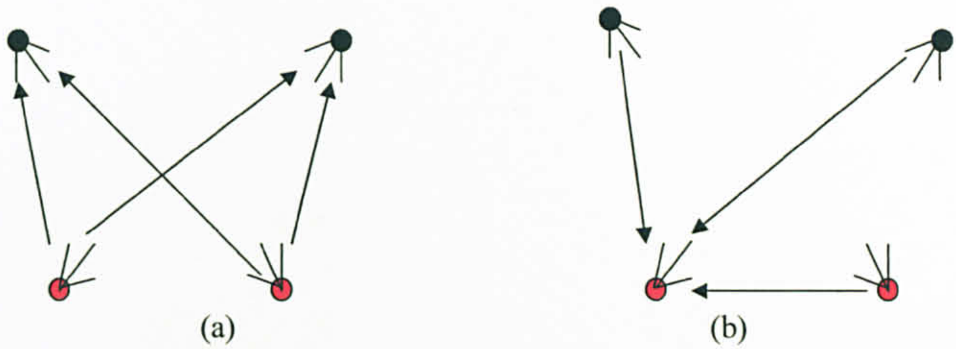


Figure 2-9: Information Data Flow (a) Decentralized (b) Cluster-based [24]

2.5 Vision Characteristics Background

This section gives an overview of how the images are taken and the importance of resolution on the estimation accuracy. This is important since the optics used in the camera and its resolution would determine how good, is the vision based technique in estimating the coordinates of the nodes. Accordingly, we present the pinhole camera model and the issue of camera resolution in the next two subsections. The following section provides analytical description of the pinhole camera model and how it can be useful in this work. In chapter 4, the accuracy of this technique in the estimation process, particularly, the effect of the object's height variation on the estimation is explained.

2.5.1 The Pinhole Camera Model

In this subsection, the pinhole camera model of Figure 2-10 which shows that there is a relationship between the real distance (d) of the object from the camera's lens and the image's height (h') on the image plane with respect of the focal distance (f), and the real object's height (h) is analyzed.

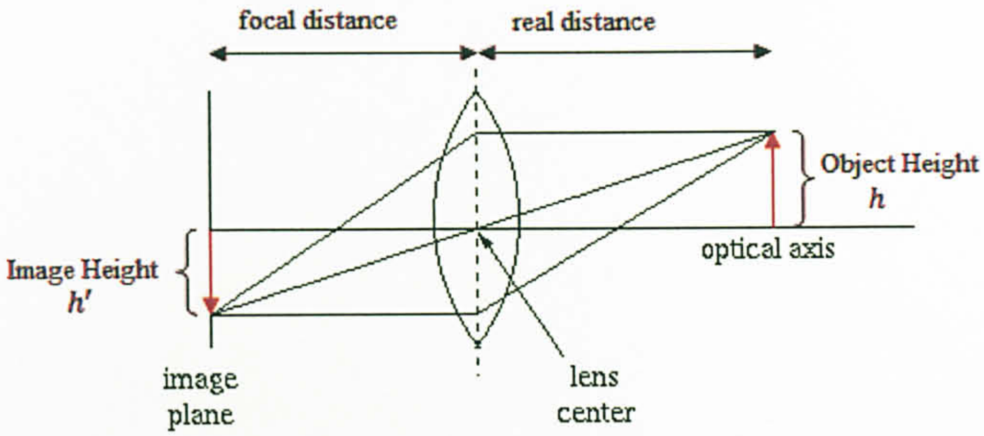


Figure 2-10: The Pinhole Camera Model

We can simplify this camera model into two triangles AOB, and DOE as shown in Figure 2-11 to explain the relationship more clearly. Using the Euclidean construction of parallel lines in the two triangles, equation (2.5) is obtained,

$$\frac{\overline{AB}}{\overline{AO}} = \frac{\overline{DE}}{\overline{DO}} \qquad 2.5$$

which translates into:

$$\frac{h\prime}{f} = \frac{h}{d} \qquad 2.6$$

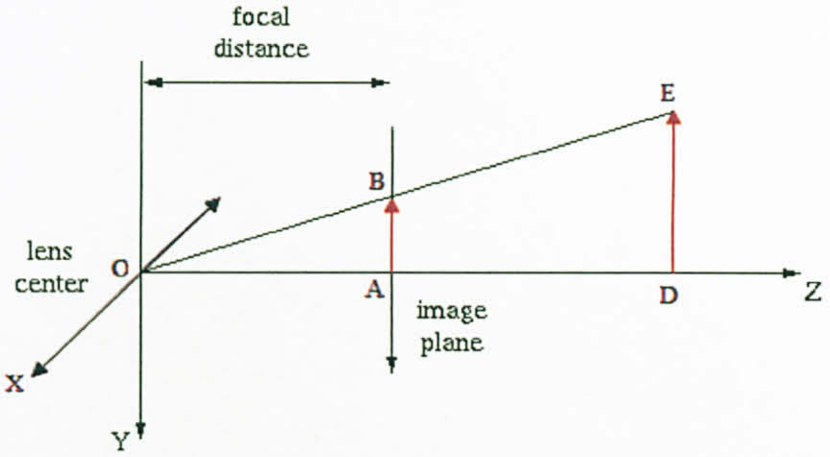


Figure 2-11: The Relationship Triangles

From the above geometry the distances between the lens of the camera and any object with known height can be estimated. And if this principle is used, it will produce a new vision-based localization technique and this is the main reason behind this work. Like the other estimation techniques there are some parameters that may affect the process of the estimation (such as the camera resolution and the object's height variation). In the next subsection, a background study only in the camera resolution is provided. While in chapter 4 both the parameters to prove that it is a useful technique to be used for the localization purposes of a camera sensor network will be tested experimentally.

2.5.2 The Camera Resolution

The resolution of the camera is defined as the number of tiny light-sensitive squares (called pixels) on the sensor of the digital camera, and its ability to distinguish separate visual information such as details and fine patterns [26]. These pixels are only sensitive on a portion on its area called photodiodes as shown in Figure 2-12, which, in turn, are sensitive to a certain range of brightness that determines their dynamic range.

It is very important to note that not all the pixels on the camera sensor are participating in the process of capturing the picture because many of them are not illuminated by the image circle produced by the lens. This leads to the definitions that are derived from pixels:

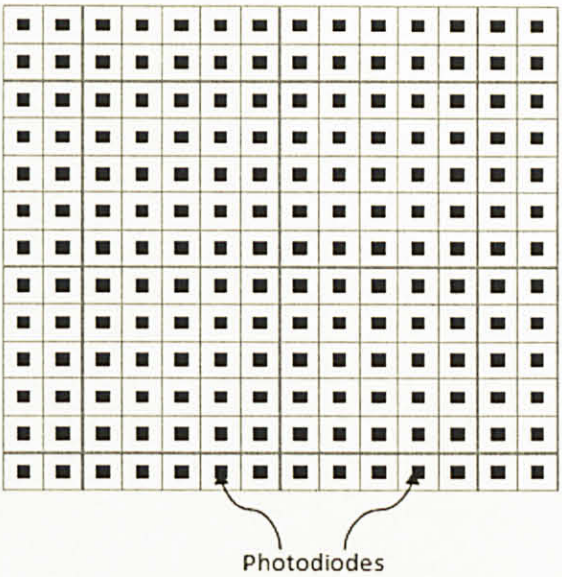


Figure 2-12: The Sensitive Area of the Pixels [26]

- Total Pixels: Defined as the total number of pixels on the camera's sensor. Not all of these pixels are used to capture the image as shown in Figure 2-13-a, where the image circle is not covering all the area of the total pixels.

- Picture Pixels: Defined as the number of pixels placed in the picture area which is used to capture the image as shown in Figure 2-13-b.

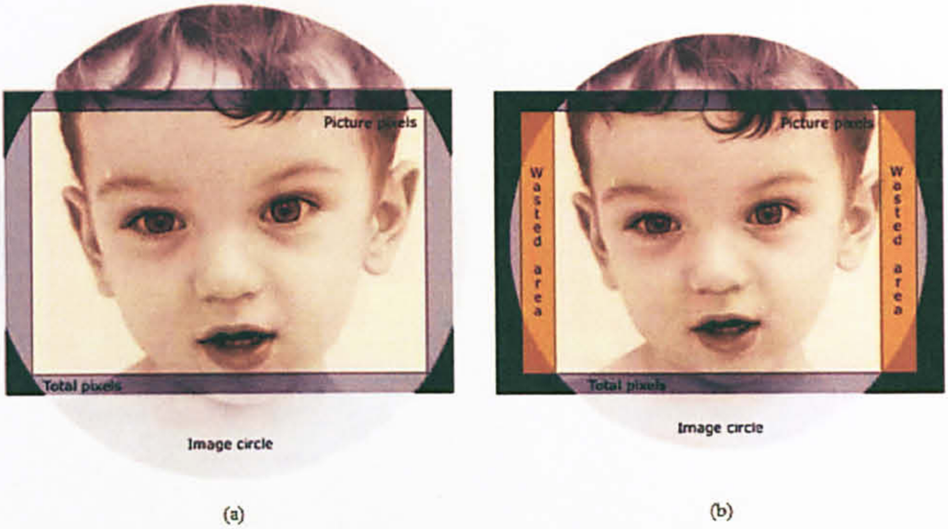


Figure 2-13: Types of Pixels count (a) Not All Pixels Used to Capture the Image (b) The Picture Pixels More Small than the Picture Area [26]

- Effective Pixels: Assumed to be the same as the picture pixels by including pixels used in calibrating the black.

- Recorded Pixels: These pixels are not physically located on the sensor and they are defined as the number of pixels interpolated and stored to create a viewable image.

- Output Pixels: Defined as the standard number of pixels cropped from the recorded pixels to fit the standard dimension such as VGA, SVGA, and XGA or to fit aspect ratio such as 3:2, 4:3, and 16:9.

The picture pixels, effective pixels, and output pixels give more indication than the total number of pixels on the capabilities and limitations of the camera in use.

The chart in [27] as shown in Table 2-1 is used to prove that the standard video resolution 640x480 and above is good, since most details are discernable and it enables the camera nodes to do some image/vision processing on the data. This is specially useful today given the progress embedded systems have made lately, where even low cost cameras come with such nominal resolution.

Table 2-1: Digital Camera Resolution Chart [27]

Capture Resolution	Video Display	Print Size						
		2x3"	4x5"/4x6"	5x7"	8x10"	11x14"	16x20"	20x30"
320x240	Acceptable	Good	Acceptable	Poor	Poor	Poor	Poor	Poor
640x480 - 0.3 Megapixel	Good	Excellent	Good	Poor	Poor	Poor	Poor	Poor
800x600	Excellent	Photo Quality	Very Good	Acceptable	Poor	Poor	Poor	Poor
1024x768	Excellent	Photo Quality	Excellent	Good	Acceptable	Poor	Poor	Poor
1280x960 - 1 Megapixel	Excellent	Photo Quality	Photo Quality	Very Good	Good	Poor	Poor	Poor
1536x1180	Excellent	Photo Quality	Photo Quality	Excellent	Very Good	Acceptable	Poor	Poor
1600x1200 - 2 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Very Good	Acceptable	Acceptable	Poor
2048x1536 - 3 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Excellent	Good	Acceptable	Acceptable
2240x1680 - 4 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Very Good	Good	Acceptable
2560x1920 - 5 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Excellent	Very Good	Very Good
3032x2008 - 6 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Excellent	Very Good
3072x2304 - 7 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Excellent	Excellent
3264x2448 - 8 Megapixel	Excellent	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Excellent
10 Megapixel +	Excellent	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality	Photo Quality

Moreover, in chapter 4 on *Experimental Testing,* a practical, study is provided not only to demonstrate the effect of the camera resolution on the error of the distance estimation process but also as a calibration of the camera used in our experiments later.

2.6 Conclusion

Recently, many algorithms have been proposed to solve the localization problem in WSN. Some of these algorithms are based on GPS connected to the whole network, but it is not only an expensive suggestion but also not sufficient to provide all the information needed. In this chapter, background studies of the literature have been provided on the localization and calibration concepts in WSN and VWSN with an explanation of the localization problem in real typical scenario. Moreover, the difference between the vision based and wireless signal based localization techniques are highlighted. Also, some of the proposed algorithms of both types are presented with explanation of the main concepts. In addition, background study of the vision characteristic is provided with some of the parameters that may affect the vision based estimation.

CHAPTER THREE: PROPOSED VISION BASED TECHNIQUE

This chapter presents the proposed localization technique that is based on camera vision. In section 3.1 of this chapter some assumptions relating to the system model are explained. In section 3.2 the system model and an analytical description of the proposed technique that is used to localize the sensor nodes and build-up one-plane topology of the network are discussed. Section 3.3 discusses an application scenario of tracking an object in a stationary camera network whose nodes have been localized by the above technique. And section 3.4 concludes the chapter.

3.1 Video Sensor Network Model and Assumption

In this section a VWSN is considered in which all sensor nodes are equipped with movable camera hardware that is essential for monitoring and other purposes. In order to locate the sensor nodes in terms of global coordinates, one of the nodes (called *master node*) is provided with special hardware, *e.g.,* global positioning system (GPS) receiver that will help obtain its worldwide location. The height, *h*, of the real sensors is presumably known. This assumption can be justified for the real practical applications, since all the sensor nodes in any given network generally are procured from the same manufacturer. Unlike the traditional localization algorithm, the vision data from the network is used to help localize the sensor nodes. Hence some vision based features, like shape and chromatic signs, become critical. Each sensor node has a chromatic signs (*Red, Green*) that gets turned on/off to distinguish between the phases it is passing through. Red sign indicates the completion of the calibration phase and thereby help classify the node as a well calibrated node. Similarly, the green sign indicates that the node has not been calibrated yet. Having both signs turned-on in certain node helps the other nodes in the same tier to detect that node in their FOV. At first, the master node starts searching for other two nodes having both *red and green* signs turned-on so they can be differentiated from the other nodes in the network. Once the master node has estimated their coordinates, together they can act as three reference nodes with respect to which the coordinates of all other nodes can be obtained. How the master node estimates the coordinates of the two reference nodes is explained in section 3.2. These three nodes are now called reference nodes in the first tier of the network will respectively broadcast their coordinates while displaying both *red and green* signs to be distinguished from the rest. All these steps will be clarified in details in the proposed technique section. This research assumes that the camera nodes (*normal and reference*) are randomly distributed over the sensing area.

Nodes that can clearly see the reference nodes in their FOV are considered being part of the *first tier* of the network. The *second - tier* consist of nodes that can clearly see the first-tier on their FOV. After the coordinates are obtained, the camera node S_i is described by the vector $C_i = [(x_i, y_i), \alpha_i]$, *i=1,2,...,N*, where *N* is the number of the sensor nodes in the network, α_i and (x_i, y_i) are the orientation angle and the coordinate of the sensor S_i respectively.

3.2 The Proposed Vision Technique

This section will provide extensive analytical explanation of the procedure that is used to localize and calibrate the camera nodes in one-plane relative coordinate system. The procedure to localize and calibrate the nodes in one cluster has been explained in flow-chart as shown in Figure 3-1.

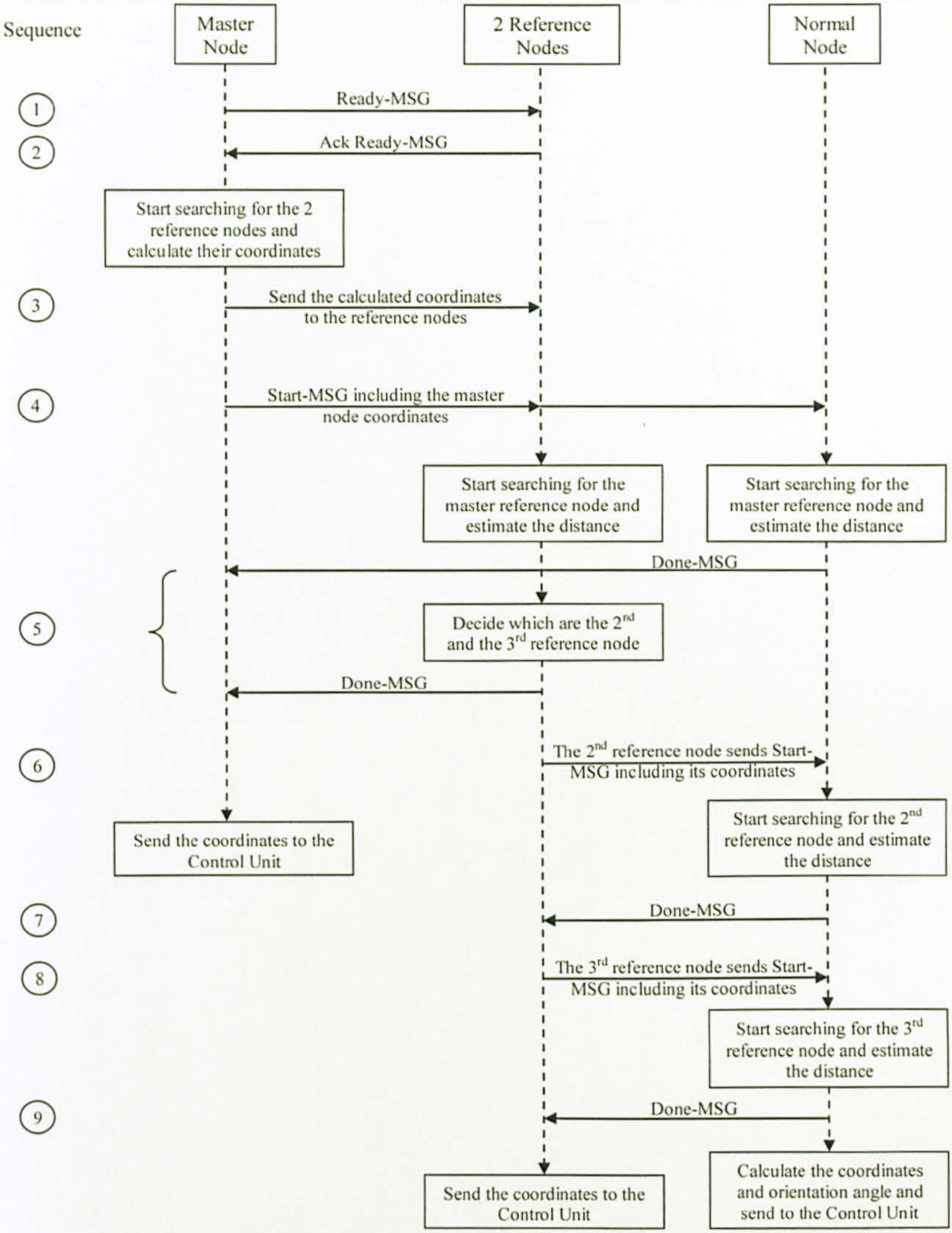

Sequence
Master Node
2 Reference Nodes
Normal Node
1
Ready-MSG
2
Ack Ready-MSG
Start searching for the 2 reference nodes and calculate their coordinates
3
Send the calculated coordinates to the reference nodes
4
Start-MSG including the master node coordinates
Start searching for the master reference node and estimate the distance
Start searching for the master reference node and estimate the distance
Done-MSG
5
Decide which are the 2nd and the 3rd reference node
Done-MSG
6
The 2nd reference node sends Start-MSG including its coordinates
Send the coordinates to the Control Unit
Start searching for the 2nd reference node and estimate the distance
7
Done-MSG
8
The 3rd reference node sends Start-MSG including its coordinates
Start searching for the 3rd reference node and estimate the distance
9
Done-MSG
Send the coordinates to the Control Unit
Calculate the coordinates and orientation angle and send to the Control Unit

Figure 3-1: The Flow-Chart of the Localization and Calibration Procedure in One Cluster

Figure 3-1 shows the entire procedure of localizing and calibrating the nodes in one cluster with their reference nodes. A clear explanation of how the reference and normal camera nodes participate and carry out the search-process in the proposed vision localization and calibration technique are discussed in Appendix A.2 through Appendix A.4. Once the nodes in one cluster localize and calibrate themselves, as will be explained in the next few subsections, they will send their coordinates to the control unit which has the ability to choose three nodes, already localized, somewhere close to the boundary of the present cluster and designate them as reference nodes for the neighboring clusters. This procedure is repeated till the entire network topology is constructed and completed at the control unit. The time constrain of this procedure is explained in chapter 5.

Although this procedure consists of many sequences, it can be distributed into three main steps as follow:

3.2.1 Defining the Core of the Network

In this step, one chosen node (called *master node*) starts to build the topology of the network by considering itself at the origin (0,0) of the coordinate system and its initial orientation is assumed to be in the positive y axis as shown in Figure 3-2.

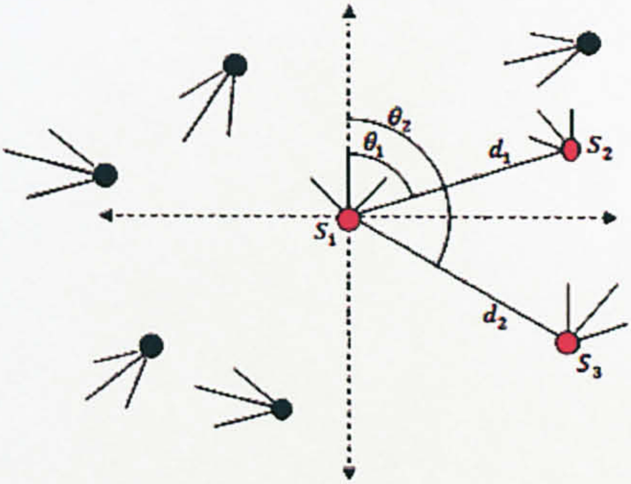


Figure 3-2: The initial coordinate system

The master node, then, start searching for the other two reference nodes in its field of view from 0° (its initial orientation) to 360°. These reference nodes can be recognized by their chromatic signs (their red and green stripes are assumed turned- on). When the master-node detects a reference node it notes the angle θ_i which represents the angle between the new orientation and the initial orientation, as shown in Figure 3-3, and calculates the distance d_i between them using equation (2.6) which is derived from the vision characteristic and the pinhole camera model.

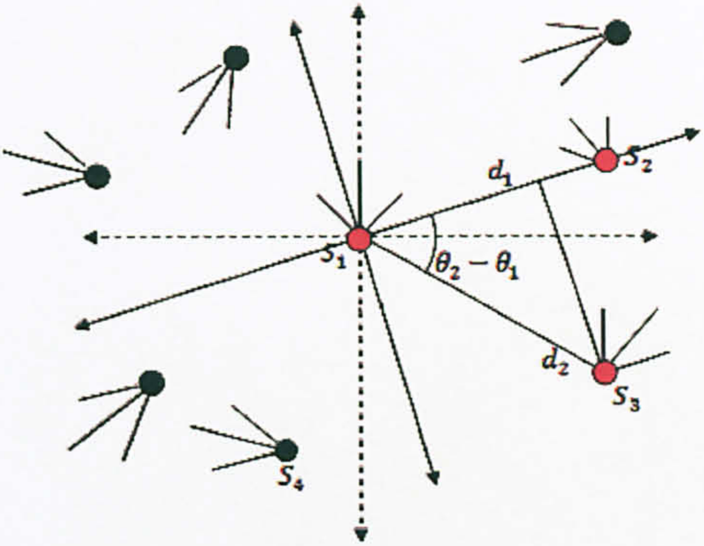


Figure 3-3: The final coordinate system

After one search-cycle ($0^\circ - 360^\circ$) the master node decides to choose the first reference node it detects in its FOV as a *secondary master* node where the line connecting the master and the secondary master nodes is assigned to be the new *x* axis for the final coordinate system. As shown in Figure 3-3 the new coordinates of S_2 will be $(d_1, 0)$ and the orientations of the master node will be θ_1.

From the above figure it is shown that the reference nodes S_1, S_3 and the new *x* axis of the coordinate system will always perform a right angle triangle which will facilitate the computation of the coordinates of the *third reference* node S_3 as:

$$\left(d_2 \cos \Delta\theta_{2,1}, - d_2 \sin \Delta\theta_{2,1}\right) \qquad 3.1$$

3.2.2 Localizing the Rest of the Nodes in the Network

In this step, the three reference nodes broadcast their coordinates *one-by-one* to the entire network. The remaining sensor nodes in the same tier start to search, in their FOV, for theses reference nodes also *one-by-one* and recognize them by their chromatic *red and green* signs that are assumed turned-on. Each time the node detects one of the reference nodes it will note the corresponding coordinates and estimate the distance between them using equation (2.6). It will also note the searching angle $\beta_{i,1}$ - the angle between the initial orientation of each camera node α_i and the orientation when it detects the *master reference* node.

After estimating the three distances $\left(d_{1,i}, d_{2,i}, d_{3,i}\right)$ to the reference nodes, the tri-lateration technique of the assisting localization method is used as in [14] to form a system of equation *Ay=b*, to calculate the sensor node's coordinates. This yields the following:

$$y = (A^T A)^{-1} A^T b \qquad 3.2$$

where,

$$A = \begin{bmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_1) & (y_3 - y_1) \end{bmatrix}, y = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

$$b = {}^1\!/_2 \begin{bmatrix} \|S_2\|^2 - \|S_1\|^2 - {d_{i,2}}^2 + {d_{i,1}}^2 \\ \|S_3\|^2 - \|S_1\|^2 - {d_{i,3}}^2 + {d_{i,1}}^2 \end{bmatrix}$$

A brief example and the derivation of the equations relating to the localization process for one camera node with the help of the reference nodes coordinates and the three estimated distances is presented in Appendix B.

3.2.3 Calibrating the Sensor Node

This step involves calculating the orientation angle of each sensor node relative to the network's coordinate system and also the coverage area in the FOV of each sensor node in the network to complete the localization and calibration vision technique. The common overlapping area is also calculated in percentage between two sensor nodes in the network.

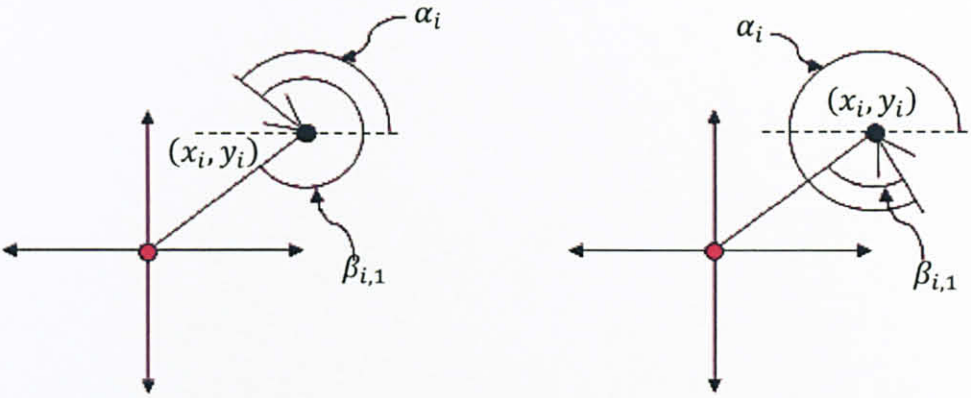
3.2.3.1 Calculating the Orientation Angles

The goal in this part is to obtain the orientation angle α_i of the node relative to the coordinate system of the whole camera network. Doing so completes the localization and calibration for the node. Based on the quadrant of the coordinate system in which the camera node lies in and the angle $\beta_{i,1}$ that the node discovers the origin to be at, the orientation can be determined as follow:

- For the node which lies in the first quadrant of the coordinate system:

$$\alpha_i = \begin{cases} \beta_{i,1} - \omega, & if \ \omega > 0 \\ \beta_{i,1} - \omega + 2\pi, & if \ \omega \leq 0 \end{cases}$$

where $\omega = \beta_{i,1} - \left[\pi - \tan^{-1}\left(\frac{y_i - y_1}{x_i - x_1}\right)\right]$



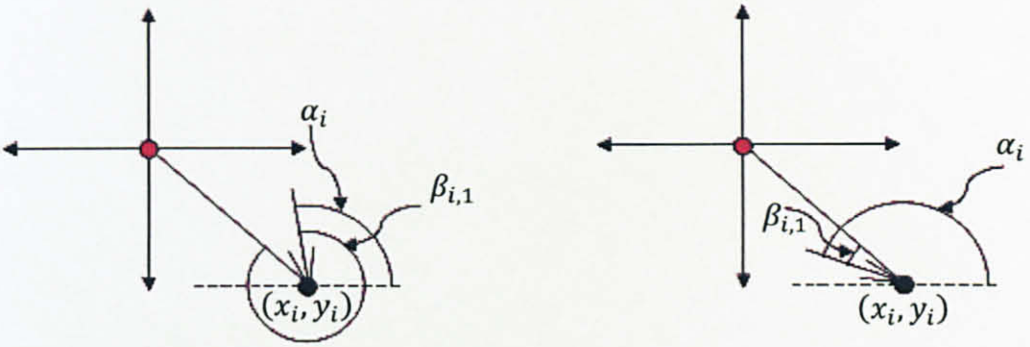
(a) $\omega > 0$ (b) $\omega \leq 0$

Figure 3-4: The Node Orientation Angle in the 1st Quadrant

- For the node which lies in the fourth quadrant of the coordinate system:

$$\alpha_i = \begin{cases} \omega - \pi, & if \ \omega \geq \pi \\ \omega + \pi, & if \ \ \omega < \pi \end{cases}$$

where $\omega = \beta_{i,1} + \tan^{-1}\left(\frac{y_i - y_1}{x_i - x_1}\right)$



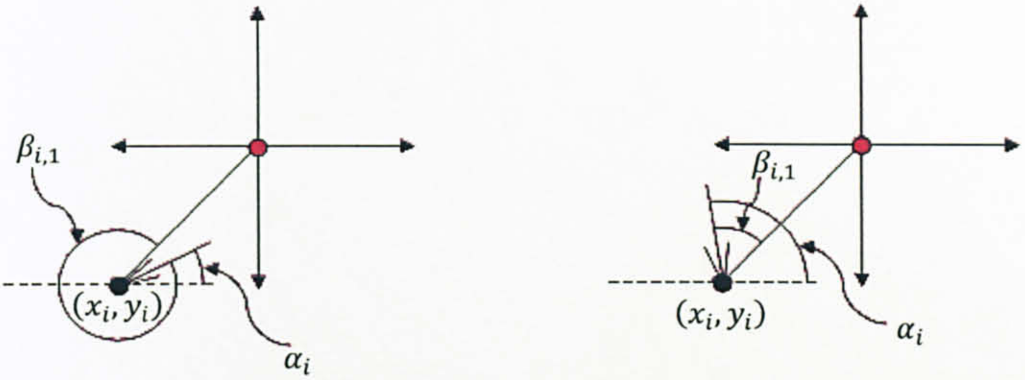
(a) $\omega \geq \pi$ (b) $\omega < \pi$

Figure 3-5: The Node Orientation Angle in the 4th Quadrant

- For the node which lies in the third quadrant of the coordinate system:

$$\alpha_i = \begin{cases} \omega - 2\pi, & if\ \omega \geq 2\pi \\ \omega, & if\ \ \omega < 2\pi \end{cases}$$

where $\omega = \beta_{i,1} + \tan^{-1}\left(\frac{y_i - y_1}{x_i - x_1}\right)$



(a) $\omega \geq 2\pi$ (b) $\omega < 2\pi$

Figure 3-6: The Node Orientation Angle in the 3rd Quadrant

- For the node which lies in the second quadrant of the coordinate system:

$$\alpha_i = \begin{cases} \omega, & if\ \omega > 0 \\ \omega + 2\pi, & if\ \ \omega \leq 0 \end{cases}$$

where $\omega = \beta_{i,1} + \tan^{-1}\left(\frac{y_i - y_1}{x_i - x_1}\right)$

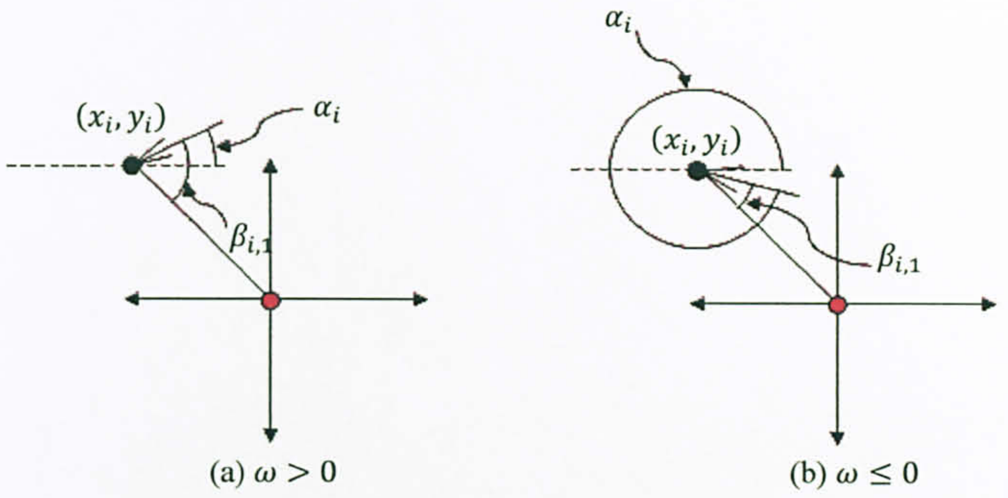


Figure 3-7: The Node Orientation Angle in the 2nd Quadrant

Once the camera sensor node calculates its own coordinates and orientation angle, it sends an updating message that contains the calibration vector $C_i[(x_i, y_i), \alpha_i]$ to the CPU to help build-up the whole network topology and calculate the overlapping areas in the respective sensing field. The methods to calculate the coverage area and overlapping area are discussed in the next section. The CPU can adjust these areas and make some modifications by sending an adjustment messages to the camera nodes in order to get a good coverage.

3.2.3.2 Calculating the Coverage Area and the Percentage of the Overlapped Area between Two Nodes

The main idea behind this part is to calculate the area covered by a given sensor node and investigate the percentage area overlapping any two nodes in the network. As mentioned in chapter 2, these areas are basically determined by *nodes's parameters*, like coordinates and the orientation angle of the sensor node, and *network's parameters*, such as the camera FOV angle and the length of its sides. The former vary from one node to another while the latter remain constant for the entire video sensor network.

Assuming that there are two sensor nodes S_1 and S_2 with known coordinates and orientation angles that can be clalibrated as $C_1[(x_1, y_1), \alpha_1]$ and $C_2[(x_2, y_2), \alpha_2]$ respectively as shown in Figure 3-8. Each sensor node will cover an area of triangle ΔABC, where:

$A = B$, and known as the sides of the FOV angle.

$C' = A\varphi$ where φ is the angle of the FOV.

However, for calculating overlapping areas, the arc C' can be replaced by straight-line C. Thus, $C = 2A\sin(\frac{\varphi}{2})$.

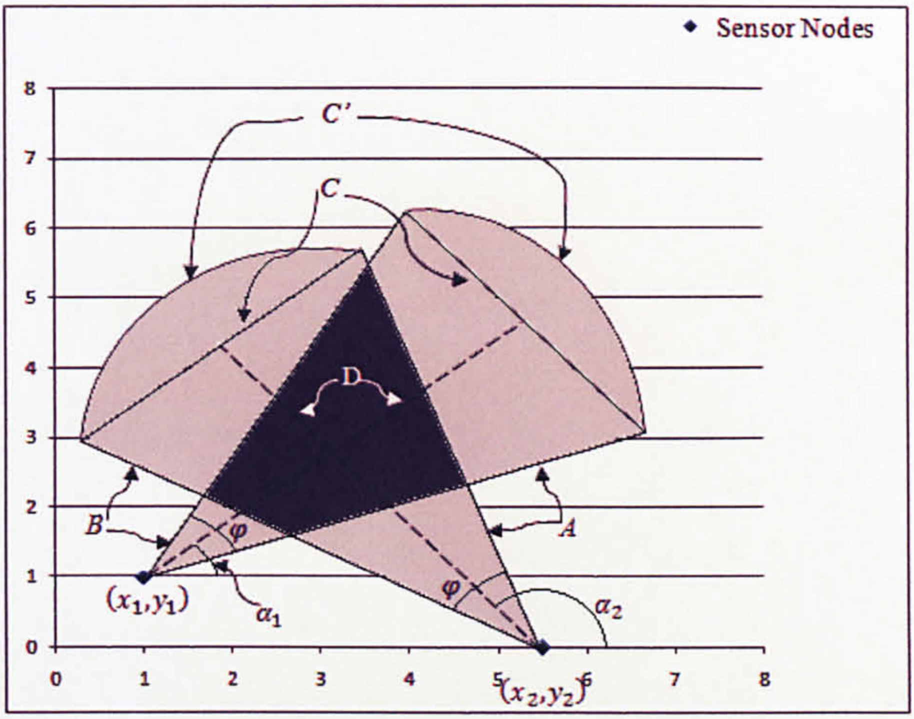


Figure 3-8: Explaination of the Node Coverage Area and the Overlapped Area Between Two Sensor Nodes

To calculate the coverage area of the sensor node, say, S_1, the following equations are used:

$$Area = {}^{1}/_{2}\; C\; D \tag{3.3}$$

$$C = 2\sin\left(\frac{\varphi}{2}\right)\, A \tag{3.4}$$

$$D = \cos\left(\frac{\varphi}{2}\right)\, A \tag{3.5}$$

$$\therefore Area = sin\left(\frac{\varphi}{2}\right)\, cos\left(\frac{\varphi}{2}\right)\, A^2 \tag{3.6}$$

On the other hand, in order to calculate the overlapping area between any two sensor nodes, S_1 and S_2, it is required to find out the intersection points of the side lines of these sensors by solving the intersection of the straight line equations of both S_1 and S_2. Knowing that the straight line equation is in the form of $y = mx + b$, for sensor S_1, the equations will be:

$$y = \tan\left(\alpha_1 + \frac{\varphi}{2}\right)x + \left(y_1 - \tan\left(\alpha_1 + \frac{\varphi}{2}\right)x_1\right) \tag{3.7}$$

$$y = \tan\left(\alpha_1 - \frac{\varphi}{2}\right)x + \left(y_1 - \tan\left(\alpha_1 - \frac{\varphi}{2}\right)x_1\right) \tag{3.8}$$

$$y = \left(\frac{\sin\left(\alpha_1 + \frac{\varphi}{2}\right) - \sin\left(\alpha_1 - \frac{\varphi}{2}\right)}{\cos\left(\alpha_1 + \frac{\varphi}{2}\right) - \cos\left(\alpha_1 - \frac{\varphi}{2}\right)}\right)x + \left(y_1 - \left(\frac{\sin\left(\alpha_1 + \frac{\varphi}{2}\right) - \sin\left(\alpha_1 - \frac{\varphi}{2}\right)}{\cos\left(\alpha_1 + \frac{\varphi}{2}\right) - \cos\left(\alpha_1 - \frac{\varphi}{2}\right)}\right)x_1\right) \tag{3.9}$$

Similarly, those for sensor S_2 will be:

$$y = \tan\left(\alpha_2 + \frac{\varphi}{2}\right)x + \left(y_2 - \tan\left(\alpha_2 + \frac{\varphi}{2}\right)x_2\right) \tag{3.10}$$

$$y = \tan\left(\alpha_2 - \frac{\varphi}{2}\right)x + \left(y_2 - \tan\left(\alpha_2 - \frac{\varphi}{2}\right)x_2\right) \tag{3.11}$$

$$y = \left(\frac{\sin\left(\alpha_2 + \frac{\varphi}{2}\right) - \sin\left(\alpha_2 - \frac{\varphi}{2}\right)}{\cos\left(\alpha_2 + \frac{\varphi}{2}\right) - \cos\left(\alpha_2 - \frac{\varphi}{2}\right)}\right)x + \left(y_2 - \left(\frac{\sin\left(\alpha_2 + \frac{\varphi}{2}\right) - \sin\left(\alpha_2 - \frac{\varphi}{2}\right)}{\cos\left(\alpha_2 + \frac{\varphi}{2}\right) - \cos\left(\alpha_2 - \frac{\varphi}{2}\right)}\right)x_2\right) \tag{3.12}$$

Solving each equations of (3.7), (3.8) and (3.9) one time with equation (3.10) and other times with equation (3.11) and equation (3.12), yields a solution of nine points that may have six points or less that comprise the corners of the overlapping area between the two sensor nodes. To decide which points participate in constructing this area, the condition that it should be part of the logical regions of both equations that are used for their solution is required. The logical pairs for different regions are:

$$\left[x_1\ \ (A\cos\left(\alpha_1-\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[x_2\ \ (A\cos\left(\alpha_2+\frac{\varphi}{2}\right)+x_2\right] \text{ for (3.7) with (3.10).}$$

$$\left[x_1\ \ (A\cos\left(\alpha_1-\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[x_2\ \ (A\cos\left(\alpha_2-\frac{\varphi}{2}\right)+x_2\right] \text{ for (3.7) with (3.11).}$$

$$\left[x_1\ \ (A\cos\left(\alpha_1-\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[(A\cos\left(\alpha_2+\frac{\varphi}{2}\right)+x_2\ \ (A\cos\left(\alpha_2-\frac{\varphi}{2}\right)+x_2\right]$$

for (3.7) with (3.12).

$$\left[x_1\ \ (A\cos\left(\alpha_1+\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[x_2\ \ (A\cos\left(\alpha_2+\frac{\varphi}{2}\right)+x_2\right] \text{ for (3.8) with (3.10).}$$

$$\left[x_1\ \ (A\cos\left(\alpha_1+\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[x_2\ \ (A\cos\left(\alpha_2-\frac{\varphi}{2}\right)+x_2\right] \text{ for (3.8) with (3.11).}$$

$$\left[x_1\ \ (A\cos\left(\alpha_1+\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[(A\cos\left(\alpha_2+\frac{\varphi}{2}\right)+x_2\ \ (A\cos\left(\alpha_2-\frac{\varphi}{2}\right)+x_2\right]$$

for (3.8) with (3.12).

$$\left[(A\cos\left(\alpha_1+\frac{\varphi}{2}\right)+x_1\ \ (A\cos\left(\alpha_1-\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[x_2\ \ (A\cos\left(\alpha_2+\frac{\varphi}{2}\right)+x_2\right]$$

for (3.9) with (3.10).

$$\left[(A\cos\left(\alpha_1+\frac{\varphi}{2}\right)+x_1\ \ (A\cos\left(\alpha_1-\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[x_2\ \ (A\cos\left(\alpha_2-\frac{\varphi}{2}\right)+x_2\right]$$

for (3.9) with (3.11).

$$\left[(A\cos\left(\alpha_1+\frac{\varphi}{2}\right)+x_1\ \ (A\cos\left(\alpha_1-\frac{\varphi}{2}\right)+x_1\right] \text{ and } \left[(A\cos\left(\alpha_2+\frac{\varphi}{2}\right)+x_2\ \ (A\cos\left(\alpha_2-\frac{\varphi}{2}\right)+x_2\right]$$

for (3.9) with (3.12).

After the CPU constructs the coverage area for all the nodes, it can calculate the percentage of the overlapping area between the two nodes given by:

$$\frac{common\ \ overlapping\ \ area}{Area\ 1{+}Area\ 2} \times 100\ \% \qquad 3.13$$

where, *Area1* and *Area 2* are the coverage areas of S_1 and S_2 respectively.

3.3 Detecting and Tracking a Moving Object in Stationary Surveillance Camera Network

In this section of the work, a video surveillance network scenario is presented where the network detects and tracks a target based on simple vision characteristic. Assuming a wireless camera network that has been randomly deployed and its nodes start localizing themselves using the recent vision based localization technique. This network starts to monitor the area and constructs video surveillance network that detect any other objects entering the sensing field. To do so, each sensor node detects the targets in its FOV and sends its calibration parameters, consisted of its coordinates (x_i, y_i) and its orientation angle α_i, to the CPU which calculates the relative coordinates and the speed of the target and tracks it using frame to frame position of the target.

3.3.1 Detecting the Target

In order to detect a target in the FOV of a few camera nodes, the frame difference technique with image filtering operations are used. These filtering operations are used to the undesired noise produced in the logical image from the frame difference technique. Based on the pinhole camera model in Figure 3-9, the angular offset ψ of a target can be found from the orientation angle α of the sensor node. This can be done using the following formula that can be easily derived from the camera's parameters:

$$\psi = tan^{-1}\left(\frac{2d}{D} tan\left(\frac{\varphi}{2}\right)\right) \qquad 3.14$$

Here, D is the horizontal width of the image plane in pixels, d is the distance in pixels between the object and the center of the image plane, and φ is the FOV angle of the camera.

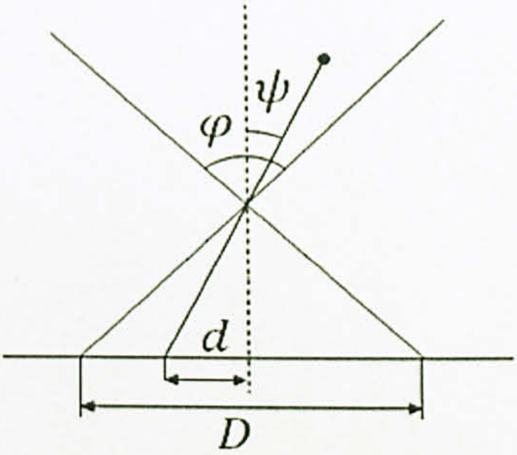
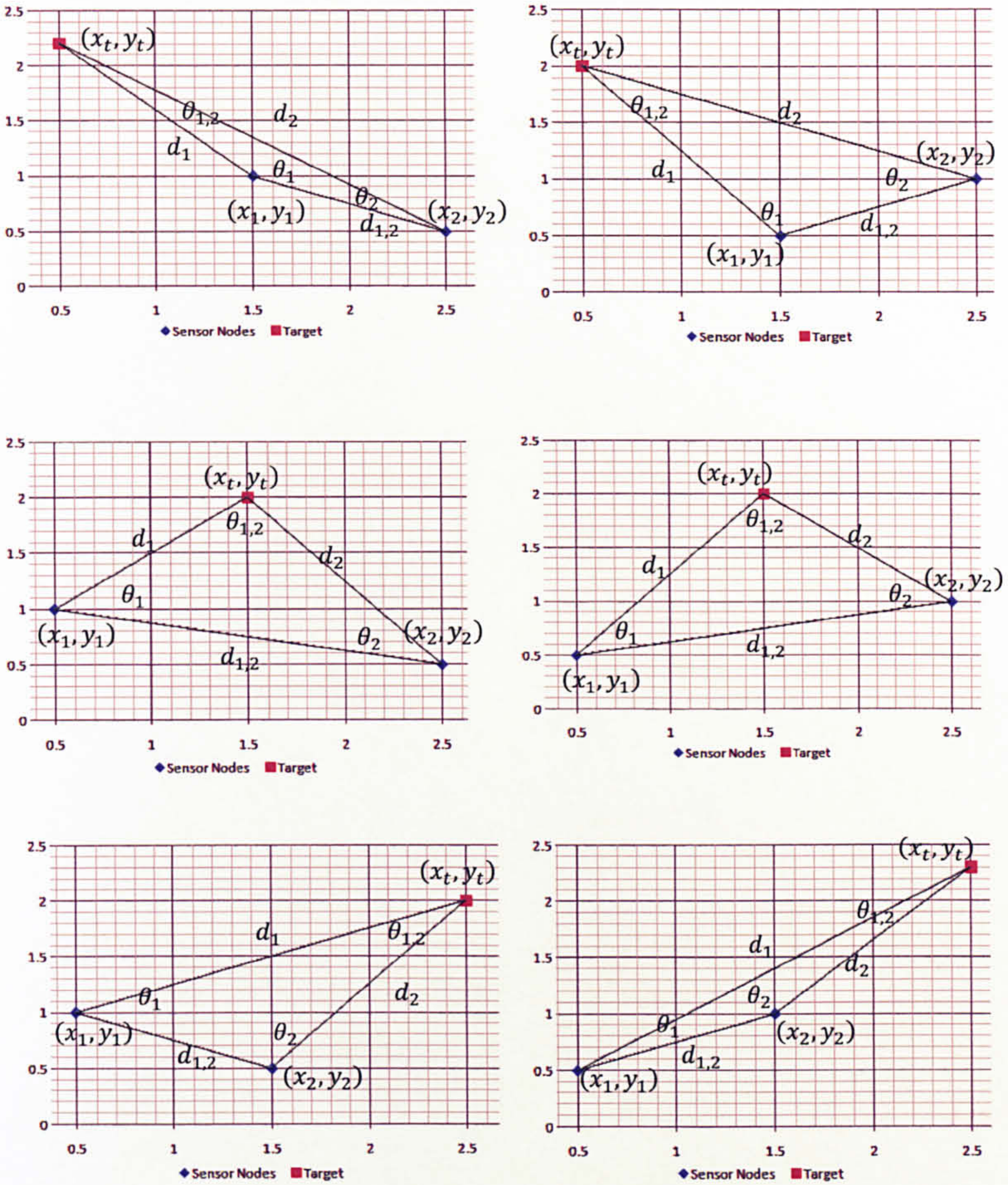


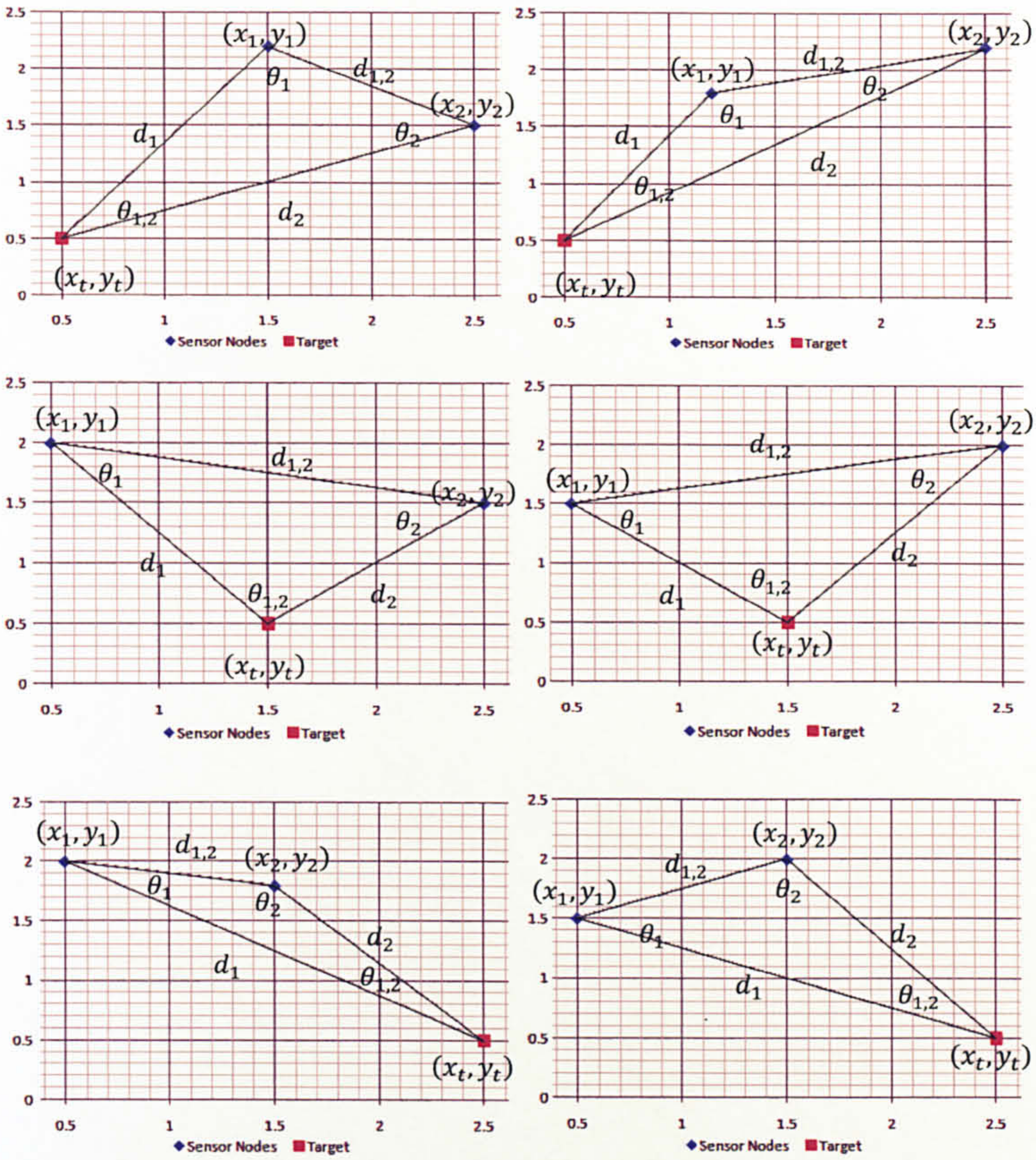
Figure 3-9: The Pinhole Camera Model

3.3.2 Localizing the Target

In this step of the scenario, the sensor node has already detected the target and sent, to the CPU, its calibration information (the coordinate, and the orientation angle) as well as the angular offset from the previous step. The CPU now starts localizing the target in the relative coordinate system after receiving two or more detection messages from the sensing field. By assuming that, at least, two sensor nodes S_1 and S_2 with calibration vectors $C_1[(x_1, y_1), \alpha_1]$ and $C_2[(x_2, y_2), \alpha_2]$ have detected a target located at (x_t, y_t), there are several relative locations possible between the nodes and the target. However, in each of these cases, a triangle between the two nodes and the target can be constructed as shown in Figure 3-10.


2.5
2
1.5
1
0.5
0
(x_t, y_t)
$\theta_{1,2}$
d_2
d_1
θ_1
(x_1, y_1)
θ_2
$d_{1,2}$
(x_2, y_2)
0.5
1
1.5
2
2.5
◆Sensor Nodes
■Target
2.5
2
1.5
1
0.5
0
(x_t, y_t)
$\theta_{1,2}$
d_2
(x_2, y_2)
d_1
θ_2
θ_1
$d_{1,2}$
(x_1, y_1)
0.5
1
1.5
2
2.5
◆Sensor Nodes
■Target
2.5
2
1.5
1
0.5
0
(x_t, y_t)
$\theta_{1,2}$
d_1
d_2
θ_1
(x_1, y_1)
θ_2
(x_2, y_2)
$d_{1,2}$
0.5
1
1.5
2
2.5
◆Sensor Nodes
■Target
2.5
2
1.5
1
0.5
0
(x_t, y_t)
$\theta_{1,2}$
d_2
d_1
(x_2, y_2)
θ_2
θ_1
$d_{1,2}$
(x_1, y_1)
0.5
1
1.5
2
2.5
◆Sensor Nodes
■Target
2.5
2
1.5
1
0.5
0
(x_t, y_t)
d_1
$\theta_{1,2}$
θ_1
d_2
(x_1, y_1)
θ_2
$d_{1,2}$
(x_2, y_2)
0.5
1
1.5
2
2.5
◆Sensor Nodes
■Target
2.5
2
1.5
1
0.5
0
(x_t, y_t)
$\theta_{1,2}$
d_1
d_2
θ_2
θ_1
(x_2, y_2)
$d_{1,2}$
(x_1, y_1)
0.5
1
1.5
2
2.5
◆Sensor Nodes
■Target

(a)



(b)

Figure 3-10: The Possible Cases of Detecting The target (a) $(\alpha_1 - \varphi_1) \,||\, (\alpha_2 - \varphi_2) < \pi$. (b) $(\alpha_1 - \varphi_1) \,\&\, (\alpha_2 - \varphi_2) \geq \pi$

All the above cases shown in Figure 3-10 can be classified into two cases for the purpose of calculating the internal angles for S_1, θ_1, for S_2, θ_2 and that between S_1 and

S_2, $\theta_{1,2}$. These angles can be calculated using the triangulation principle with the known orientation angles α_i and the angular offsets φ_i, where $i = 1,2$. These angles for the two cases are as follow:

Case 1:

$$\begin{cases} \theta_1 &= (\alpha_1 - \varphi_1) - tan^{-1}\left(\frac{y_2-y_1}{x_2-x_1}\right) \\ \theta_2 &= \pi - (\alpha_2 - \varphi_2) + tan^{-1}\left(\frac{y_2-y_1}{x_2-x_1}\right) \\ \theta_{1,2} &= \pi - (\theta_1 + \theta_2) \end{cases} \qquad 3.15$$

when $(\alpha_1 - \varphi_1)||\,(\alpha_2 - \varphi_2) < \pi$.

Case 2:

$$\begin{cases} \theta_1 &= 2\pi - (\alpha_1 - \varphi_1) + tan^{-1}\left(\frac{y_2-y_1}{x_2-x_1}\right) \\ \theta_2 &= (\alpha_2 - \varphi_2) - \pi - tan^{-1}\left(\frac{y_2-y_1}{x_2-x_1}\right) \\ \theta_{1,2} &= \pi - (\theta_1 + \theta_2) \end{cases} \qquad 3.16$$

when $(\alpha_1 - \varphi_1)\ \&\ (\alpha_2 - \varphi_2) \geq \pi$.

The Euclidean distance between the two sensor nodes is given by.

$$d_{1,2} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad 3.17$$

Also, the relationship between the angles and sides length of the triangle is given by:

$$\frac{sin\,\theta_1}{d_2} = \frac{sin\,\theta_2}{d_1} = \frac{sin\,\theta_{1,2}}{d_{1,2}} \qquad 3.18$$

Thus, from equation (3.18) the following two equations can be obtained:

$$\begin{cases} d_1 = \frac{sin\,\theta_2}{sin\,\theta_{1,2}} \times d_{1,2} \\ d_2 = \frac{sin\,\theta_1}{sin\,\theta_{1,2}} \times d_{1,2} \end{cases} \qquad 3.19$$

Now, the coordinates of the target can be calculated in the same relative coordinate system of the network as:

$$(x_i + d_i\,cos(\alpha_i - \varphi_i)\ ,\ y_i + d_i\,sin(\alpha_i - \varphi_i))\ ,\ i = 1\ or\ 2.$$

3.3.3 Target speed

Once the coordinates of the target have been obtained and updated in the relative coordinate system at a certain time, the Euclidean distance traveled by the target by the next updating time gives the target's average speed as: $v = \frac{d}{t}$, where *d* is the Euclidean distance between two updating instances of time, *t*.

3.4 Conclusion

This chapter describes in detail the proposed vision localization and calibration technique that attempts to obtain the one-plane coordinates of all the camera nodes in a video sensor network whose nodes are randomly deployed. The orientation angles of each camera node are also obtained relative to the network reference system. Furthermore, a method to allow the network to calculate the coverage and overlapping areas has been developed. Following that, an application scenario of a stationary surveillance camera network that detects and localizes a moving target is presented.

In the following chapter the experimental tests that have been carried out to investigate the proposed vision based localization and calibration technique are explained.

CHAPTER FOUR: METHODOLOGY – EXPERIMENTAL TESTING

This chapter describes, in details, the experimental tests that have been carried out to evaluate the performance of the proposed vision based localization and calibration technique. Section 4.1 introduces the experiment and explains the assumptions made in the methodology in the laboratory environment. This is followed by several experiments to establish the resolvable range that can be obtained for a camera with given resolution and camera sensor node of some fixed height; and also an experiment to establish the relationship of actual heights of objects to their image heights for fixed range and resolution. In section 4.3, the necessary practical tests are provided for various phases of the proposed localization and calibration technique. They constitute 3 different phases namely, defining the core network, localizing the remaining nodes in the network and calibrating the camera node. Section 4.4 provides experimental test for the stationary surveillance camera network that tracks and localizes an arbitrary moving object. Finally, this chapter is ended with a conclusion for the all experiments.

4.1 Experimental Methodology

In this section of the chapter, the network model and its assumptions has been provided to evolve suitable laboratory experiments that would justify the results in actual network deployment.

The proposed vision based localization and calibration technique described in the previous chapter depends mainly upon the representation of the *reference nodes* in the image-plane of the camera node that is trying to configure itself. This step requires that all the nodes in the network are parallel to each other in the image-plane of the camera node. This is feasible if all the camera nodes stand up so that their optical axes are parallel to the ground floor as shown in Figure 4-1.

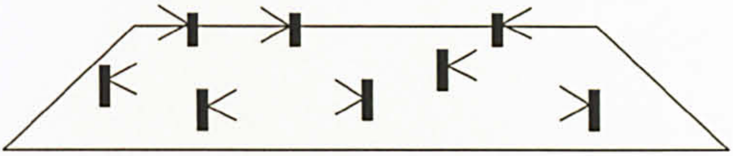


Figure 4-1: Schematic of the Model Used in the Proposed Technique

To achieve this, an innovation in the shape of the camera node is proposed so that whenever the camera node is randomly deployed, it will be standing up on the ground floor. The innovated shape is a heavy arc-shaped base in the sensor node as shown in Figure 4-2.

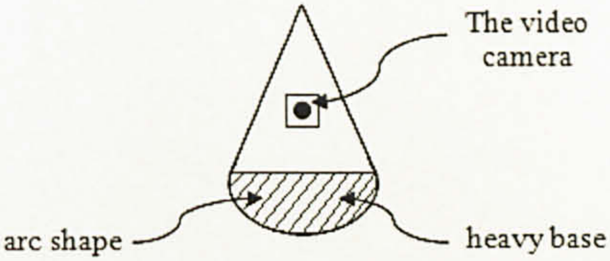


Figure 4-2: The Camera Node Shape

For localization purposes, as the real height *h* of the camera node and its representation in the image-plane is relied, it is assumed that both the real height *h* of the camera node and the shape of the camera nodes are same throughout the network and are a-priori known. The assumption of the shape, used, to help the camera node stand up and be easily detected in the image-plane is pre-requisite to successful vision-based technique.

In laboratory experiments, however, known objects of certain shape are used on flat table-top as shown in Figure 4-3. The Sony camera of 0.3 Megapixels is used, as the real camera nodes are not available in the laboratory.

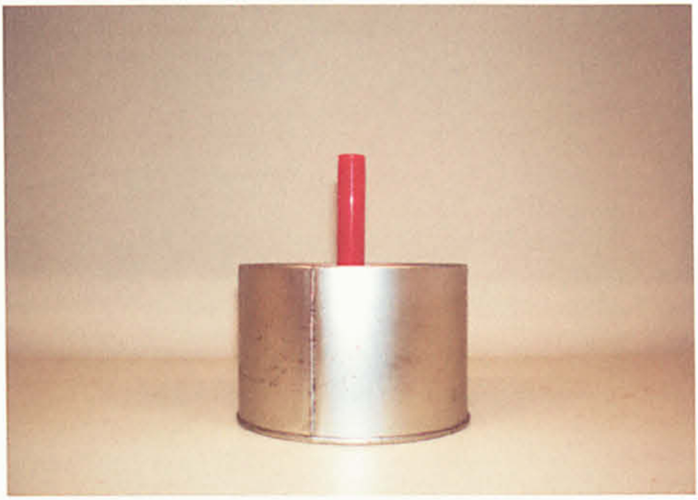


Figure 4-3: Object with Chromatic Sign Used in Instead of the Real Camera Node

4.2 Testing the Physical Constraints

Since the camera resolution is implicitly used in estimating the distance of the nodes, it is important to establish the relationship of measured heights in the image-plane to estimate the distances and illustrate the impact of the camera node resolution on this relationship. Another reason behind these experiments is to determine the minimum and the maximum resolvable distances (the lower and the upper boundaries of the resolvable distances) between an arbitrary object and the camera nodes for given acceptable estimation error. In the laboratory experiments, the upper-bound on the estimated distance is obtained when the distance variation of 10 cm represents, no change (*zero* variation) in the image height. Even though the value of maximum accepted error, 10 cm, is arbitrary, it represents a practical value for the estimated distances of the order of 5 m (about 2% error) for the given object height and camera node resolution. The following two subsections provide a detailed description on how these experiments are conducted and the relationship each experiment establishes.

4.2.1 Determination of the Upper Bound of the Resolvable Range

This subsection provides a practical test that illustrates the relationship of the camera resolution in resolving the distances between the camera node and an arbitrary object of fixed height 6 cm. The purpose of this test is to determine, for a given camera resolution, the upper bound of the estimated distance – the largest distance that can be accurately estimated by resolving the image height of a chosen object in the image plane. Two camera nodes with different resolutions are used to estimate several distances between the camera node and the object. Initially, the camera node with 2 Megapixels resolution (1632x1224 pixels) and 621.48 mm focal length is used to obtain the upper bound. Similarly, this is repeated for another camera node with 0.5 Megapixels resolution and 426.21 mm focal length.

In this work, 33 observations are taken at different distances using the 2 camera nodes, mentioned above, the image heights are obtained from the image-plane to estimate the distances using equation (2.6). Table 4-1 and Table 4-2 show the results from the 2 experiments.

Table 4-1: The Estimated Distances Using 2 Megapixel Resolution Camera

No.	Real Distance(cm)	Image height (cm)	Estimated Distances (cm)
1	60	6.279	59.38
2	70	5.398	69.09
3	80	4.657	80.06
4	90	4.057	91.91
5	100	3.739	99.72
6	110	3.422	108.97
7	120	3.104	120.11
8	130	2.858	130.49
9	140	2.681	139.08
10	150	2.505	148.87
11	160	2.328	160.15
12	170	2.223	167.78
13	180	2.081	179.15
14	190	1.976	188.75
15	200	1.870	199.43

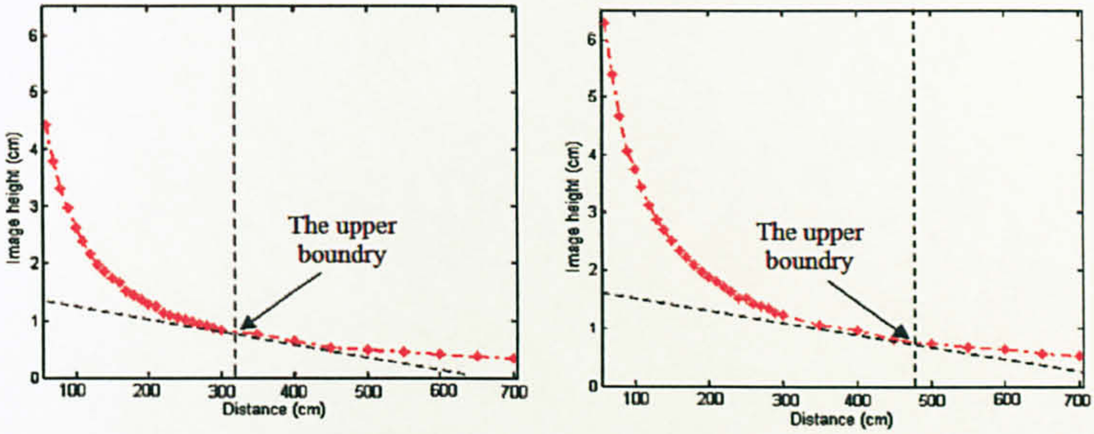
16	210	1.799	207.25
17	220	1.693	220.21
18	230	1.623	229.78
19	240	1.517	245.81
20	250	1.517	245.81
21	260	1.411	264.25
22	270	1.376	271.03
23	280	1.341	278.16
24	290	1.270	293.61
25	300	1.235	302.00
26	350	1.058	352.33
27	400	0.953	391.48
28	450	0.811	459.57
29	500	0.741	503.33
30	550	0.670	556.32
31	600	0.635	587.22
32	650	0.564	660.63
33	700	0.529	704.67

Table 4-2: The Estimated Distances Using 0.5 Megapixels Resolution Camera

No.	Real Distance(cm)	Image height (cm)	Estimated Distances (cm)
1	60	4.422	57.83
2	70	3.780	67.66
3	80	3.288	77.77
4	90	2.948	86.74
5	100	2.608	98.06
6	110	2.381	107.40
7	120	2.154	118.70
8	130	1.965	130.12
9	140	1.852	138.08
10	150	1.739	147.09
11	160	1.663	153.77
12	170	1.512	169.15
13	180	1.436	178.05
14	190	1.361	187.95
15	200	1.285	199.00
16	210	1.247	205.03
17	220	1.134	225.54
18	230	1.096	233.31
19	240	1.058	241.65
20	250	1.020	250.59
21	260	0.983	260.23

22	270	0.945	270.64
23	280	0.907	281.92
24	290	0.869	294.18
25	300	0.831	307.55
26	350	0.756	338.30
27	400	0.643	398.00
28	450	0.529	483.29
29	500	0.491	520.47
30	550	0.452	563.84
31	600	0.416	615.10
32	650	0.378	676.61
33	700	0.340	751.78

From Table 4-1 and Table 4-2, it is shown that the image height *h'* decays with respect to the real distance *d* between the object and the camera node of fixed resolution as shown in Figure 4-4. For larger distances, the variation of the image height against distance is small. For a chosen estimation error of 10 cm, the distance that shows no change in image height is defined as the upper bound of the estimation distance boundaries. From Figure 4-4-a and Figure 4-4-b, the distances 320 cm and 480 cm are the upper bounds for the camera node resolution of 0.5 MegaPixels and 2 MegaPixels respectively.



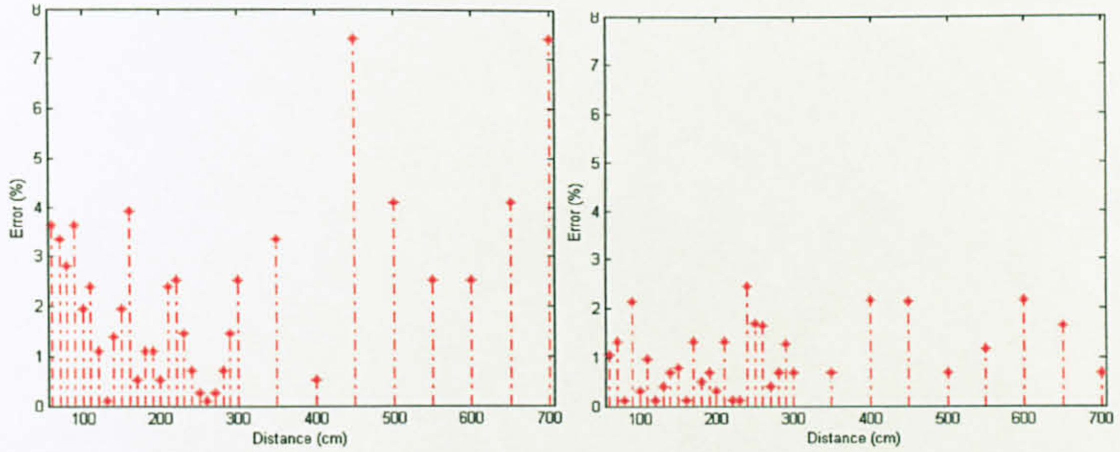
(a) Using 0.5 MegaPixels (b) Using 2 MegaPixels

Figure 4-4: Variation in the Image Heights and Defining the Upper Distance Boundary (a) Using 0.5 Megapixels Resolution Camera (b) Using 2 Megapixels Resolution Camera

Figure 4-4 illustrates that by increasing the resolution of the camera node the resolvability of the image heights can be increased, which in turn, will increase the upper bound of the estimation distance boundary and the ability of estimating objects within that range with good accuracy. This boundary represents the lengths of the sides in the definition of FOV angle, as mentioned in chapter 3.

Furthermore, the observations from Table 4-1 and 4-2 are used also to calculate the relative error values for all the observations and plot them in Figure 4-5. We obtain an average relative error of 0.96 % and 2.22 % and an average error of 2.93 cm and 7.22 cm using 2 MegaPixels and 0.5 MegaPixels camera resolutions respectively. Figure 4-5-a shows that, at the resolution of the camera representing video standard output (0.5 Mpixels), we get relative error less than 5 % most of the time.

Moreover, each camera also has a lower bound of the estimation distance that determines the minimum distance at which a given object completely fills the image-plane vertically. The next subsection describes the experiment that helps in obtaining this value.



(a) Using 0.5 MegaPixels

(b) Using 2 MegaPixels

Figure 4-5: The Relative Error of the Estimated Distance (a) Using 0.5 Megapixels Resolution Camera (b) Using 2 Megapixels Resolution Camera

4.2.2 Determination of the Lower Bound of the Resolvable Range

Logically, if the distance between any arbitrary object and the camera node is closed, at some distance the image of the object will fill the whole image-plane. This can be justified theoretically using the physical laws of the lenses, where the object's image keeps increasing in the image-plane until it reaches the largest possible size when the distance between the object and the camera is reduced to the focal length. This is due to the fact that the refracted rays, from the lens, are parallel to each other as shown in Figure 4-6 when the object is at its focus.

Since only the fixed number of pixels occupies the image-plane, practically the image of an object, in some cases, will fill the image-plane even before the object reaches the focal length point. In these cases, the object is considered as big and giant, which means that the range of the resolvable range is decreased and the lower bound is the distance where the image of the object fills the image-plane.

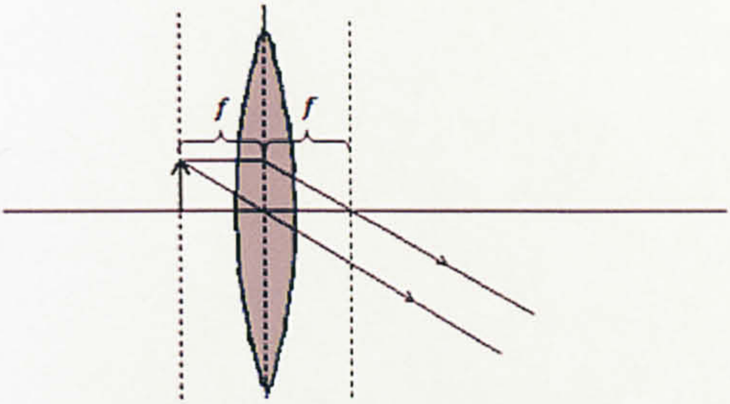


Figure 4-6: Explaination for the Imaging of an Object Stands in the Focal Length

In this subsection, a practical test has been implemented that demonstrates the variation of the image heights for different objects of varying real heights to determine the lower bound of the distance between the object and the camera node with given resolution. Many observations have been taken for 3 different objects with heights of

350 cm, 81 cm, and 6 cm respectively and the image heights have been extracted from the image-plane to determine the lower distance bound between the 3 objects and a camera node with resolution of 640×480 Pixels as shown in Figure 4-7.

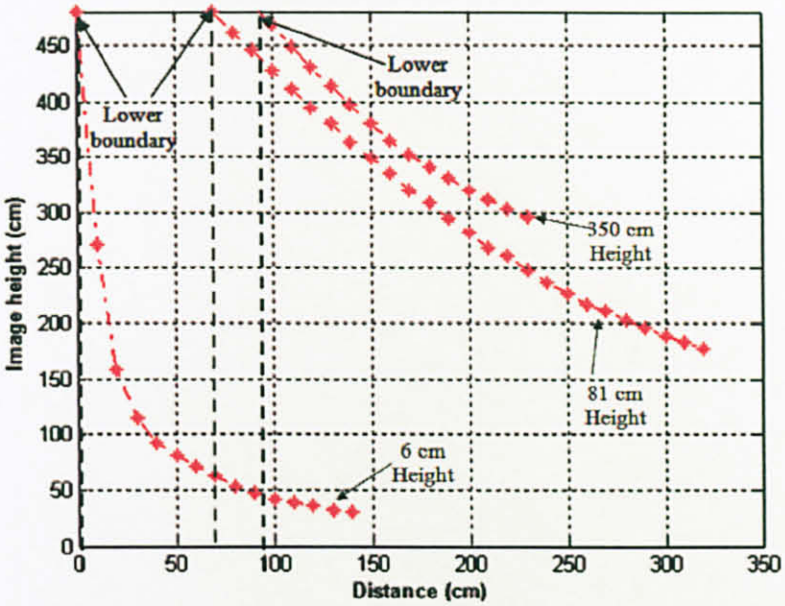


Figure 4-7: Variation in the Image Heights for 3 Objects With Different Heights and their Lower Distance Boundaries

Figure 4-7 shows that the images for the first 2 objects fill the image-plane vertically before they reach the focal length of the camera nodes. Lower bounds to be equal to 70 cm and 95 cm are obtained for the 2 objects of height 81 cm and 350 cm respectively. Figure 4-7 also shows that the lower bound for the 3rd object with height of 6 cm is equal to the focal length.

From these 2 tests, in this section, the resolvable distance range is obtained to be restricted between the focal length of the camera node as lower distance boundary, which is 42.621 cm, and the distance of 350 cm as upper distance boundary.

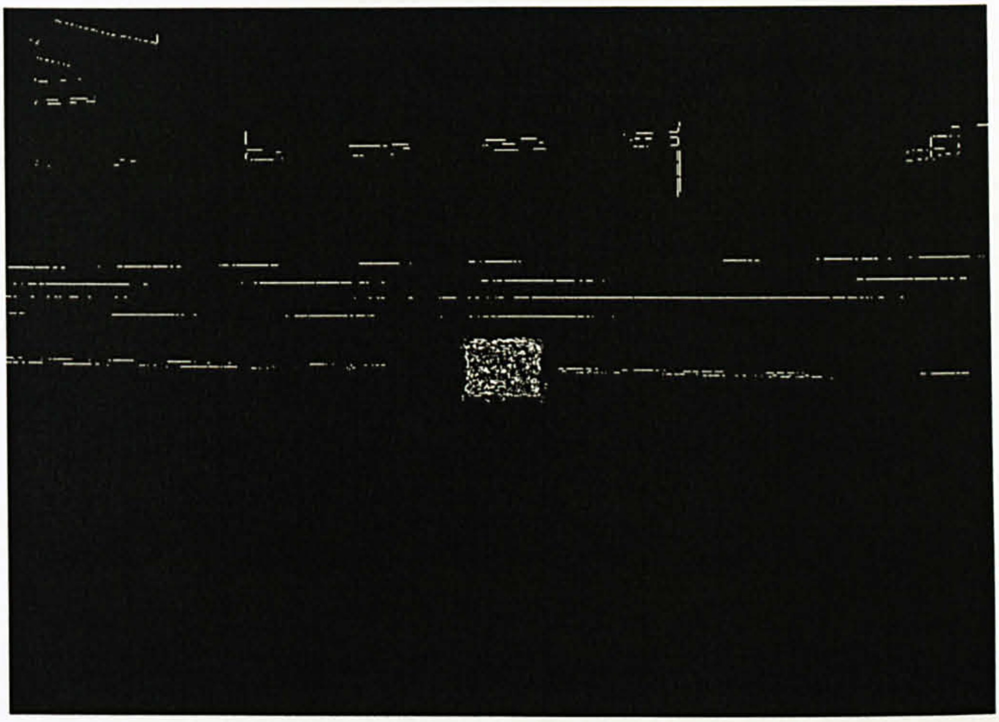
4.3 Testing the Proposed Vision Technique

In this section, practical tests are carried out, demonstrating the proposed vision based localization and calibration technique, first using the 2 MegaPixels camera network with a resolution of 1632×1224 Pixels and 621.48 mm focal length and later using a camera network with video standard display of 640×480 Pixels (0.3 MegaPixels) with focal length of 426.21 mm. In the following subsections, various phases of testing the proposed vision technique will be described in detail for configuring a camera network of 2 MegaPixels cameras only. The comparison between camera networks with cameras of another resolution will be discussed in chapter 5.

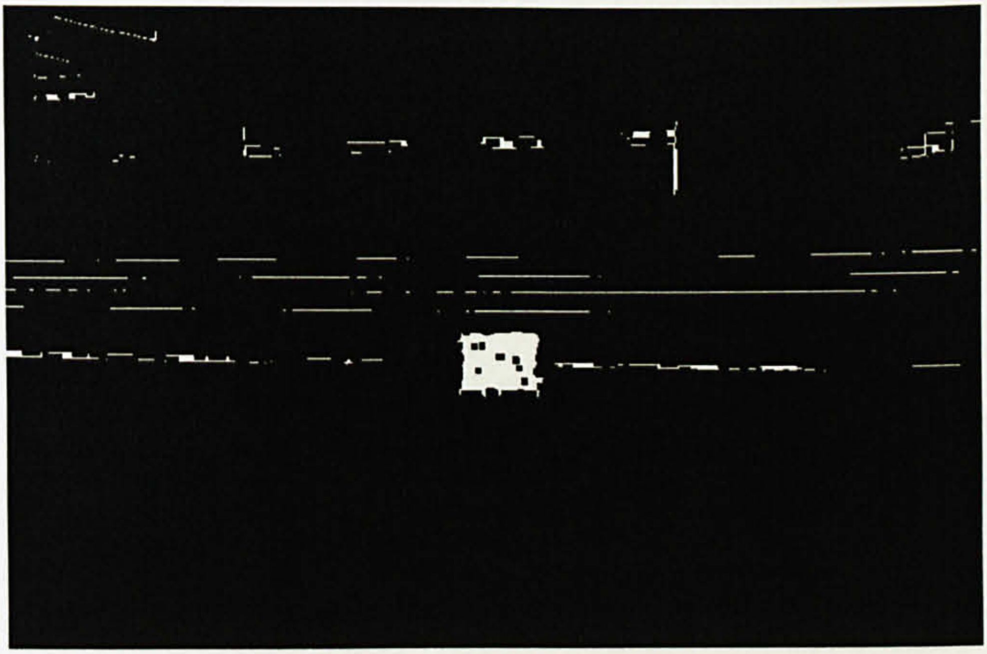
4.3.1 Phase I: Defining the Core of the Network

This phase is about defining and obtaining the coordinates of the network's core that consists of one *master node* and two *reference nodes*. One camera node has been chosen to be the *master node* and programmed to assume itself at the origin of the coordinate system. This chosen node starts searching in its FOV for two objects of 6 cm real height. These objects are assumed to represent the other two *reference nodes*.

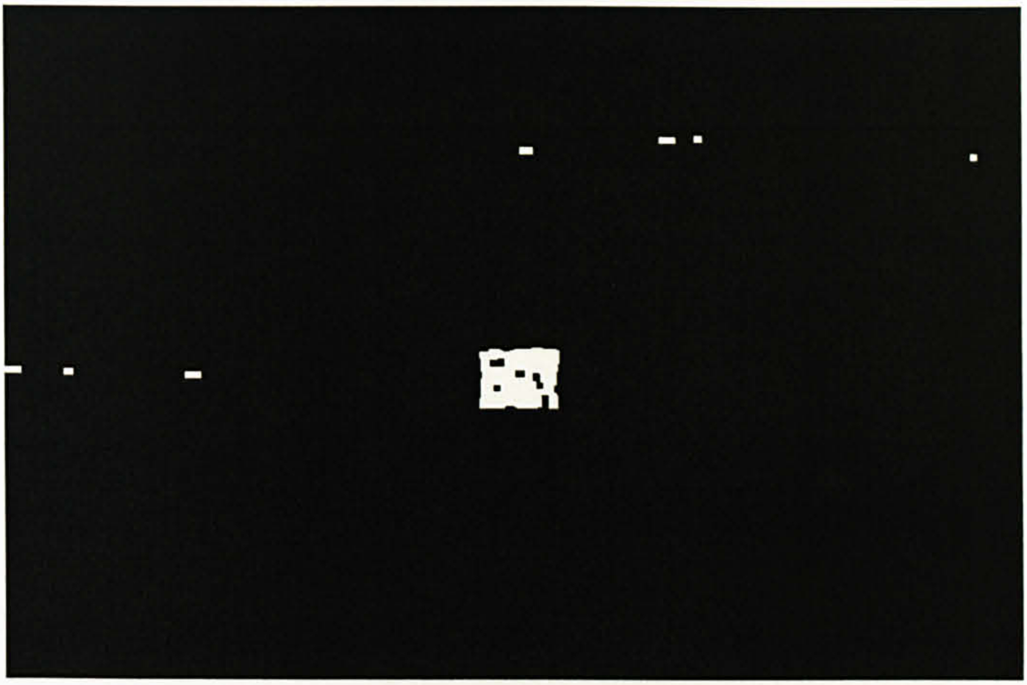
It is assumed that the reference nodes exhibit both *green* and *red* signs (vertical patches turned *green* and *red*) so they can be distinguish from the other nodes. Relying on the a-priori known chromatic shape (both chromatic signs *red* and *green* turned ON and also some edge features) of the *reference nodes*, the *master node* can detect and recognize them from the rest of the network in its FOV as shown in Figure 4-8-a., Next, the morphological closing operation is performed on the logic image to close the nearby pixels in connected groups, so that the object of interest is segmented clearly as in Figure 4-8-b. Then, the morphological opening operation is performed to open the non-grouped pixels and they are shown to disappear as in Figure 4-8-c. To remove the small connected components that represent noise in the image and keep only the large component that represents the object of interest (the *reference node*), filtering is performed and shown in Figure 4-8-d.



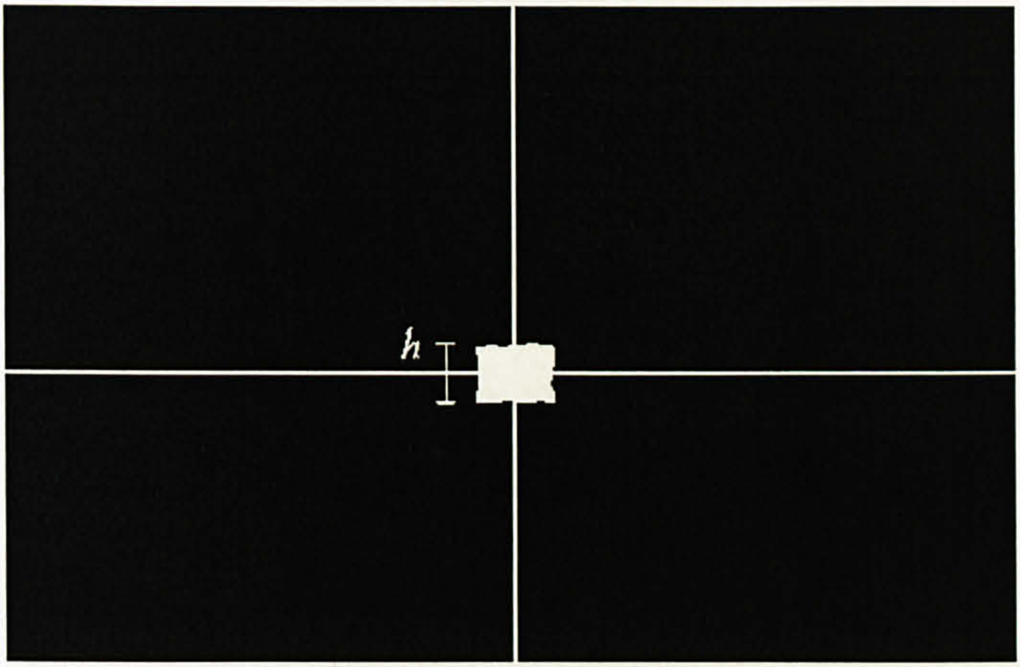
(a) Using Color and Edge Detections



(b) Morphologically Close Image



(c) Morphologically Open Image



(d) Remove the Small Connected Components and Extract the Sensor Node

Figure 4-8: Sequence of Detecting and Extracting the Reference Node (a) Detect the Node By the Mean of Color and Edge Detection (b),(c), and (d) Filtering the Detected Node and Extract it From the Image Plane

The two *reference nodes* are extracted in the image plane of the *master node* and $h'_1$and h'_2 are obtained at the angles θ_1 and θ_2 respectively. By substituting the image's height values in the vision characteristic equation (2.6), the distances d_1, and d_2 are estimated as shown in Table 4-3. These measurements are repeated 4 times to investigate the performance of this step as is discussed in chapter 5.

Table 4-3: The Estimated Distances Between the Reference Nodes

No	$\theta_1(^\circ)$	$\theta_2(^\circ)$	h_1(mm)	h_2(mm)	d_1(m)	d_2(m)
1	50.4	151.2	10.94	16.58	3.41	2.25
2	113.4	340.2	16.23	11.64	2.30	3.20
3	213.3	242.1	10.58	16.58	3.52	2.25
4	45	332.1	26.81	16.58	1.39	2.25

By using the results in Table 4-3 the coordinate systems are obtained and shown in Table 4-4 and Table 4-5, where Table 4-5 shows the relative coordinate systems of the core network at the time of starting this estimation step (the initial coordinate systems assumed by the *master node*) and Table 4-4 shows the coordinate systems at the end of this estimation step (the final coordinate systems as represented in the CPU).

Table 4-4: The Final Coordinate System

	Real Coordinates		Estimated Coordinates	
No	Sensor 2	Sensor 3	Sensor 2	Sensor 3
1	(3.20,0)	(-0.47,-2.19)	(3.41,0)	(-0.43,-2.21)
2	(2.24,0)	(-2.24,2.24)	(2.30,0)	(-2.18,2.34)
3	(3.61,0)	(1.94,-1.11)	(3.52,0)	(1.97,-1.10)
4	(1.41,0)	(0.69,2.12)	(1.39,0)	(0.66,2.15)

Table 4-5: The Initial Coordinate System

No	Real Coordinates		Estimated Coordinates	
	Sensor 2	Sensor 3	Sensor 2	Sensor 3
1	(2.5,2)	(1,-2)	(2.61,2.19)	(1.09,-1.97)
2	(2,-1)	(-1,3)	(2.11,-0.92)	(-1.07,3.02)
3	(-2,-3)	(-2,-1)	(-1.92,-2.96)	(-1.99,-1.06)
4	(1,1)	(-1,2)	(0.98,0.98)	(-1.06,1.99)

Both are same representation of the real geographical locations of the nodes but from different angle. In other words, the initial coordinate systems can be constructed by rotating-back the final coordinate systems by an angle equal to the first searching angle β_1 as shown in Figure 4-9.

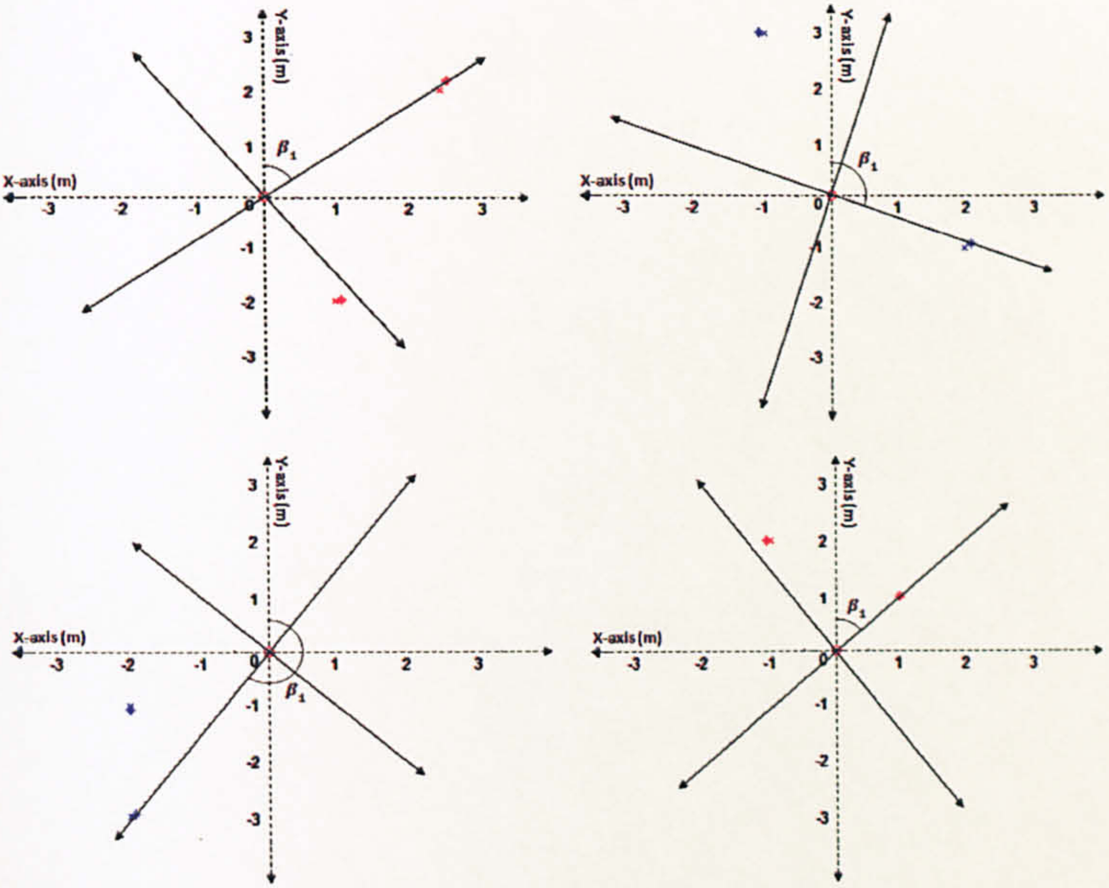


Figure 4-9: The Constructed Coordinate Systems

4.3.2 Phase II: Localizing the Rest of the Nodes in the Network

In these experiments, the coordinates of all the other nodes are obtained by extending this vision-based technique to the entire network.

Assuming that, after the coordinates of the core network have been discovered, the *master reference node* will broadcast its coordinates to the entire network. The camera nodes that receive this broadcast message store this coordinates and start searching in their FOV for the *master node*. The master node can be recognized by the *red* and *green* chromatic signs turned ON. When any node detects the *master reference node*, it extracts it from the image-plane and gets the corresponding image height h'_1 at the angle $\beta_{i,1}$.

After that, the other two *reference nodes* (the *2nd reference node* first then the *3rd reference node* after that) will repeat broadcasting their coordinates allowing the remaining camera nodes to repeat the steps mentioned above and get h'_2, and h'_3 at the angles $\beta_{i,2}$ and $\beta_{i,3}$ respectively.

The experiment is started by searching for 3 known objects of 6 cm height that are assumed to represent the 3 *reference nodes* with the same detection and extraction method as used in *phase I*. The results in Table 4-6 are considered being the information from one sensor node only and more results will be shown in chapter 5 to evaluate the vision technique as a localization method. From this information the node's coordinates can be estimated using equation (3.2).

Table 4-6: Information Used to Localize the Node

Information from the Camera Node								
The Reference Nodes Coordinates								
$S_1(0,0)$			S_2 $(4.09,0)$			S_3 $(2.95,-1.02)$		
Angles (°)			Heights (mm)			Distances (m)		
$\beta_{4,1}$	$\beta_{4,2}$	$\beta_{4,3}$	h'_1	h'_2	h'_3	d_1	d_2	d_3
277.2	148.5	186	16.58	16.23	16.58	2.25	2.3	2.25
Real Coordinate (2,1)				Est. Coordinate (2.02,1.06)				

4.3.3 Phase III: Calibrating the Camera Node

Since the visual information, provided by the camera nodes, presents partial scene from the world-view –a view that is seen in the camera nodes' FOV, it is important to obtain the orientation angles α_i of the camera nodes in the coordinate system relative to the core of the network. This knowledge together with their coordinates can help the remote server to construct the overall scene by suitable union of all the partial scenes. Calculating the orientation angles is prerequisite to constructing the overall scene and helps complete the calibration process. Therefore, the calibration parameters C_i is defined, for each camera node, consisting of its coordinates and its orientation angle $C_i[(x_i, y_i), \alpha_i]$.

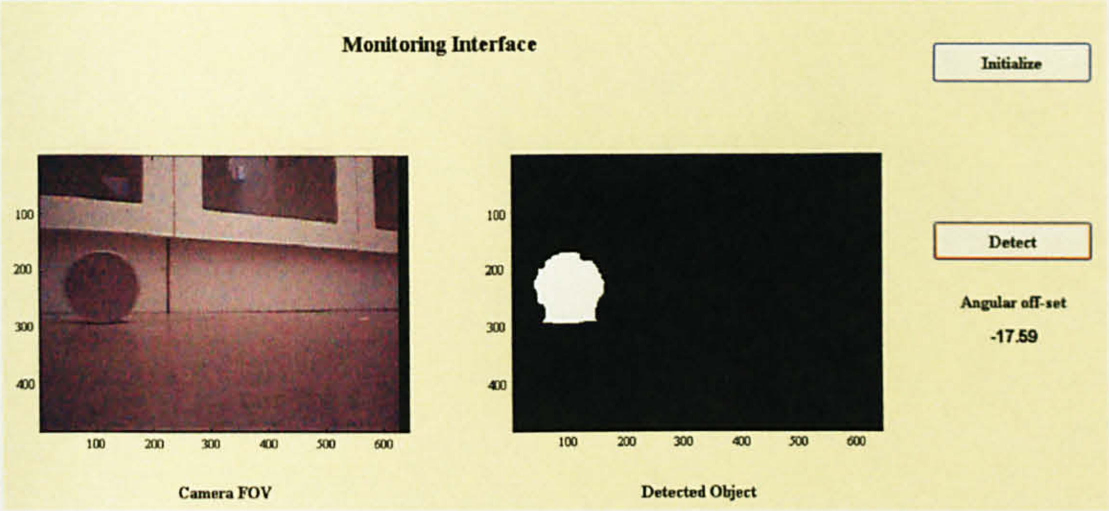
In this phase, the previous information in Table 4-6 is used to complete the calibration process of one node using normal calculations. The estimated coordinates (2.02,1.06) is used to decide that the camera node is lying in the 1st quadrant of the relative coordinate system. The coordinates is used, also, to investigate the angle where the camera node is lying relative to the origin of the coordinate system. Coupled with the knowledge of the angle the camera node used to search for the master node, in this case, the searching angle $\beta_{4,1} = 277.2^{\circ}$, the relative orientation angle α_4 of the camera node can be calculated as mentioned in section (3.2.3.1). In the present case, it is estimated to be equal to 124.69° while the actual orientation angle is 125°. In addition, given these calibration parameters, the coverage area of this camera node using equation (3.6) is calculated to gives an area equal to 3.67 m^2.

4.4 Stationary Case of Tracking Network

In this experiment, the same camera network that has been configured and calibrated in the previous section with the proposed vision based localization technique is used. The resolvable distance of this network is in the order of 3m, the core network can monitor an area of 8×8 m^2. An arbitrary object is allowed to pass through this sensing area. Two different nodes calibrated as $C_1[(-0.43,-2.21),70.2^{\circ}]$, and $C_2[(3.4,0.0),150.3^{\circ}]$ locate and track the target object as follow:

4.4.1 Detecting and Locating the Target

Since the scene viewed by any given camera is stationary, a new target object moving the FOV can be detected by the frame difference technique. Figure 4-10 shows that the camera nodes detect the target in their FOV and calculate the angular off-set from their respective orientation.



Monitoring Interface

Angular off-set
-17.59

Figure 4-10: Snapshot of the Sensor Node Monitoring Interface

This experiment obtains and estimates angular off-sets equal to $\varphi_1 = 18.98^\circ$, $\varphi_2 = 12.10^\circ$ from S_1 and S_2 respectively.

4.4.2 Localizing the Target

In order to localize the target, the distance $d_{1,2} = 4.43$ m between the two camera nodes and the slop of the line $tan^{-1}\left(\frac{y_2-y_1}{x_2-x_1}\right) = 0.52\, rad = 29.92^\circ$ are obtained. Using the equations in (3.14) the angles that the target makes with the nodes relative to the common line through the nodes are calculated. These angles $\theta_1, \theta_2, \theta_{1,2}$ are defined and labeled in Figure 4-11. After calculating these angles, equation (3.21) is used to get the relative distances $d_1 = 4.21\, m$, and $d_2 = 1.61\, m$ from the two nodes respectively. Finally, the target can be localized in the relative coordinate system at the point (2.21,1.07), while the actual coordinates are (2.10,0.99) as shown in Figure 4-11.

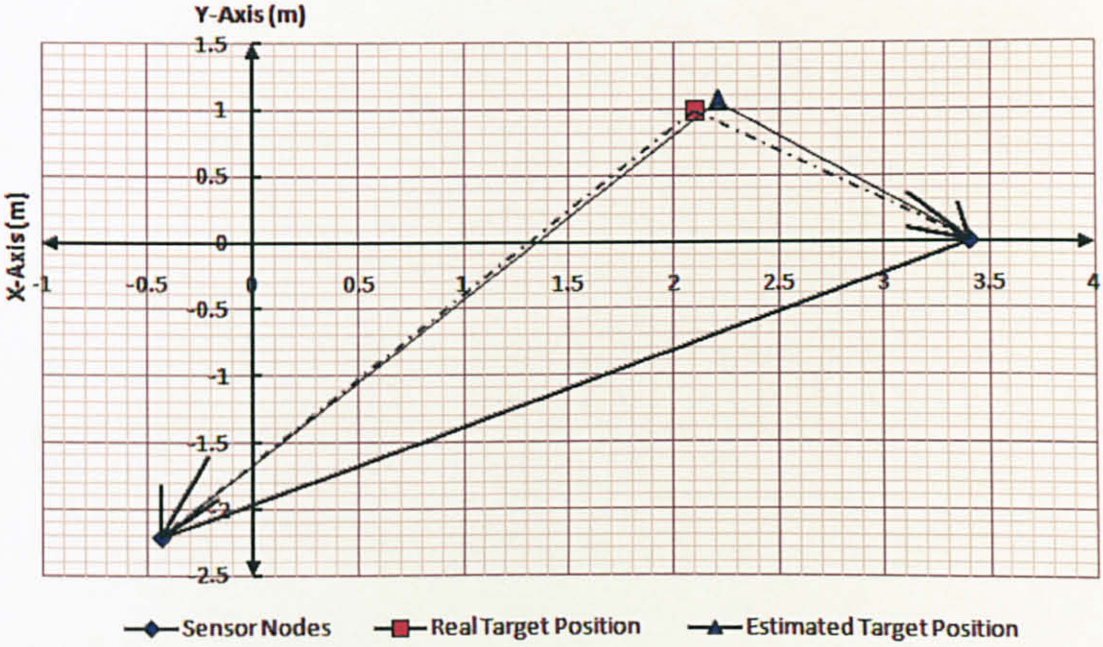


Figure 4-11: Localize the Target in the Sensing Field

In summary, the vision-based localization and tracking technique works in 3 stages. The first stage is where a predetermined node takes the role of master node and helps locate 2 other reference nodes that form the core of the network. In the second stage, the core network helps other nodes to self locate and calibrate the entire network in terms of the common area they see together. In the last stage, they detect any moving target and locate and track it as it moves within the network.

4.5 Conclusion

In this chapter, the proposed vision-based method has been described for localizing and tracking purposes. The chapter starts by explaining the network model and some of the practical assumptions that justify the results in actual network deployment. The upper and the lower bounds of the resolvable distance range have been obtained by experimental testing. Moreover, an investigation on the impact of the camera resolution on the resolvable range has been explained also. This is followed by the description of the vision based localization and calibration technique consisting of 3 stages namely: defining the core of the network, localizing the remaining nodes in the network, and calibrating the camera nodes by estimating its orientation angle.

In addition, an application of using stationary camera nodes that are, typically, configured using the proposed vision based localization and calibration technique as a surveillance network is presented. This network tracks an arbitrary objects moving around the sensing field. The steps of the detection and localization process have been provided also.

The next chapter analyzes the results obtained from this chapter and discusses the performance of this vision based localization and calibration technique.

CHAPTER FIVE: ANALYSIS AND DISCUSSION

In the last chapter, one-plane camera network has been localized and calibrated in order to demonstrate the feasibility of the proposed vision based localization and calibration technique. This chapter provides analysis and discussion for the results obtained in the previous chapter. The next section presents the influence of the resolution of camera nodes on the accuracy of localizing 1-tier camera network. Section 5.2 discusses the robustness of the proposed technique in multi-tier camera network against the error accumulated from one tier to the following tiers. Section 5.3 provides analytical study of the performance a stationary surveillance camera network that detects and tracks an arbitrary moving object. In section 5.4, a comparison with some other vision based and wireless signal based localization techniques is presented with a comparative statistical studies to amplify the distinctions of the proposed technique. Finally, a brief conclusion is provided for the all experiments that have been carried out.

5.1 The Issue of Camera Resolution

In this section, the influence of the camera resolution on the accuracy of estimating the coordinates of the camera nodes is studied using the proposed vision based localization and calibration technique. To do so, 1-tier camera networks have been deployed, consisting of 12 nodes including the *reference nodes*. First, a deployment of a 2 MegaPixels camera nodes have been obtained and later a 0.3 MegaPixels (video output of standard 640 × 480 pixels) camera nodes. By applying the proposed technique in both networks, the coordinates of the nodes are estimated as shown in Table 5-1 and Table 5-2 respectively.

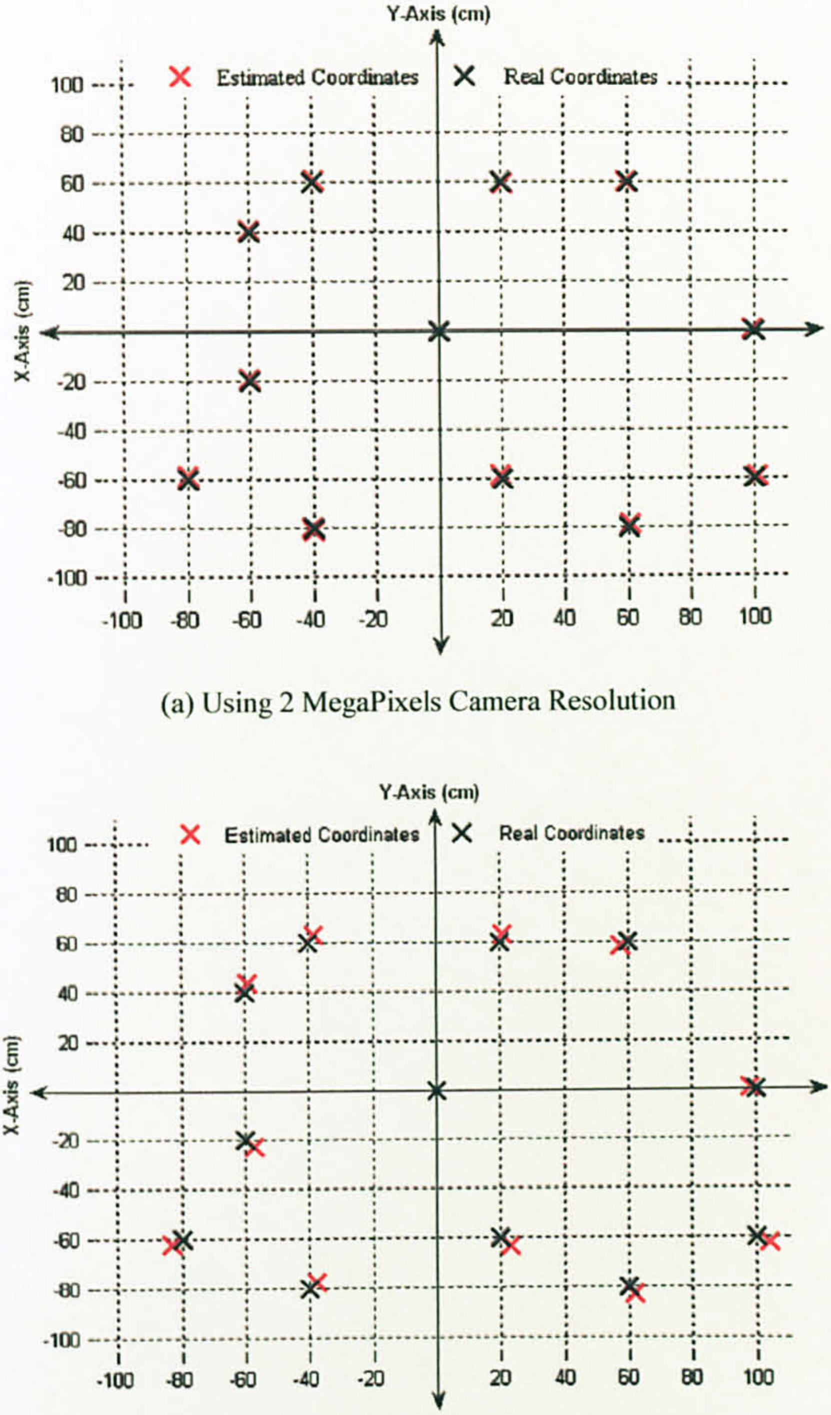
Table 5-1: Estimated Coordinates for the Nodes in the 1st Tier of 2 MegaPixels Camera Network

Node No.	Real Coordinates	Estimated Coordinates	Error (cm)
1	(0.0 , 0.0)	(0.0 , 0.0)	0
2	(100 , 0.0)	(99.72 , 0.0)	0.28
3	(-40 , -80)	(-40.11 , -80.06)	0.13
4	(60 , 60)	(59.40 , 60.38)	0.71
5	(20 , 60)	(20.30 , 59.38)	0.69
6	(-40 , 60)	(-39.11 , 60.91)	1.27
7	(-60 , 40)	(-60.24 , 40.78)	0.82
8	(-60 , -20)	(-60.15 , -19.32)	0.70
9	(-80 , -60)	(-79.91 , -58.91)	1.09
10	(-40 , -80)	(-39.78 , -81.34)	1.36
11	(60 , -80)	(60.39 , -78.81)	1.25
12	(100 , -60)	(101.03 , -59.81)	1.05

Table 5-2: Estimated Coordinates for the Nodes in the 1st Tier of 640 × 480 Pixels Camera Network

Node No.	Real Coordinates	Estimated Coordinates	Error (cm)
1	(0.0 , 0.0)	(0.0 , 0.0)	0
2	(100 , 0.0)	(98.06 , 0.0)	1.94
3	(-40 , -80)	(-37.95 , -77.77)	3.03
4	(60 , 60)	(57.83 , 58.00)	2.95
5	(20 , 60)	(20.92 , 62.92)	3.06
6	(-40 , 60)	(-38.23 , 62.84)	3.35
7	(-60 , 40)	(-59.00 , 43.47)	3.61
8	(-60 , -20)	(-57.10 , -22.61)	3.90
9	(-80 , -60)	(-83.18 , -62.11)	3.82
10	(-40 , -80)	(-44.00 , -77.98)	4.80
11	(60 , -80)	(61.65 , -83.04)	3.46
12	(100 , -60)	(104.18 , -62.34)	4.79

Using these results, the network topologies are constructed for both camera networks as shown in Figure 5-1. This figure, also, shows that the estimated coordinates using the proposed vision based localization and calibration technique are close enough to the real ones.

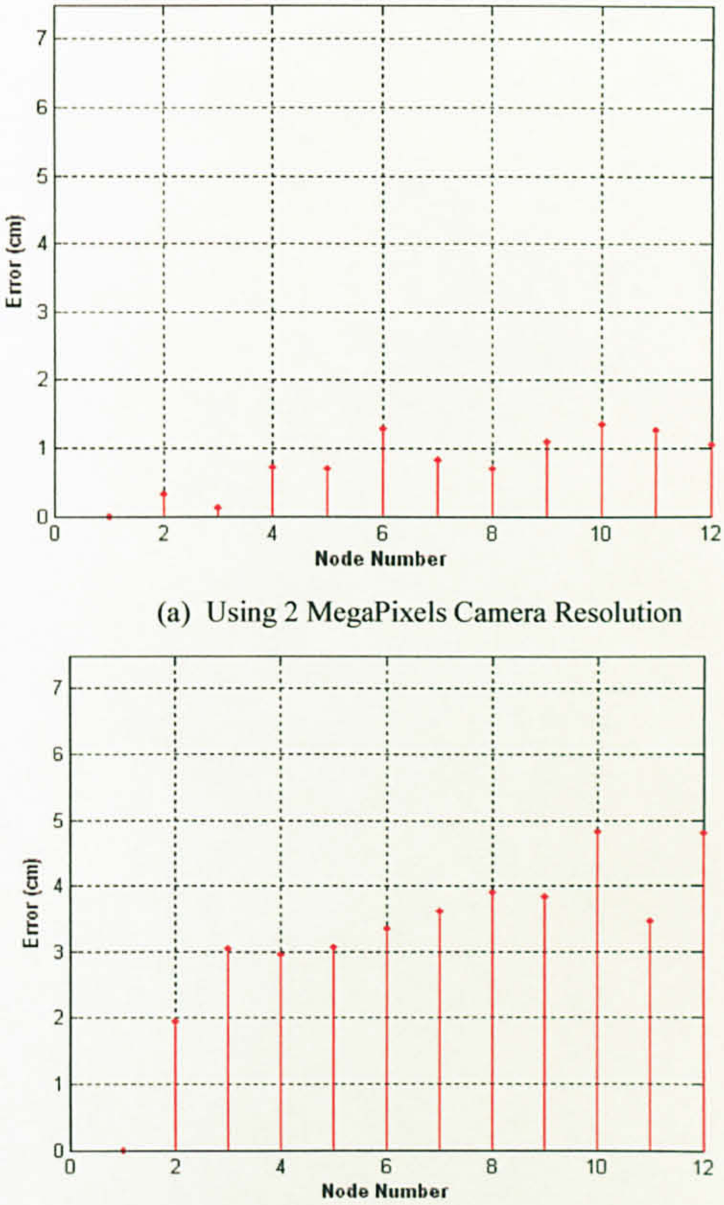


(a) Using 2 MegaPixels Camera Resolution

(b) Using 640 × 480 Pixels Camera Resolution

Figure 5-1: The Constructed Network Topology (a) Using 2 MegaPixels Camera Resolution (b) Using 0.3 MegaPixels Camera Resolution

The average error of the estimation using estimated coordinates for each node in the network is calculated and plotted as shown in Figure 5-2. The average errors are obtained to be equal to 0.78 cm for 2.0 MegaPixels camera network and 3.23 cm for 0.3 MegaPixels camera network respectively.



(a) Using 2 MegaPixels Camera Resolution

(b) Using 0.3 MegaPixels Camera Resolution

Figure 5-2: The Error of Estimating the Coordinates for Each Node (a) Using 2 MegaPixels Camera Resolution (b) Using 0.3 MegaPixels Camera Resolution

Figure 5-2 shows the absolute error in estimating the camera nodes' coordinates for both networks in 1-tier only. This figure obtains the upper-bounds for the estimation errors to be equal to 1.36 cm and 4.79 cm for 2 MegaPixels and 0.3 MegaPixels camera networks of size 200 × 200 cm^2 respectively. This underscores the good performance of the proposed technique. Furthermore, the accuracy of the estimation increases by increasing the resolution of the camera nodes. As a suggestion, the camera nodes should not only provide standard video output (0.3 MegaPixels), but also they are recommended to be able to snapshot images with higher resolutions, as the trend is with most cameras. This suggestion will increase the accuracy of the proposed technique besides decreasing the costs (memory and processing costs) of the camera node itself.

5.2 The Accumulated Estimation Error In Multi-tier Network

As explained earlier in chapter 3, the estimated coordinates of the *reference nodes* are used to help the nodes of tier-1 to estimate their own coordinates. Similarly, the nodes of tier-2 use the estimated coordinates of the tier-1 nodes to localize themselves. This causes an accumulation of errors in tier-2 estimation. In order to study this, a large enough camera network is deployed consisting of 82 simulated camera nodes as shown in Figure 5-3. Not all the camera nodes are real ones due to the lack in the number of the real sensor nodes in the lab. Instead, a dummy objects as refereed to Figure 4-3 are used to simulate the other nodes and interchangeably with the real camera nodes.

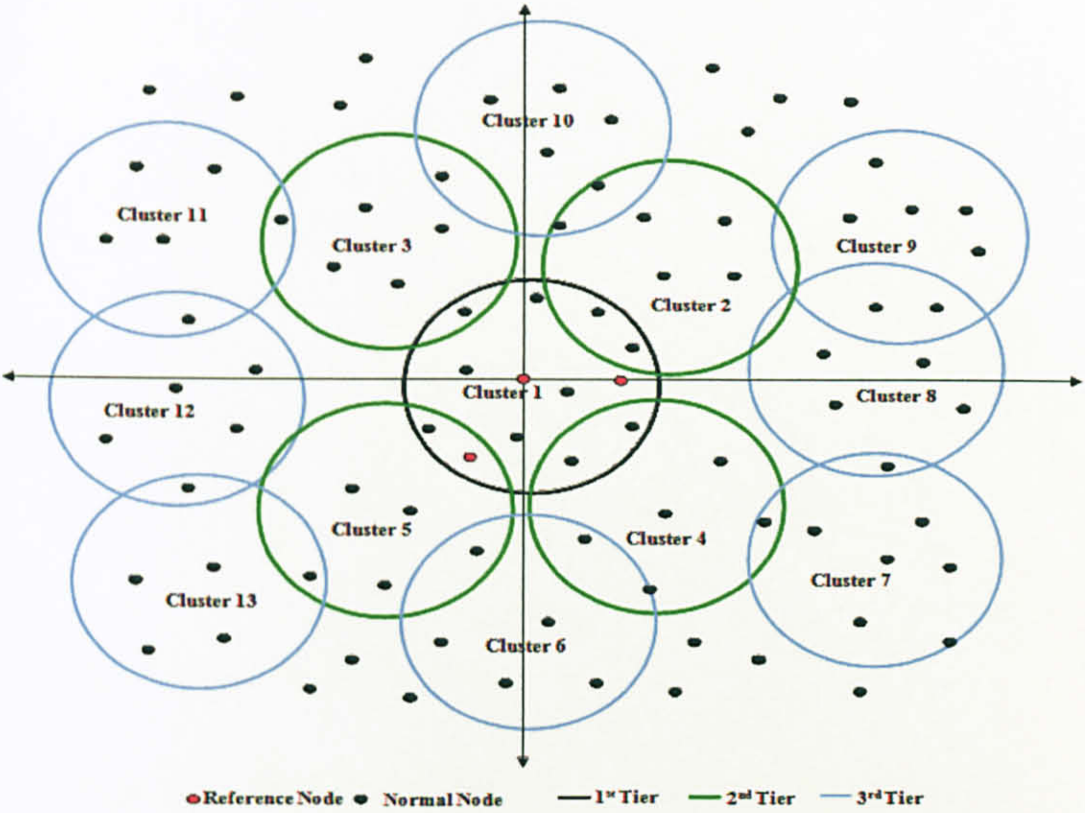


Figure 5-3: Tiers and Cluster Distribution in the Network Deployment

After the camera nodes in the first tier localize themselves relative to the *reference nodes*, some of them are selected to be *reference nodes* for the camera nodes in the second tier. Each tier in the network is divided into many clusters to decrease the time-cost of configuring the whole network. Figure 5-3 shows that all the camera nodes in 1-tier can be localized at the same time. The accumulated error is calculated for each cluster when the estimated coordinates of tier-1 is used to localize the rest of the nodes in the network. A measurement for the absolute error relative to the cluster cross section (200 cm for cluster of size 200 × 200 cm^2) has been done and the average percentage error for each cluster is obtained and plotted in Figure 5-4.

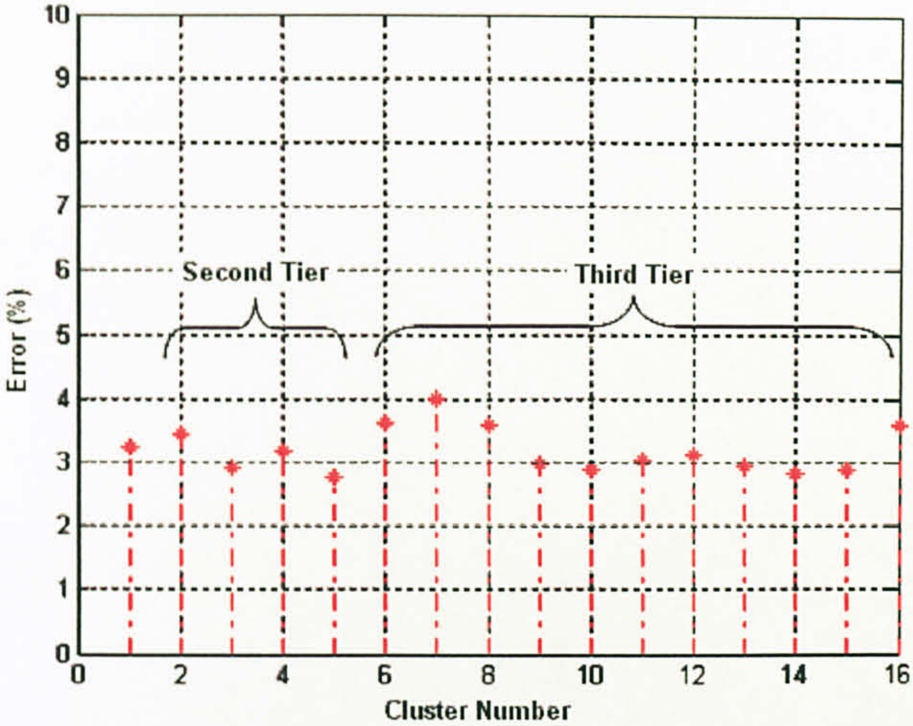


Figure 5-4: The Percentage Average Error for Each Cluster in the Entier Network

From figure 5-4, an average percentage error of 3.18 % for the entire network has been obtained which shows that the accumulated error in estimating the coordinates, even in second or thirds tier, does not grow. In other words, the estimation error does not accumulate in reusing the estimated coordinates to estimate new coordinates in subsequent tiers. This proves the robustness of the multi-tier approach of the proposed vision technique in large networks deployment. Another advantage of the multi-tier approach is that, as the effective area increases from tier 1 to tier 2 and so on, the relative error of estimating the nodes positions (relative to one side of the square area) will decrease due to the fact that the average percentage error remains fixed but the effective area increases. However, this happens at the cost of more time taken to estimate the whole network because the estimation process is carried out in stages – from tier 1 to tier 2 and so on.

5.3 First Application Scenario: Analysis of Stationary Camera Tracking Network

This section provides analytical study of the performance of a stationary camera network in a typical surveillance application. A 1-tier camera network has been deployed consisting of 10 camera nodes with camera resolution of 0.3 MegaPixels. Each camera node in the network is localized and calibrated using the proposed vision based technique. The results of the localization process are obtained as shown in Table 5-3 and Figure 5-5.

Table 5-3: The Coordinates' Results Obtained While Configuring the Camera Nodes

Node No.	Real Coordinates	Est. Coordinates	Real Orientation	Est. Orientation
1	(0.0 , 0.0)	(0.0 , 0.0)	160	160.2
2	(80 , 0.0)	(82.97 , 0.0)	125	125.1
3	(-55 , -80)	(-56.92 , -81.98)	250	250.2
4	(140 , 90)	(137.88 , 91.74)	160	159.3
5	(60 , 160)	(59.11 , 163.48)	45	45
6	(-80 , 140)	(-82.87 , 141.92)	120	119.7
7	(-150 , 50)	(-153.24 , 50.97)	75	75.6
8	(-120 , -60)	(-119.21 , -58.23)	215	215.1
9	(165 , -70)	(167.11 , -71.77)	300	300.6
10	(55 , -145)	(55.89 , -146.63)	240	239.4

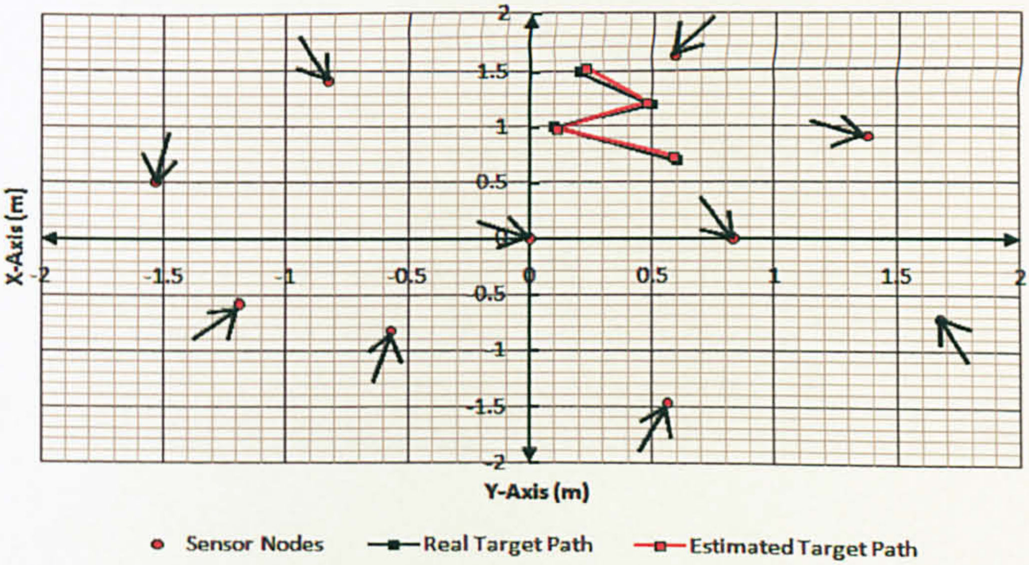


Figure 5-5: The Estimated Network Topology With the Object's Path Through The Sensing Area

All these nodes cooperate and, together, monitor an area of $300 \times 300\ \text{cm}^2$. An arbitrary object is allowed to pass through the sensing area where a typical case is depicted in Figure 5-5. The object is detected by 2 camera nodes - those localized by the proposed technique and described by $C_2[(82.97,0.0), 125.1^\circ]$, and $C_4[(137.88,91.74), 159.3^\circ]$. The object coordinates is estimated in Table 5-4 and an average error is obtained to be equal to 3.37 cm at 4 locations using 2 camera nodes.

Table 5-4: The Coordinates' Results Obtained While Tracking the Object Using 2 Camera Nodes

Observation No.	Object Real Coordinates	Object Estimated Coordinates
1	(20 , 150)	(22.78 , 153.12)
2	(50 , 120)	(48.11 , 121.99)
3	(10 , 100)	(12.40 , 98.12)
4	(60 , 70)	(58.98 , 73.36)

Furthermore, the accuracy of estimating the target can be improved by allowing more camera nodes to participate in the estimation process. Towards that end, the number of the camera nodes is increased. Figure 5-6 shows that this increment, in the number of the nodes participating in locating the target, increases the accuracy of the estimation.

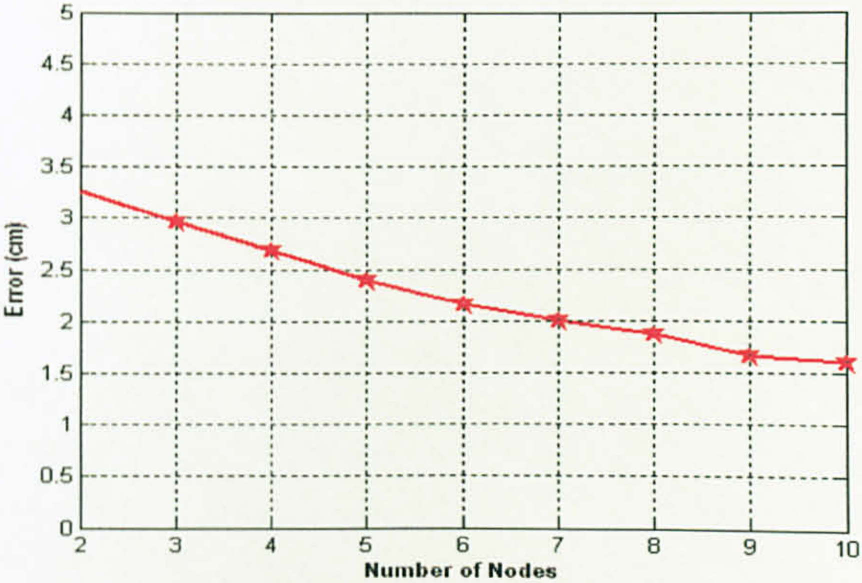


Figure 5-6: The Error of Estimating the Object Coordinate Regarding the Number of Nodes Those Participating in the Estimation Process

Figure 5-6 shows that the average error of localizing an arbitrary object passing through the sensing area decreases by increasing the number of the camera nodes localizing the moving object. Moreover, it shows that there is lower-bound of the estimation error – a value that cannot be improved any more even if the number of the camera nodes is increased. This boundary depends on the camera resolution.

5.4 Performance Comparison with other Localization Techniques

In this section, the proposed technique is compared with some recent localization techniques in the sensor networks. In sensor networks, some techniques rely on the measurement of the power strength or the time of arrival (TOA), or time-difference of arrival (TDOA) and the angle of arrival (AOA) of the wireless signals. These techniques are used for comparison purposes since both categories attempt to localize the sensor nodes in relative coordinates systems. In the next two subsections, a comparison with other vision based techniques and non-vision (wireless) techniques are presented. The relevant practical results, presented in other researches, are captured and made use of those to complete this comparison in both vision and wireless signals localization techniques.

5.4.1 Comparison with Other Vision Based Technique

In this subsection, our results are compared with that of *Huang Lee* [23]. The research in [23] proposed a vision based collaborative node localization technique in surveillance networks using opportunistic target observations. A method that defines and determines the positions and orientations of the nodes is implemented in a relative coordinate system in a decentralized and cluster-based way. The technique assumes that both the reference nodes and the sensor nodes requiring localization can observe a moving target simultaneously. These positions and orientations obtained in the relative coordinate system relative to a two reference nodes with known coordinates. For comparison purposes, the experimental results of a network deployment consisting of 5 sensor nodes including the reference nodes in an area of 12 × 12 foot² which is equal to 3.66 × 3.66 m² are captured as shown in Figure 5-7 and Table 5-5.

Table 5-5: Experimental Results of Distributed Scheme [24]

Target Coordinates			
	True	Linear	Nonlinear
Obs. 1	(1 , 5)	(1.06 , 5.18)	(1.06 , 5.11)
Obs. 2	(0 , 5.5)	*(0.02 , 5.56)*	(0.01 , 5.63)
Obs. 3	(-1 , 7)	(-1.02 , 7.22)	*(-1.01 , 7.17)*
Obs. 4	(0.5 , 8)	(0.48 , 8.14)	(0.48 , 8.15)
Obs. 5	(2 , 6.5)	(1.96 , 6.57)	(1.97 , 6.60)
Sensor Coordinates and Orientations			
Sensor 1	(-5 , 10)	(-5.75 , 10.97)	(-5.83 , 10.97)
	-40°	-39.27°	-39.93°
Sensor 2	(7 , 6.5)	(7.07 , 6.36)	(7.04 , 6.44)
	180°	179.94°	180.644°
Sensor 3	(6 , 2)	(6.023 , 5.31)	(6.023 , 5.31)
	135°	135.79°	135.79°

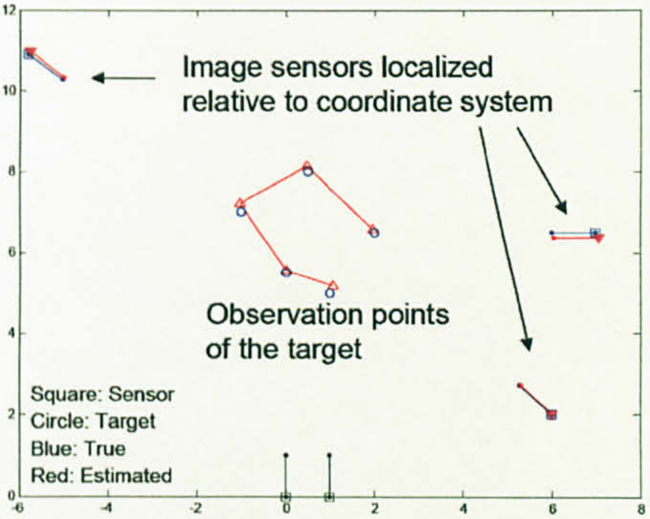


Figure 5-7: Experimental Result of the Linear Method in the Decentralized Scheme [24]

From the results in Table 5-5 the error of estimating the sensor nodes' coordinates is calculated as shown in Table 5-6.

Table 5-6: The Error of Estimating the Sensor Nodes Coordinates and the Orientations

Sensor Number	Coordinates Estimation Error (cm)	Orientation Estimation Error (°)
Sensor 1	37.373	0.73
Sensor 2	4.758	0.06
Sensor 3	100.89	0.79

By comparing the results obtained from the proposed vision technique in a network deployment of 10 nodes in about the same area and localizing the nodes first and then a moving target, the average errors in estimating the camera node coordinates using the proposed technique and the other technique are found to be equal to 3.23 cm and 47.67 cm respectively and the average error in estimating the coordinates of the moving target is equal to 3.37 cm and 4.242 cm respectively. In addition, the average error in estimating the orientation angles is calculated in both technique 0.28° using the proposed technique and 0.53° using the other technique.

These results clearly show that the proposed vision based technique is more accurate in localizing the camera nodes and a moving target in the relative coordinate system as compared to the other vision technique. For comparison purposes, the similarities and the distinctions in both techniques are emphasized. Essentially, both techniques attempt to localize a camera sensor network in relative coordinate system based on vision characteristics. In the other technique, two reference nodes with known separation length between them are used to define the origin-line of the coordinate system and in the proposed technique one reference node that assumes itself at the origin-point of the coordinate system is used to estimate the coordinates of some other two reference nodes assumed to be randomly deployed. However, to achieve localization, the camera nodes in the proposed technique should be able to change their orientation angle during the localization process by rotating the camera – something that is not necessary in the other technique, but simultaneous observations of the moving target is necessary from more than one node to be able to localize. This can be considered as an advantage since all the nodes can be localized in short amount of time, but as a disadvantage if all the nodes have to see the moving target simultaneously to begin the localization process – something that is impractical in a typical random deployment. In the proposed technique, the localization and calibration processes are distributed into three stages where, in the first stage, the 3 reference nodes are localized after one complete searching-cycle from the master node which depends on the movable hardware provided in the camera node. In the second stage, all the nodes in one tier are localized at the same time which, in the minimum case, can be equal to one complete searching-cycle time from the camera node and, in the maximum case, as a three complete cycles' time. This makes the time cost of the proposed technique varying between two values as follow:

$$time\ cost \in [(n \times \tau) + \tau \quad (n \times 3\tau) + \tau] \qquad 5.1$$

where, n is the number of tiers, τ is the time of one searching cycle.

Furthermore, for statistical study, the experiment results of the localization are obtained and carried out 15 times for a 4 node-network including the reference nodes. These results have been compared with the statistical results of the other technique in a network consisting of 3 nodes as shown in Figure 5-8, Figure 5-9, Table 5-7 and Table 5-8.

Table 5-7: Statistical Results for 15 Runs Using the Proposed Technique

	True	Average	Variance
Sensor 1	(0.0 , 0.0)	(0.0 , 0.0)	0.0
Orientation	45°	45.33	1.2°
Sensor 2	(2.25 , 0.0)	(2.26 , 0.0)	0.03
Orientation	135°	134.94°	1.56°
Sensor 3	(-2.25 , 2.25)	(-2.23 , 2.31)	0.03
Orientation	315°	3.14.88°	1.82°
Sensor 4	(2.0 , 3.0)	(2.01 , 3.02)	0.05
Orientation	0.0°	0.9°	0.32°

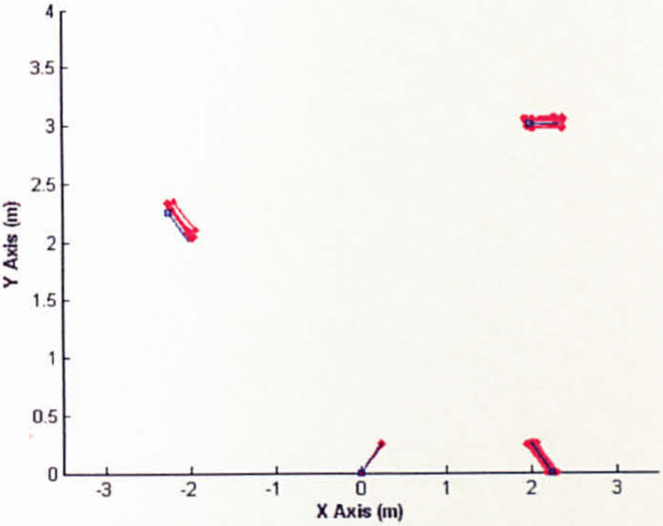


Figure 5-8: Experimental Results for 15 Runs Using the Proposed Technique

Table 5-8: Statistical Results for 15 Runs Using Other Vison Technique [24]

	True	Average	Variance
Coordinates	(-5 , 21.5)	(-4.49 , 19.43)	0.6310
Orientation	315°	309.63°	1.5814°
Obs. 1	(-1 , 14.5)	(-1.02 , 14.16)	0.5527
Obs. 2	(-1 , 12.5)	(-1.12 , 12.50)	0.2587
Obs. 3	(1 , 11.5)	(0.83 , 10.67)	0.8568
Obs. 4	(2 , 13.5)	(1.79 , 12.22)	1.2966
Obs. 5	(3 , 10)	(2.70 , 9.07)	0.9864

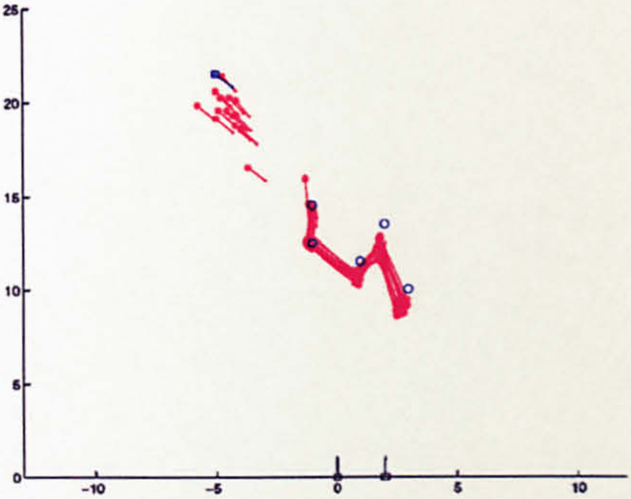


Figure 5-9: Experimental Result for 15 Runs Using Other Vision Technique [24]

These statistical studies show that the average errors in estimating the coordinates of the camera nodes and their angles of orientation using the proposed technique and the other technique are equal to 1.4 cm, 0.9°, and 64.9 cm and 5.37° respectively. The variances are obtained from both results to be around 5 cm with 0.32° using the proposed technique and 19.23 cm with 1.58° using the other technique.

5.4.2 Comparison with other Wireless Signal Based Technique

In this subsection, the vision based localization and calibration technique is compared with some wireless signal based localization technique popular in the WSNs. As such, the two techniques are very different from each other in the way they estimate the coordinates of the sensor nodes, but they have been proposed for the same purpose and intent. Basically, these techniques perform the localization process based on the estimation of the received signal-power as in [28], or based on the wave signal characteristics (e.g. the propagation speed of the waves to perform distances estimation using the TDOA and TOA methods) as in [14] [29]. These techniques can handle the slow moving nodes in the Mobile Ad-Hoc Wireless Networks (MANET) with more or less same difficulties as the vision based localization techniques even though the vision based techniques were essentially proposed for static VWSN.

The idea behind this comparison, without loss of generality, is to define the estimation accuracy of both types of localization techniques.

5.4.2.1 Comparison with Signal Strength Technique

First, the results obtained from the proposed technique are compared with those obtained from a scheme based on estimating the power received from two reference nodes placed in known coordinates proposed by *P. Bergamo* [28]. *Bergamo et al* construct a simulation of 20 nodes network deployed in rectangular area of dimensions

100 × 100 m². The nodes are assumed to be stationary and the position errors are averaged on many different locations for each node as shown in Figure 5-10.

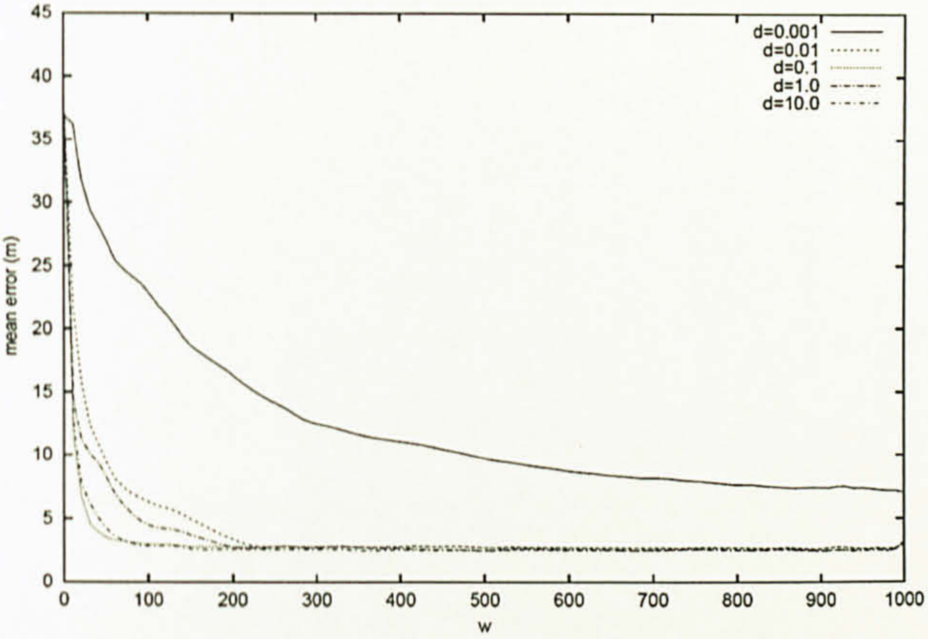


Figure 5-10: Mean Error as a Function of Window Size (No Mobility) [28]

Figure 5-1 shows that the best performance in estimating the nodes positions can be reach by using small window size – window comprising of only those power signals that have already been estimated. The fading samples are assumed un-correlated. This wireless based technique notes that the mean error never reaches the value of zero and, in particular, the lower bound of the mean error is equal to 2.5 m which is equal to 2.5% error relative to the area dimensions. This lower bound can be obtained with different values of window sizes depending on the fading of the signal in the surrounding environment. As shown in Figure 5-10, the size of the window *w* increases when the normalized maximum Doppler frequency *d* decreases to reach the acceptable position error. For statistical study of the wireless based technique, the variance in the estimation process is calculated as shown in Figure 5-11.

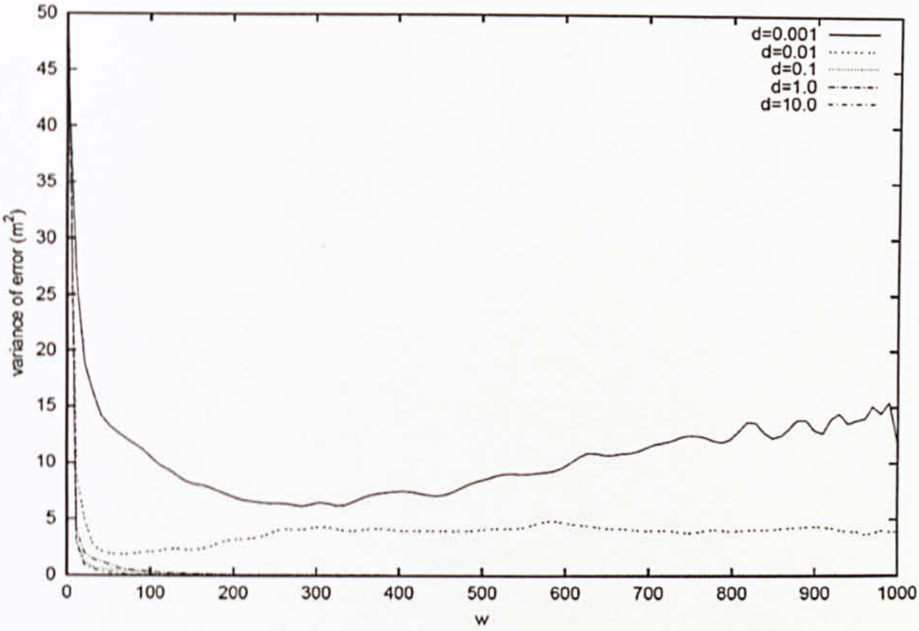
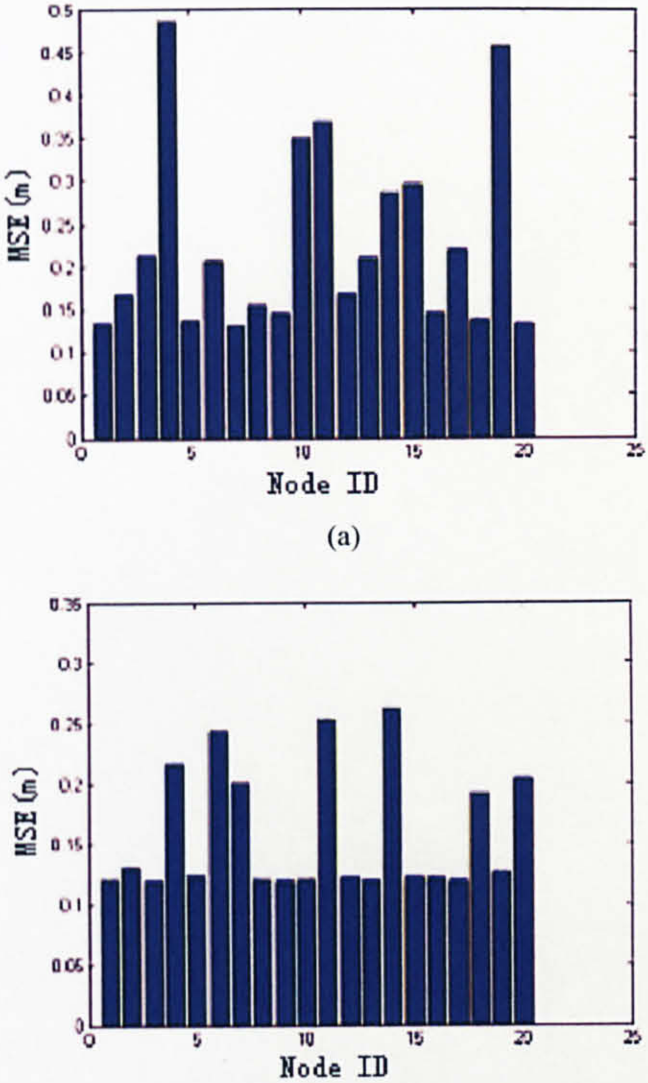


Figure 5-11: Variance of Erorr as a Function of Window Size (No Mobility) [28]

Figure 5-11 shows that the variance of the estimation error is near to zero which implies that this technique is stable when the fading samples are not so correlated. The variance of the error will increase when the value of *d* decreases.

5.4.2.2 Comparison with TDOA, TOA and AOA Technique

Another localization technique for sensor nodes and a target is proposed by *Xing-Yu* [14] [29]. It determines the coordinates of the nodes by placing reference sources at known coordinates outside the region of observation. It relies on calculating the TDOA, TOA and AOA of the signals to localize the sensor nodes and the target. In this technique a simulation of 20-nodes network is constructed to show the performance of in an area of 100 × 100 m^2. Four nodes with known positions are used to localize the rest of the network. For validation purposes, the mean squared errors (MSE) are calculated and plotted in Figure 5-12-a and Figure 5-12-b for versus nodes for using the AOA method.



(a)

(b)

Figure 5-12: MSE of Each Node (a) With Clock Bias (b) With Absolute Time Reference [14]

From Figure 5-12 the MSE is obtained and averaged over all the nodes to be equal to 0.23 m with clock bias and 0.16 m without any bias, also described as absolute time reference. Similarly, the maximum MSE is 0.49 m with clock bias and 0.26 m without bias. These results are used to obtain the average percentage errors to be 2.31% using the TDOA method and 0.53% using the AOA method ignoring the affect of the signal disturbance and fading. However, the proposed vision technique in this thesis gives about 3.18% average error for given resolution. The wireless based technique, however, requires an accurate synchronization method in a typical deployment.

The disadvantage of the wireless signal based localization technique is that it requires a good model of the noise disturbance and the fading arising from multi-paths and Doppler, a large window for averaging the estimates, an accurate synchronization method among the nodes and, also, sometimes, extra hardware like array of antennas etc.. On the other hand, these techniques are good at localizing large number of nodes in the sensing area in a very short time decreasing the time cost of localizing the nodes in large network.

This study notes that the only similarity between the vision based and the wireless signal based localization techniques is that both of them attempt to localize the sensor nodes. This clarifies the reason why the comparison of the proposed vision technique with the wireless based techniques is handled only in terms of accuracy measured and the absolute percentage error of the estimation process.

5.5 Conclusion

In this chapter, the vision based localization and calibration technique which has been proposed to localize one-plane camera network is analyzed. The camera tracking network which has been calibrated using the proposed technique is also analyzed and the performance of such network for surveillance and tracking application has been investigated. Moreover, the proposed technique is compared with some other vision based and wireless signal based localization techniques to evaluate the performance of the proposed technique. Some suggestions to enhance the accuracy of localizing any other object passing through the sensing area have been made.

The next chapter concludes the study and emphasize the importance of this work as a vision based localization and calibration technique in the video sensor networks with some suggestions for future works in the same field.

CHAPTER SIX: CONCLUSIONS AND FUTURE WORKS

This chapter concludes the presentation of this research works and summarizes some future works that may enhance the performance of the proposed vision based localization and calibration technique. Also, some suggestions are provided that would add extra reliability to the surveillance application.

6.1 Conclusions

As mentioned before, the area of video sensor network and its application has become very attractive for researchers. The applications of VWSN span both the civilian and military fields. This is facilitated by the rich visual information provided by the camera nodes allowing situation awareness, event understanding, and decision making. This property makes the use of such networks in surveillance, monitoring, tracking and events detecting and reporting applications very efficient.

In this thesis, a localization technique is presented for video sensor networks based on a simple vision concept, the relationship between the image heights in the image plane and the real distances between the camera nodes. This technique localizes and calibrates *all* the camera nodes randomly deployed in one-plane. The proposed vision technique attempts to localize the entire network with the help of *only* three reference nodes, also deployed randomly like the other nodes. Most other techniques require that these reference nodes are equipped with special hardware like GPS receivers to globally locate them. In comparison, the proposed technique requires only one reference node with GPS and manages to globally locate all the nodes. The technique consists of three main steps namely, localizing the core of the network, localizing the remaining nodes, and calibrating all the nodes relative to the coordinate system of the reference nodes, and, thereby, determining the global coordinates. The ability of the proposed vision based technique has been demonstrated in localizing the sensor nodes with average percentage error not greater than 3.18% - a performance better than the other vision based technique.

Furthermore, a surveillance application of a camera network has been presented that is assumed to be localized and calibrated using the proposed vision technique. The results of localizing and tracking an arbitrary object moving around in the sensing field showcase the robustness of the proposed technique against the accumulated errors of the localization estimation for both the camera nodes and also the target object.

6.2 Extensions and Future Works

This section provides some ideas to enhance the performance of the proposed vision based localization and calibration technique.

As extension to this work, the performance of the localization technique can be enhance by implementing an adaptive searching algorithm so as to decrease the time-cost of the localization process. This can be done by making a full scan in the FOV of the camera to decide the next step of scanning the FOV. As mentioned in chapter 3, the main idea in this technique is to estimate the distances between the nodes using the relationship between the real distance and the height of the images in the image-plane. For enhancement purposes, two snap-shots, one with zero zoom and the other with different zoom, are suggested to make use of averaging technique to find the best estimate.

For the surveillance application, object recognition and classification features are suggested to be added in the application for more reliability.

In conclusion, the work in this thesis shows that the vision based localization and calibration technique provide visual information suitable for surveillance and tracking purposes which open new areas for researchers to propose techniques that provide good accuracy beside handling the localization problem in VWSNs. In summary, this thesis introduces a novel vision based localization and calibration technique for surveillance camera network.

List of Publications

(Available on IEEE Xplore)

[1] Sharif A. M. Sharif, Varun Jeoti, "Video Wireless Sensor Network: Co-Operative Vision Based Localization Method", *In Proceedings of the Second Asia International Conference on Modelling and Simulations (AMS'08),* pp. 570-573, Malaysia, May, 2008.

[2] Sharif A. M. Sharif, Varun Jeoti, "Configuring Video Sensor Network Using a Novel Vision Based Technique", *In Proceedings of the International Symposium on Information Technology 2008 (ITSIM'08),* vol. 4, pp. 2460-2464, Malaysia, August, 2008.

BIBLIOGRAPHY

[1] Rohini Krishnapura, "Making Sense of Sensor Networks", *In the ACM Student Magazine, Crossroads 9.4: Computer Networking,* Issue 9.4, 2003.

[2] Zhu Han, K. J. Ray Liu., "Resource Allocation for Wireless Networks: Basics, Techniques, and Applications", *Cambridge University Press,* 1st Edition, April 2008.

[3] Mainwaring A., Culler, D., Polastre, J., Szewczyk, R., and Anderson, J., "Wireless Sensor Networks for Habitat Monitoring", *In Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA'02)*, pp. 88-97, Georgia, Septemper 2002.

[4] Stankovic, J. A., Cao, Q., Doan, T., Fang, L., He, Z., Kiran, R., Lin, S., Son, S., Stoleru, R., Wood, A., "Wireless Sensor Networks for In-Home Healthcare: Potential and Challenges", *In Proceedings of Workshop on High Confidence Medical Devices Software and Systems (HCMDSS)*, 2005.

[5] Jr., Edgar H. Callaway, "Wireless Sensor Networks: Architectures and Protocol", *Auerbach Publications CRC Press*, 1st Edition, August 2003.

[6] Tatiana Bokareva, Wen Hu, Salil Kanhere, Branko Ristic, Neil Gordon, Travis Bessell, Mark Rutten, and Sanjay Jha, "Wireless sensor networks for battlefield surveillance", *In Land Warfare Conference,* October 2006.

[7] Sha, Kewei, Shi, Weisong and Watkins, "Using Wireless Sensor Networks for Fire Rescue Applications: Requirements and Challenges", *In Proceedings of the IEEE on Electro/Information Tecknology EIT,* pp. 239-244, May 2006.

[8] Terwilliger, Mark, "LOCALIZATION IN WRIELESS SENSOR NETWORKS", *PhD Thesis from Westren Michigan University*, April 2006.

[9] Rudafshani, Masoomeh, "Localization in Wireless Ad Hoc Sensor Networks", *MSc Thesis from York University*, Toronto, June 2006.

[10] J. Bachrach, and C. Taylor, "HandBook of Sensor Networks", *In Localization Chapter, Wiley Press*, 2005.

[11] Mohammad Ilyas, Imad, "Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems", *In CRC Press*, 2005.

[12] Doherty, L., Pister, K. S. J., and El Ghaoui, L., "Convex Position Estimation in Wireless Sensor Networks", *In Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communication societies INFOCOM*, vol. 3, pp. 1655-1663, USA, April 2001.

[13] Shang, Y., Ruml, W., Zhang, Y., and Fromherz, M. P. J., "Localization from Mere Connectivity", *In Proceedings of the 4th ACM International Symposium on Mobile ad hoc Networking & Computing MobiHoc*, pp. 201-212, USA, June 2003.

[14] Xing-Yu Pi, Hong-Yi Yu, and Yan-Jun Zhang, "An Assisting Localization Method for Wireless Sensor Networks", *In Proceedings of Mobile Technology, Applications and Systems, the 2nd International Conference*, China, November 2005.

[15] Nissanka B. P., Anit Chakraborty, and Hari B., "The Cricket Location-Support System", *In Proceedings of the 6th ACM International Conference on Mobile Computing and Networking MOBICOM*, pp. 32-43, USA, 2000.

[16] Nasipuri and K. Li, "A Directionality based Location Discovery Scheme for Wireless Sensor Networks", *In First ACM International Workshop on Wireless Sensor Networks and Applications ACM WSNA'02,* Atlanta, GA, September 2002.

[17] Alippi, C., and Vanini, G., "A RSSI-based and calibrated centralized localization technique for Wireless Sensor Networks", *In Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)*, pp. 301-305, Milano, 2006.

[18] Bulusu N., Heidemann J., and Estrin D., "GPS-less Low Cost Outdoor Localization For Very Small Devices", *In IEEE Personal Communications Magazine*, vol. 7, pp. 28-34, October 2000.

[19] Niculescu D., and Nath B., "Ad Hoc Positioning System (APS)", *In Proceedings of IEEE Global Telecommunications Conference GLOBECOM '01*, pp. 2926-2931, Texas, 2001.

[20] Nagpal, R., Shrobe, H., and Bachrach, J., "Organizing a Global Coordinate System from Local Information on an Ad Hoc Sensor Network", *In Proceeding of the 2nd international Workshop on Information Processing in Sensor Networks IPSN 2003*, pp. 333-348, California, 2003.

[21] Kleinrock, L., and Silvester, J., "Optimum Transmission RADII for Packet Radio Networks or Why Six is A Magic Number", *In Proceedings of National Telecomm Conference*, vol. 1, pp. 431-435, California, 1978.

[22] He, T., Huang, C., Brian, M. B., John, A. S., Abdelzaher, T., "Range-Free Localization Schemes for Large Scale Sensor Networks", *In Proceedings of the 9th Annual International Conference on Mobile Computing and Networking MobiCom '03*, pp. 81-95, Califorina, September 2003.

[23] Liu, X., Kulakarni, P., Shenoy, P. and Ganesan, D., "Snapshot: A Self-Calibration Protocol for Camera Sensor Networks", *In Proceedings of the 3rd IEEE International Conference on Broadband Advanced Sensor Networks (BASENETS 2006)*, pp. 1-10, San Jose, 2006.

[24] Lee, H., Aghajan, H., "Collaborative Node Localization in Surveillance Networks using Opportunistic Target Observations", *In Proceedings of the 4th ACM International Workshop on Video Surveillance & Sensor Networks"*, pp. 9-18, Santa Barbara, CA, USA, October 2006.

[25] Lee, H., Dong, H., Aghajan, H., "Robot-Assisted Localization Techniques for Wireless Image Sensor Networks", *In Proceedings of the Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks,* Reston, VA, USA, Septemper, 2006.

[26] GEOFFRION, JR., "Understanding Digital Camera Resolution", The Luminous Landscape webpage, http://www.luminous-landscape.com/tutorials/understanding-series/res-demyst.shtml, last exploration December 21, 2008 03:10 PM.

[27] B&H Photo and Electronics, "Digital Camera Resolution", http://www.bhphotovideo.com/FrameWork/charts/resolutionChartPopup.html, last exploration Dec 21, 2008 03:15 PM.

[28] Mazzini, P. Bergamo and G., "Localization in sensor networks with fading and mobility", *In Proceedings of the 13th IEEE International Symposium, Personal Indoor and Mobile Radio Communication PIMRC,* vol.2, pp. 750-754, Italy, 2002.

[29] Xing-yu Pi, Hong-yi Yu, "A Distributed and Cooperative Target Localization Algorithm in Wireless Sensor Networks", *In Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Application and Technologies (PDCAT'05),* pp. 887-889, 2005.

[30] Wilson, P., and Fernandez, J., "FACIAL FEATURE DETECTION USING HAAR CLASSIFIERS", *In Journal of Computing Sciences in Colleges JCSC,* vol. 21, pp. 127-133, April 2006.

[31] Gary Bradski, Adrian Kaehler, Vadim Pisarevsky, "Learning-Based Computer Vision with Intel's Open Source Computer Vision Library", *In Intel Technology Journal,*
vol 9, pp. 119-130, May 2005.

[32] Paul Viola and Jones Michael, "Rapid Object Detection using a Boosted Cascade of Simple Features", *In Proceedings of the IEEE Computer Vision and Pattern Recognision Conference CVPR,* vol. 1, pp. I-511-I518, USA, 2001.

[33] Intel® Integrated Performance Primitives (Intel® IPP), http://www.intel.com/support/performancetools/libraries/ipp/sb/cs-010656.htm, 23-Aug-2007.

[34] Adolf, Florian, "How-to build a cascade of boosted classifiers based on Haar-like features", *University of Applied Sciences Trier, Department of Design and Computer Science,* May, 2003.

[35] Florian-M. Adolf, "Untitled Download Page", *informatik: Robotik Homepage* http://robotik.inflomatik.info/other/opencv/OpenCV_ObjectDetection_Demo.exe , last exploration December 21, 2008 03:40 PM.

[36] Wikipedia the free encyclopedia, "Wireless Sensor Network", http://en.wikipedia.org/wiki/Wireless_sensor_network, last exploration December 21, 2008 03:50 PM,

[37] Akyildiz, I. F.,Weilian Su Sankarasubramaniam,Y. Cayirci, E., "A Survey on Sensor Networks", *In the IEEE Communication Magazine,* vol. 40, pp. 102-114, Atlanta, August 2002.

[38] H. Lee and H. Aghajan, "Collaborative Self-Localization Techniques for Wireless Image Sensor Networks", In Proceedings of the Asilomar Conference on Signals, Systems and Computers, California, October 2005.

[39] H. Lee, L. Savidge, and H. Aghajan, "Subspace Techniques for Vision-Based Node Localization in Wireless Sensor Networks", *In Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP),* France, May 2006.

[40] Yamaguti, N., OE, S., Terada, K., "A Method of Distance Measurement by Using Monocular Camera", *In Proceedings of the 36th SICE Annual Conference, International Session Papers,* pp. 1255-1260, Tokushima, July 1997.

[41] Doros Agathangelou, Benny P. L. Lo, Jeffrey L. Wang and Guang-Zhong Yang, "Self-Configuring Video Sensor Network", *In Proceedings of the 3rd International Conference on Pervasive Computing PERVASIVE'05*, Munich, May 2005.

APPENDIX A: FLOW CHART OF THE VISION BASED LOCALIZATION TECHNIQUE AND THE PROCESS FOR EACH TYPE OF THE CAMERA NODES

In this appendix, we explain, briefly, the process of localization and calibration as used in the proposed vision based technique.

APPENDIX A.1: FLOW CHART OF THE VISION BASED LOCALIZATION TECHNIQUE

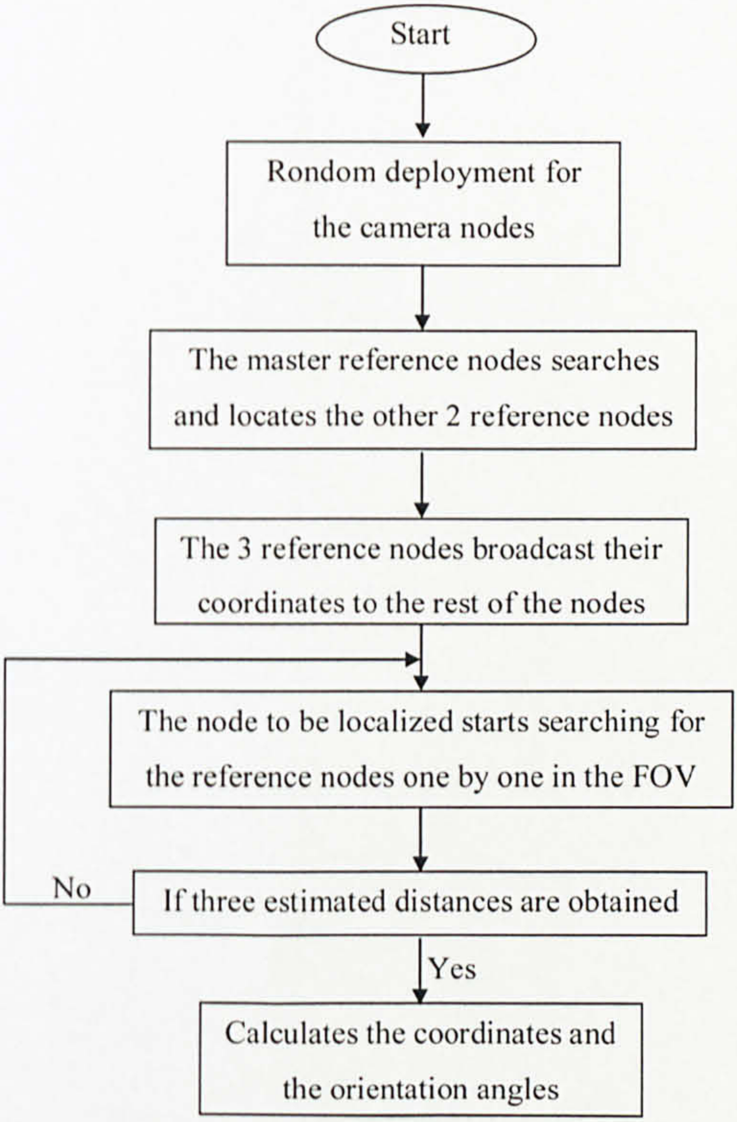

Start
Rondom deployment for the camera nodes
The master reference nodes searches and locates the other 2 reference nodes
The 3 reference nodes broadcast their coordinates to the rest of the nodes
The node to be localized starts searching for the reference nodes one by one in the FOV
If three estimated distances are obtained
No
Yes
Calculates the coordinates and the orientation angles

Figure A-1: Flow Chart of the Vision Based Localization Technique

APPENDIX A.2: FLOW CHART FOR THE MASTER REFERENCE NODE PARTICIPATION

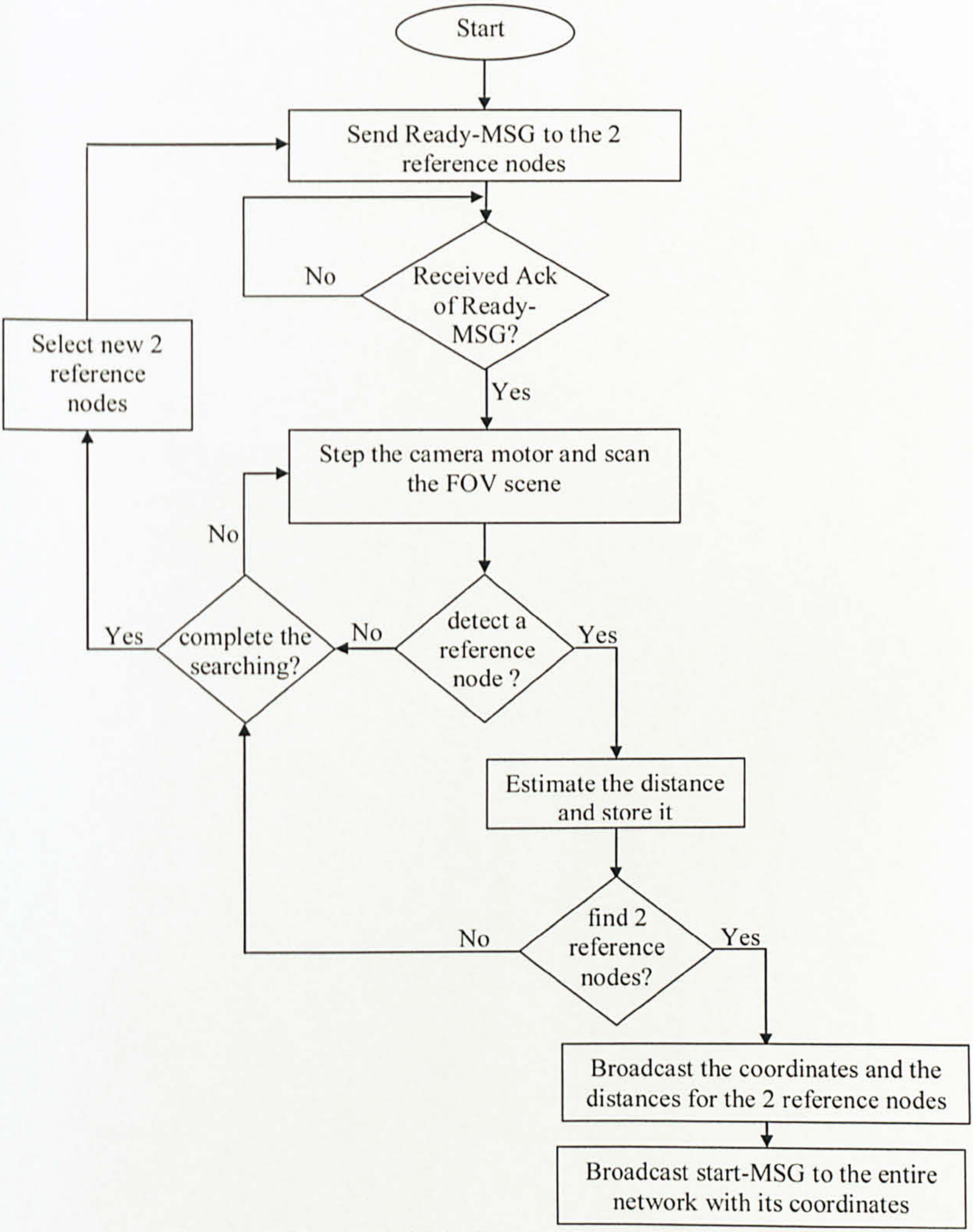

Start
Send Ready-MSG to the 2 reference nodes
No
Received Ack of Ready-MSG?
Yes
Select new 2 reference nodes
Step the camera motor and scan the FOV scene
No
Yes
complete the searching?
No
detect a reference node ?
Yes
Estimate the distance and store it
No
find 2 reference nodes?
Yes
Broadcast the coordinates and the distances for the 2 reference nodes
Broadcast start-MSG to the entire network with its coordinates

Figure A-2: Flow Chart for the Master Reference Node Participation

APPENDIX A.3: FLOW CHART FOR THE TWO REFERENCE NODES PARTICIPATION

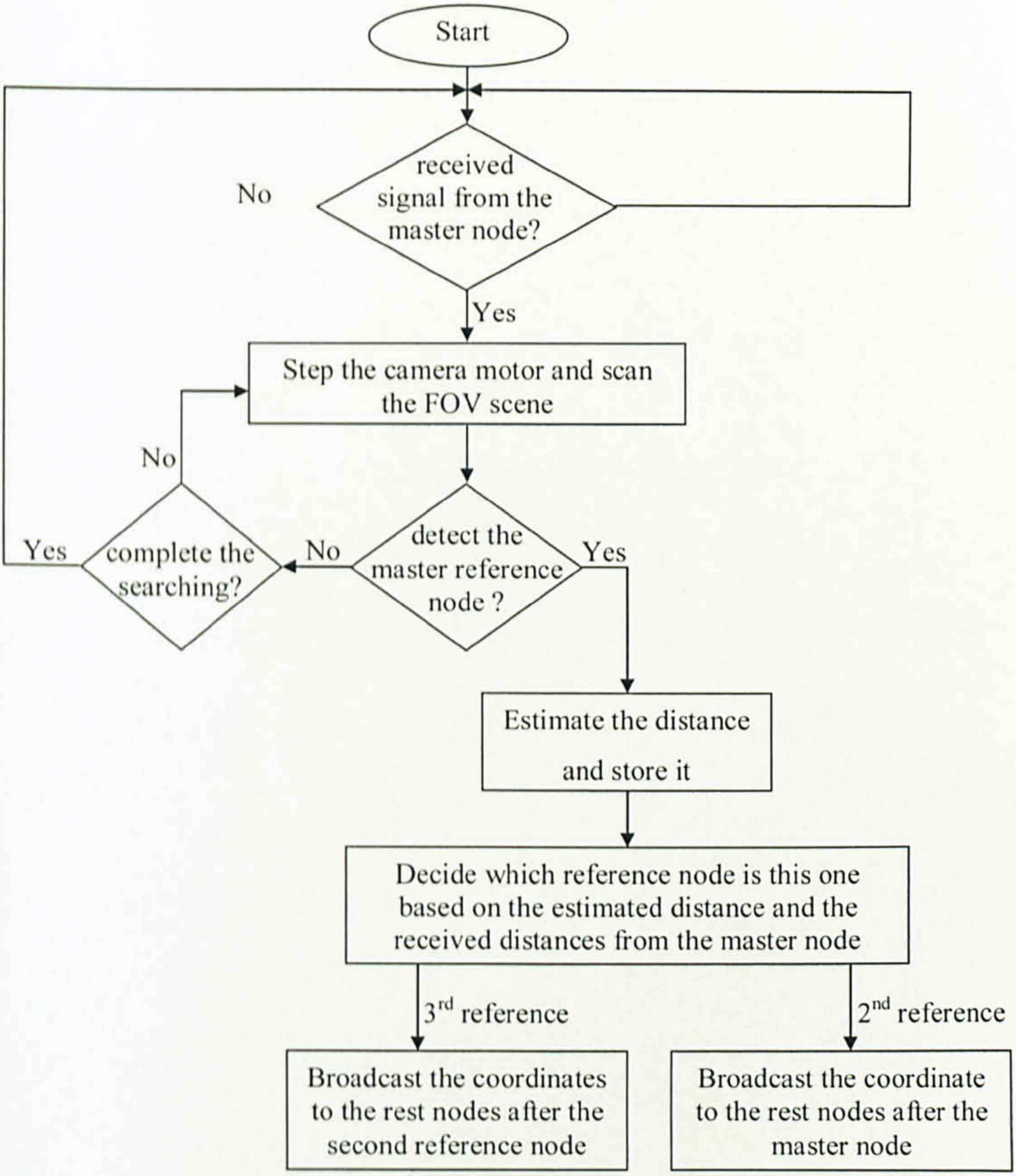

Start
received signal from the master node?
No
Yes
Step the camera motor and scan the FOV scene
detect the master reference node ?
No
Yes
complete the searching?
No
Yes
Estimate the distance and store it
Decide which reference node is this one based on the estimated distance and the received distances from the master node
3rd reference
2nd reference
Broadcast the coordinates to the rest nodes after the second reference node
Broadcast the coordinate to the rest nodes after the master node

Figure A-3: Flow Chart for the Two Reference Nodes Participation

APPENDIX A.4: FLOW CHART FOR THE NORMAL CAMERA NODE PARTICIPATION

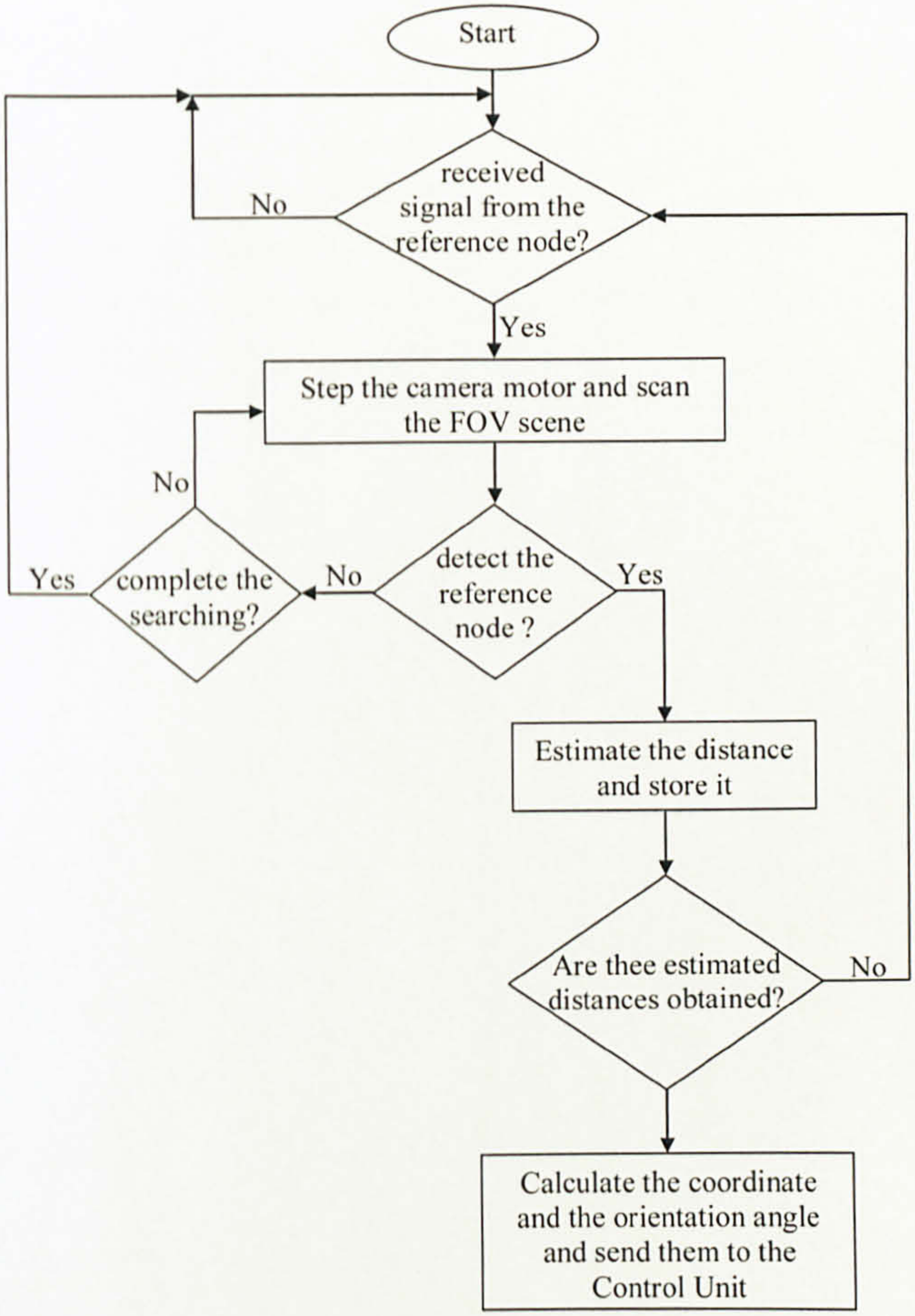

Start
received signal from the reference node?
No
Yes
Step the camera motor and scan the FOV scene
No
detect the reference node ?
No
Yes
complete the searching?
Yes
Estimate the distance and store it
Are thee estimated distances obtained?
No
Calculate the coordinate and the orientation angle and send them to the Control Unit

Figure A-4: Flow Chart for the Normal Camera Node Participation

APPENDIX B: LOCALIZATION CONCEPT AND DERIVATION

In this appendix of the thesis, the main concept behind calculating the coordinates of one sensor node is explained and derived in the relative coordinate system. It has been assumed that 3 *reference nodes* broadcast their estimated coordinates, so the camera nodes can search for them in their FOV and estimate the distances to these reference nodes. As shown in Figure B-1, 3 estimated coordinates (X_i, Y_i) , $i = 1,2,3$ in the relative coordinate system representing the *reference nodes* and 3 estimated distances between the camera node to be localized and the *reference nodes* $d_{4,i}$, $i = 1,2,3$.

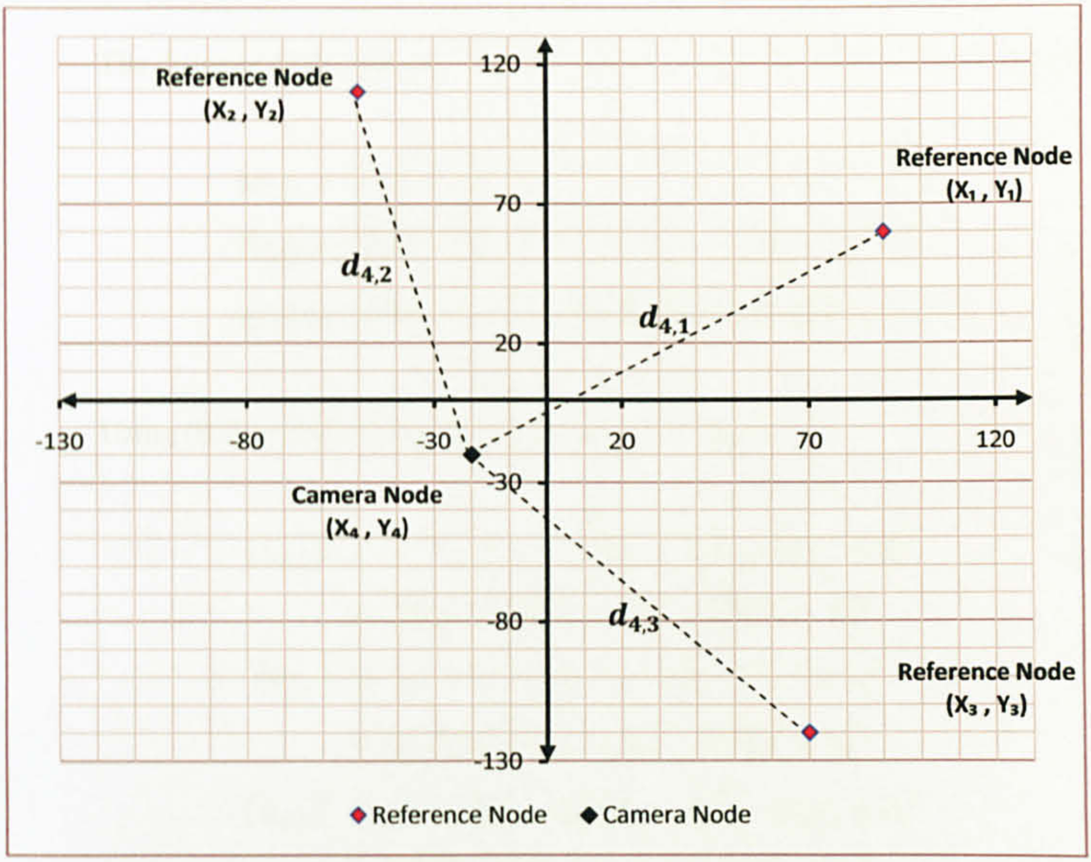


Figure B-1: A Typical Random Deployment of Nodes and Their Representation in the Relative Coordinate System

These distances are:

$$\left.\begin{aligned} d_{4,1} &= \sqrt[2]{(X_1 - X_4)^2 + (Y_1 - Y_4)^2} \\ d_{4,2} &= \sqrt[2]{(X_2 - X_4)^2 + (Y_2 - Y_4)^2} \\ d_{4,3} &= \sqrt[2]{(X_3 - X_4)^2 + (Y_3 - Y_4)^2} \end{aligned}\right\} \text{B.1}$$

$$\left.\begin{aligned} (d_{4,1})^2 &= (X_1 - X_4)^2 + (Y_1 - Y_4)^2 = [X_1{}^2 - 2X_1X_4 + X_4{}^2] + [Y_1{}^2 - 2Y_1Y_4 + Y_4{}^2] \\ (d_{4,2})^2 &= (X_2 - X_4)^2 + (Y_2 - Y_4)^2 = [X_2{}^2 - 2X_2X_4 + X_4{}^2] + [Y_2{}^2 - 2Y_2Y_4 + Y_4{}^2] \\ (d_{4,3})^2 &= (X_3 - X_4)^2 + (Y_3 - Y_4)^2 = [X_3{}^2 - 2X_3X_4 + X_4{}^2] + [Y_3{}^2 - 2Y_3Y_4 + Y_4{}^2] \end{aligned}\right\} \text{B.2}$$

The distance of the reference nodes from the origin can be calculated as follow:

$$\left.\begin{aligned} \|S_1\| &= \sqrt[2]{(X_1 - 0)^2 + (Y_1 - 0)^2} = \sqrt[2]{(X_1)^2 + (Y_1)^2} \\ \|S_2\| &= \sqrt[2]{(X_2 - 0)^2 + (Y_2 - 0)^2} = \sqrt[2]{(X_2)^2 + (Y_2)^2} \\ \|S_3\| &= \sqrt[2]{(X_3 - 0)^2 + (Y_3 - 0)^2} = \sqrt[2]{(X_3)^2 + (Y_3)^2} \end{aligned}\right\} \text{B.3}$$

Using (B.2) and (B.3), the equations are rewritten as:

$$\left.\begin{aligned} (d_{4,1})^2 &= X_1{}^2 + Y_1{}^2 - 2X_1X_4 + X_4{}^2 - 2Y_1Y_4 + Y_4{}^2 \\ &= \|S_1\|^2 - 2X_1X_4 + X_4{}^2 - 2Y_1Y_4 + Y_4{}^2 \\ (d_{4,2})^2 &= X_2{}^2 + Y_2{}^2 - 2X_2X_4 + X_4{}^2 - 2Y_2Y_4 + Y_4{}^2 \\ &= \|S_2\|^2 - 2X_2X_4 + X_4{}^2 - 2Y_2Y_4 + Y_4{}^2 \\ (d_{4,3})^2 &= X_3{}^2 + Y_3{}^2 - 2X_3X_4 + X_4{}^2 - 2Y_3Y_4 + Y_4{}^2 \\ &= \|S_3\|^2 - 2X_3\,X_4 + X_4{}^2 - 2Y_3Y_4 + Y_4{}^2 \end{aligned}\right\} \text{B.4}$$

From these 3 equations in (B.4) the following 2 equations are derived:

$$\left.\begin{aligned}(d_{4,1})^2 - (d_{4,2})^2 &= \|S_1\|^2 - \|S_2\|^2 - 2X_4[X_1 - X_2] - 2Y_4[Y_1 - Y_2]\\ (d_{4,1})^2 - (d_{4,3})^2 &= \|S_1\|^2 - \|S_3\|^2 - 2X_4[X_1 - X_3] - 2Y_4[Y_1 - Y_3]\end{aligned}\right\} \text{B.5}$$

Then, (B.5) is expressed in the form *Ay=b*.

$$\left.\begin{aligned}X_4(X_1 - X_2) + Y_4(Y_1 - Y_2) &= {}^1\!/_2\left[\|S_1\|^2 - \|S_2\|^2 - (d_{4,1})^2 + (d_{4,2})^2\right]\\ X_4(X_1 - X_3) + Y_4(Y_1 - Y_3) &= {}^1\!/_2\left[\|S_1\|^2 - \|S_3\|^2 - (d_{4,1})^2 + (d_{4,3})^2\right]\end{aligned}\right\} \text{B.6}$$

where,

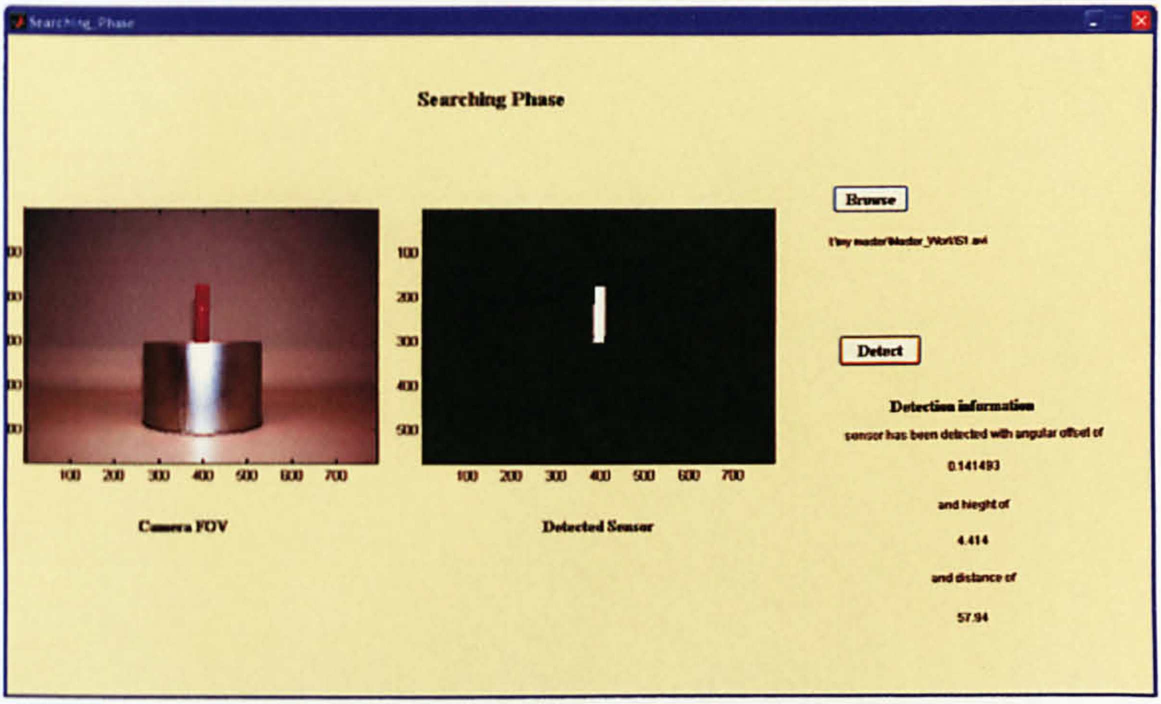
$$A = \begin{bmatrix}(X_1 - X_2) & (Y_1 - Y_2)\\ (X_1 - X_3) & (Y_1 - Y_3)\end{bmatrix}, y = \begin{bmatrix}X_4\\ Y_4\end{bmatrix}$$

$$b = {}^1\!/_2\begin{bmatrix}\|S_1\|^2 - \|S_2\|^2 - d_{4,1}{}^2 + d_{4,2}{}^2\\ \|S_1\|^2 - \|S_3\|^2 - d_{4,1}{}^2 + d_{4,3}{}^2\end{bmatrix}$$

APPENDIX C: MATLAB® IMPLEMENTATION

APPENDIX C.1: GRAPHIC USER INTERFACE OF THE CAMERA NODE SEARCHING PHASE

In this part, the code to present a GUI of the searching phase of the camera node is implemented in MATLAB® and provided. This GUI, as shown in Figure C-1, loads the video by pressing the browse button and selects the video recorded by the camera node in the searching step. The detection information of the sensor in the FOV can be shown in Figure C-1.



[illegible]

Detection information

sensor has been detected with angular offset of

0.141493

and hieght of

4.414

and distance of

57.94

Figure C-1: Snapshot of the Searching Phase GUI

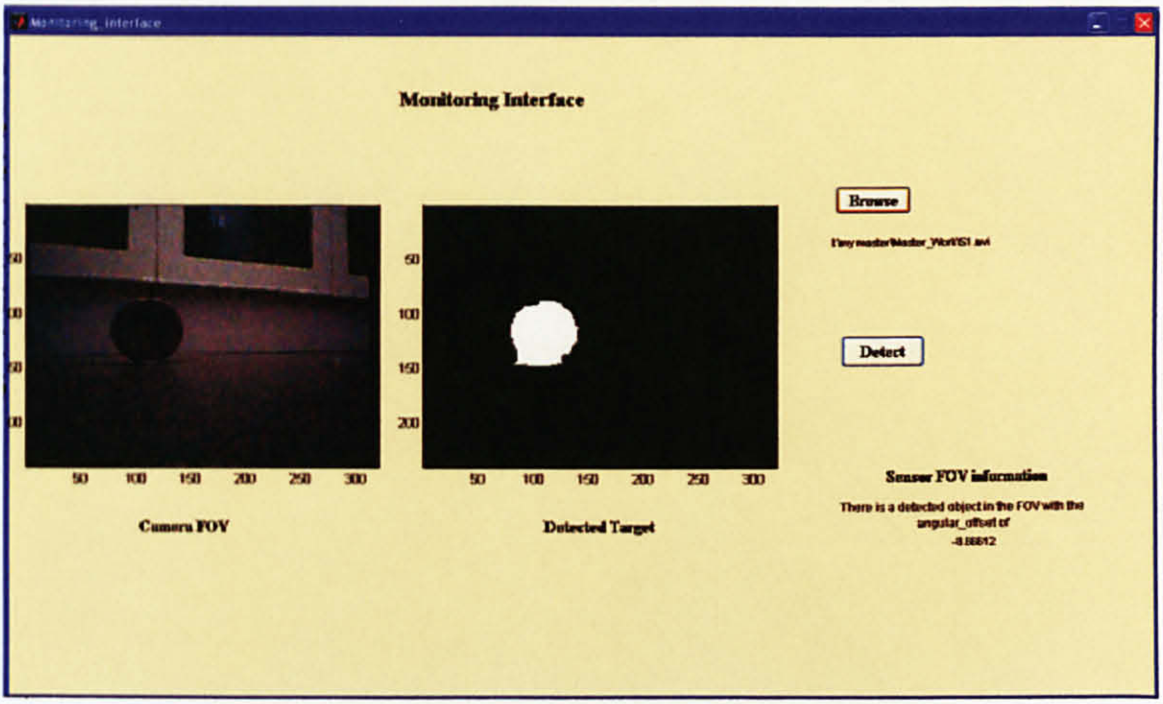
```
function varargout = Searching_Phase(varargin)
% SEARCHING_PHASE M-file for Searching_Phase.fig
%        SEARCHING_PHASE, by itself, creates a new SEARCHING_PHASE or
raises the existing
%      singleton*.
%
%       H = SEARCHING_PHASE returns the handle to a new SEARCHING_PHASE
or the handle to
%      the existing singleton*.
%
```

```
%       SEARCHING_PHASE('CALLBACK',hObject,eventData,handles,...) calls
the local
%       function named CALLBACK in SEARCHING_PHASE.M with the given
input arguments.
%
%       SEARCHING_PHASE('Property','Value',...) creates a new
SEARCHING_PHASE or raises the
%       existing singleton*.  Starting from the left, property value
pairs are
%       applied to the GUI before Searching_Phase_OpeningFunction gets
called.  An
%       unrecognized property name or invalid value makes property
application
%       stop.  All inputs are passed to Searching_Phase_OpeningFcn via
varargin.
%
%       *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only
one
%       instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Searching_Phase

% Last Modified by GUIDE v2.5 17-Nov-2008 17:55:04

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Searching_Phase_OpeningFcn, ...
                   'gui_OutputFcn',  @Searching_Phase_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end


% End initialization code - DO NOT EDIT


% --- Executes just before Searching_Phase is made visible.
function    Searching_Phase_OpeningFcn(hObject,    eventdata,    handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Searching_Phase (see VARARGIN)

% Choose default command line output for Searching_Phase

handles.output = hObject;
% Update handles structure
guidata(hObject, handles);
set(handles.pushbutton2, 'Enable', 'off');
set(handles.text6, 'Enable', 'off');

% --- Outputs from this function are returned to the command line.
function  varargout  =  Searching_Phase_OutputFcn(hObject,  eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


%% Executes on button press in Browse button.

function frames=pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)
[fileName,pathName] = uigetfile({'*.avi'}, 'Choose the source video');
NAME = fileName(1:end-4);
if(size(fileName,2)>3)
    video = mmread(strcat(pathName, fileName),[],[6 10]);
    frames=video.frames;
    num_frames=length(video.frames);
    icon=imread('icon.jpg');
    set(handles.text7, 'String', [pathName fileName]);
    set(handles.pushbutton2, 'Enable', 'on');
    handles.frames=frames;
    handles.num_frames=num_frames;
    axes(handles.axes1);
    subimage(handles.frames(1).cdata);
    msgbox('      The Video has been loaded Successfully       ','Search
Video','custom',icon)
    guidata(hObject,handles)
else
    set(handles.text7, 'String', 'No video selected');
end
```

```
%% Executes on button press in Detect button.

function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)
set(handles.text6, 'Enable', 'on');
se = strel('disk',11);
FOV=tand(round(51/2));

for i = 1:handles.num_frames
    axes(handles.axes1);
    subimage(handles.frames(i).cdata);
    logic_img=handles.frames(i).cdata(:,:,1)>=155                        &
handles.frames(i).cdata(:,:,2)<=75                                        &
handles.frames(i).cdata(:,:,3)<=75;
    close_img=imclose(logic_img,se);
    open_img=imopen(close_img,se);
    filtered_img = bwareaopen(open_img,500);
    [B,L,N] = bwboundaries(filtered_img);
    test=length(B);
    if test>0
        right=max(B{1,1}(:,2));
        left=min(B{1,1}(:,2));
        midle =round((left+right)/2);
        angular_offset=atand((2*(midle-160)/320)*FOV);

        if abs(angular_offset)<=0.7
            set(handles.text4,'String','sensor  has  been  detected  with
angular offset of');
            set(handles.text8,'String',angular_offset);
            botom=max(B{1,1}(:,1));
            top=min(B{1,1}(:,1));
            hieght=botom-top;
            set(handles.text9,'String','and hieght of');
            set(handles.text10,'String',hieght);
            distance=((600/hieght)*426.21)/100;
            set(handles.text11,'String','and distance of');
            set(handles.text12,'String',distance);
            save('data.mat', 'hieght', 'distance');
        else
            set(handles.text4,'String','sensor  has  been  detected  with
angular offset of');
            set(handles.text8,'String',angular_offset);
        end
    else
        set(handles.text4,'String','No detected sensor in the FOV');
    end
    axes(handles.axes2);
    subimage(filtered_img);
end
```

APPENDIX C.2: GRAPHIC USER INTERFACE OF THE CAMERA NODE MONITORING PHASE

In this part, the code to present a GUI of the monitoring phase of the camera node is implemented in MATLAB® and provided. This GUI, as shown in Figure C-2, loads the video by pressing the browse button and selects the video recorded by the node while monitoring the sensing field.



There is a detected object in the FOV with the angular_offset of -8.88612

Figure C-2: Snapshot of the Monitoring GUI

```
function varargout = Monitoring_interface(varargin)
% MONITORING_INTERFACE M-file for Monitoring_interface.fig
%                 MONITORING_INTERFACE,  by  itself,  creates  a  new
MONITORING_INTERFACE or raises the existing
%      singleton*.
%
%          H  =  MONITORING_INTERFACE  returns  the  handle  to  a  new
MONITORING_INTERFACE or the handle to
%      the existing singleton*.
%
%         MONITORING_INTERFACE('CALLBACK',hObject,eventData,handles,...)
calls the local
```

```
%       function named CALLBACK in MONITORING_INTERFACE.M with the given
input arguments.
%
%           MONITORING_INTERFACE('Property','Value',...) creates a new
MONITORING_INTERFACE or raises the
%       existing singleton*.  Starting from the left, property value
pairs are
%       applied to the GUI before Monitoring_interface_OpeningFunction
gets called.  An
%       unrecognized property name or invalid value makes property
application
%       stop.  All inputs are passed to Monitoring_interface_OpeningFcn
via varargin.
%
%       *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only
one
%       instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
Monitoring_interface

% Last Modified by GUIDE v2.5 17-Nov-2008 17:55:04

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn',  @Monitoring_interface_OpeningFcn,
...
                   'gui_OutputFcn',   @Monitoring_interface_OutputFcn,
...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);

if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end


% End initialization code - DO NOT EDIT


% --- Executes just before Monitoring_interface is made visible.
```

```
function  Monitoring_interface_OpeningFcn(hObject,  eventdata,  handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)
%  varargin     command  line  arguments  to  Monitoring_interface  (see
VARARGIN)

% Choose default command line output for Monitoring_interface

handles.output = hObject;
% Update handles structure
guidata(hObject, handles);
set(handles.pushbutton2, 'Enable', 'off');
set(handles.text6, 'Enable', 'off');

% --- Outputs from this function are returned to the command line.
function varargout = Monitoring_interface_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


%% Executes on button press in Browse button.

function frames=pushbutton1_Callback(hObject, eventdata, handles)

% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)

[fileName,pathName] = uigetfile({'*.avi'}, 'Choose the source video');
NAME = fileName(1:end-4);
if(size(fileName,2)>3)
    video = mmread(strcat(pathName, fileName),[],[6 10]);
    frames=video.frames;
    num_frames=length(video.frames);
    icon=imread('icon.jpg');
    set(handles.text7, 'String', [pathName fileName]);
    set(handles.pushbutton2, 'Enable', 'on');
    handles.frames=frames;
    handles.num_frames=num_frames;
    axes(handles.axes1);
    subimage(handles.frames(1).cdata);
    msgbox('      The Video has been loaded Successfully      ','Search
Video','custom',icon)
```

```
    guidata(hObject,handles)
else
    set(handles.text7, 'String', 'No video selected');
end


%% Executes on button press in Detect button.

function pushbutton2_Callback(hObject, eventdata, handles)

% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB®
% handles    structure with handles and user data (see GUIDATA)

set(handles.text6, 'Enable', 'on');
frames=handles.frames;
se = strel('disk',7);
FOV=tand(round(51/2));
background = handles.frames(1).cdata;
for i = 1:handles.num_frames
    S=frames(i).cdata;
    handles.S = S;
    axes(handles.axes1);
    subimage(handles.frames(i).cdata);
    detec=background-handles.frames(i).cdata;
    logic_img=detec(:,:,1)>40&detec(:,:,2)>40&detec(:,:,3)>40;
    double_img=double(logic_img);
    close_img=imclose(double_img,se);
    filtered_img = bwareaopen(close_img,500);
    [B,L,N] = bwboundaries(filtered_img);
    test=length(B);
    if test>0
        right=max(B{1,1}(:,2));
        left=min(B{1,1}(:,2));
        midle =round((left+right)/2);
        angular_offset=atand((2*(midle-160)/320)*FOV);
        set(handles.text4,'String','There is a detected object in the
FOV with the angular_offset of ');
        set(handles.text9,'String',angular_offset);
    else
        set(handles.text4,'String','No object in the FOV');
        set(handles.text9,'String',' ');
    end
    axes(handles.axes2);
    subimage(filtered_img);
end
```