

Encoded Depth Panorama based Information Flow in a Multi Agent System

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering

in

Software Engineering

Submitted By

Sachin Meena

Roll No. 801331026

Under the supervision of:

Dr. Jhulik Bhattacharya

Assistant Professor, CSE Department



COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY


PATIALA – 147004

May 2015


CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, "*Encoded Depth Panorama based Information flow in a Multi Agent System*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in Software Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Jhilik Bhattacharya and refers other researcher's work which are duly listed in the reference section.

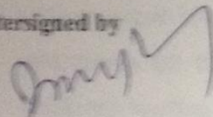
The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.


(Sachin meena)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.


(Dr. Jhilik bhattacharya)
Assistant Professor,
CSE, Department

Countersigned by

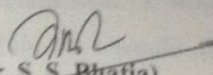

(Dr. Deepak Garg)

Head

Computer Science and Engineering Department

Thapar University

Patiala


(Dr. S. S. Bhatia)

Dean (Academic Affairs)

Thapar University

Patiala

ACKNOWLEDGEMENT

First of all I would like to thank the Almighty, who has always guided me to work on the right path of the life.

This work would not have been possible without the encouragement and able guidance of my supervisor **Dr. Jhiliak Bhattacharya**. I thank my supervisor for their time, patience, discussions and valuable comments. Their enthusiasm and optimism made this experience both rewarding and enjoyable.

I am equally grateful to **Dr. Deepak Garg**, Associate Professor and Head, Computer Science & Engineering Department, a nice person, an excellent teacher and a well-credited researcher, who always encouraged me to keep going with work and always advised me with his invaluable suggestions.

I will be failing in my duty if I don't express my gratitude to **Dr. S.S. Bhatia**, Senior Professor and Dean of Academic Affairs, Thapar University, for making provisions of infrastructure such as library facilities, computer labs equipped with net facilities, immensely useful for the learners to equip themselves with the latest in the field.

I am also thankful to the entire faculty and staff members of Computer Science and Engineering Department for their direct-indirect help, cooperation, love and affection, which made my stay at Thapar University memorable.

Last but not least, I would like to thank my family whom I dearly miss and without whose blessings none of this would have been possible. To my parents, I own thanks for their wonderful love and encouragement. I would also like to thank my brother, since he insisted that I should do so. I would also like to thank my close friends for their constant support.

Date: July, 2015

Place: Thapar University, Patiala

(Sachin Meena)

Abstract

Multiagent Systems is a research area of distributed Artificial Intelligence where agents are used for modelling any entity like web service, robot, or nodes of a grid for example. These artificial agents require to communicate and coordinate efficiently; retrieving, processing and exchanging expertise with other agents in-order to solve real time complex problems that cannot be solved individually . Research in this domain has a two-fold focus. One end concentrates on developing architectures of MAS while the other end works with the communication language between the agents. This usually requires a common language, also known as an Agent Communication Language (ACL). Although ACLs were popular in software agent systems, their use was limited in robot systems where low level data like video transfer was necessary. This continued until the introduction of backchannel for transferring the low level data. Different MAS developed and discussed in literature considers target specific data processing and communication mechanisms. The thesis addresses part of a research problem which deals with a heterogeneous system with two agents used for high level parsing of the environment. The work in its entirety involves Agent 1 as a robot equipped with a pan-tilt camera, seven IR sensors , three ultrasonic sensors and a SBC. This agent is responsible for acquiring environmental information using the sensors, processing them and sending them to Agent 2, which is a computer where semantic parsing on the received data is performed. This thesis work has two main objectives. Primarily it aims to reduce data flow between the agents for efficient usage in terms of network bandwidth and storage requirements. This is done by exploiting a encoded panoramic mosaic to reduce visual data dimension communicated between agents in real time. Secondly, it proposes a model which can be utilized to recover approximate depth information using a pair of overlapping images from a single camera.

Contents

1	Introduction	1
1.1	Agent	1
1.1.1	Types of Agent	1
1.1.2	Properties of Agent	2
1.2	Multi-Agent System	3
1.3	Data Transmission	5
2	Literature Survey	7
2.1	Agent Communication Mechanisms	7
2.1.1	Establishment of Backchannels	8
2.1.2	Flow of Backchannel	9
2.1.3	Termination of Backchannel	9
2.2	Image Panorama	10
2.2.1	Feature Detection	11
2.2.2	Image Registration	12
2.3	Data Distribution	15
3	Problem Statement	17
4	Methodology	19
4.1	Overview of Our Approach	19
4.2	Panorama Creation Approach	22
4.2.1	Feature Point Detection	22

4.2.2	Image Registration	24
4.2.3	Color Balancing	27
4.2.4	Image Stitching	28
4.3	Moving Object Detection and Representation	28
4.4	Encoded Panorama	30
4.5	Data Communication	31
4.6	Image Reconstruction from Panorama	33
4.7	Depth Mapping in Panoramic Image	34
5	Experiments and Result	36
5.1	Outdoor Scene 1	36
5.2	Indoor Scene 1	39
5.3	Outdoor Scene 2	42
5.4	Depth Measurement using depth model	46
6	Conclusion and Future work	48

List of Figures

1.1	Planning Agent Architecture	2
1.2	MAS architecture	4
2.1	Image Registration	13
2.2	Image Stitching using Harris corner detection	15
4.1	Action performed by agent 1 and agent 2	20
4.2	Flowchart of communication between agent 1 and agent 2	20
4.3	Block diagram of approach	21
4.4	Block diagram for Panorama creation	22
4.5	Feature point of images using SURF Feature	23
4.6	Feature Matching	24
4.7	Image Before Registration	26
4.8	Image After Registration	26
4.9	Color balancing between images	27
4.10	Image at time t1	29
4.11	Images at time t2	29
4.12	Image difference	30
4.13	Example of data hiding	31
4.14	Structure which send with panorama to agent 2	33
5.1	Three input images for panorama creation of an outdoor scene	36
5.2	Feature detection for outdoor scene 1	37
5.3	Matching Points for outdoor scene 1	37

5.4	Outdoor Panorama 1 at time t_1	37
5.5	Bounding Box of moving object at time t_2 and t_3 superimposed on Outdoor 1 panorama of t_1	38
5.6	Image Reconstruction for Outdoor scene 1	39
5.7	Three input images for panorama creation in an Indoor Scene	39
5.8	Feature Detection for Indoor Scene	39
5.9	Matching Points for Indoor Scene	40
5.10	Indoor Panorama at time t_1	40
5.11	Bounding Box of moving object at time t_3 superimposed on Indoor panorama of t_1	41
5.12	Image Reconstruction for Indoor Scene	42
5.13	Four input images for panorama creation of an outdoor scene	43
5.14	Feature detection in Outdoor Scene 2	43
5.15	Matching Points in Outdoor Scene 2	44
5.16	Panorama at time t_1 for Outdoor Scene 2	44
5.17	Bounding Box of moving object at time t_4 superimposed on Outdoor 2 panorama of t_1	45
5.18	Image Reconstruction for Outdoor Scene 2	46
5.19	Input Image for Depth Measurement	47

List of Tables

2.1	Establishment of Backchannel	8
5.1	Depth measurement Entry	47

Chapter 1

Introduction

1.1 Agent

Agent can be a machine, a person, a piece of software or other things. The basic definition of an agent is one who acts. An agent is any entity which senses its environment using sensors and performs act upon that environment through various types or through the effectors [1]. There is a relation between agent and AI. The aim of agent based systems is to develop the coordination among different agents to organize the collective behavior of the agent. Agent in multi-agent system has several properties likes autonomy, local view and de-centralization. The Multi-agent system is used to solve the problem which is very difficult and approximately impossible to solve by a single agent.

1.1.1 Types of Agent

Types of agent is based on the characteristics of agent. Some of the agents are discussed below.

Mobile Agent Mobile agent or software agents are those that can be transported from one environment to another environment.

Classical rule-based Agent An agent which deals with symbolic representation of the world which include knowledge about its environment, its goals and some rules that

apply in the world. Such type of agent is also called as planning agent. This encompasses an important class of agents. The general architecture of this class of agents is shown in figure 1.1.

Reactive Agent Reactive agents deal with the interaction between agents and the agent's environment. The behavior of the agents depend on the mapping between different states indicated by other agents or environment.

Learning Agent Learning agent is used or capable to improve the capability of other agent types. Learning agent deals with several learning algorithms to solve the problem.

Hybrid Agent Hybrid agent is a combination of several subsystems which are able to react very fast to the event by mixing the ideas of several approaches.

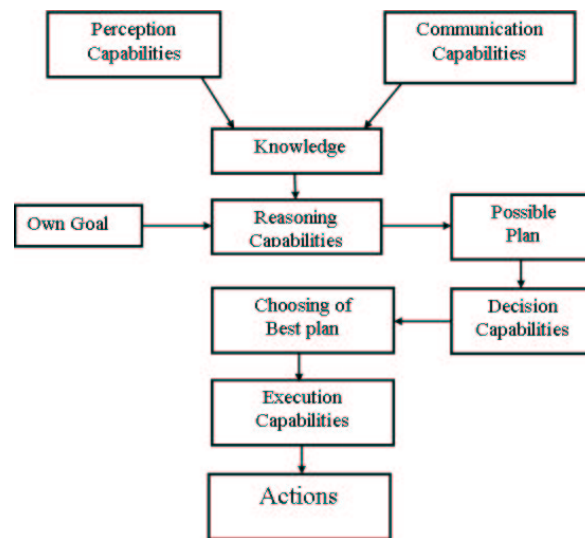


Figure 1.1: Planning Agent Architecture

1.1.2 Properties of Agent

The agent contains several properties, as discussed below.

- Mobile : Agent must be capable to transfer itself from one environment to another environment.

- Autonomous : Agent must be capable to perform some task without external intervention.
- Interactive : Agent must have the property to communicate with the different agent and with different environment.
- Adaptive : Agent must be capable to respond to other agents or respond to its environment.
- Coordinative : Agent must be capable to perform some task in shared environment with agents.
- Cooperative : Agent must be capable to coordinate with agents for achieving some purpose. It means that agent fail and succeed together. Collaboration or cooperative are the same things.
- Competitive : Agent must be capable to coordinate with agents, but success of one agent does not imply the failure of other agent. Competitive property is opposite of cooperative or collaboration property.

1.2 Multi-Agent System

Agents are computation entities and multi-agent system (MAS) is a collection of these computational entities and relation between them. MAS is a computerized system which is composed of multiple agents that are interacting within the same environment. Each agent does its specific job and is based on some algorithm. Autonomous agent observes their own environment and performs computation based on these observations. The agent behaviors are deterministic or non-deterministic and its action and behavior modify the environment or change the locations in the environment in which agent are located. Task is distributed into agents and the task for each agent is reacting based on the changes in environment by using its own skills and knowledge for decisions. Each of these agents have a specific line of action depending on which they interact with each other. Different agents in a MAS may be engaged in the same task together or different agents may perform a task in parts

synchronously or asynchronously. These specify the different reasons why MAS are better than single agents as pointed below. Figure 1.2 show the general architecture of multi agent system.

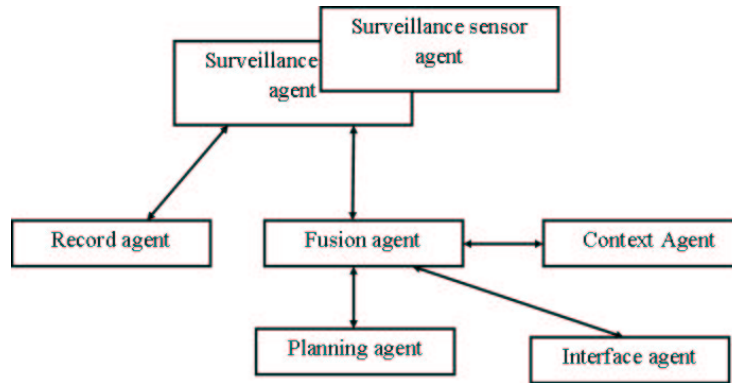


Figure 1.2: MAS architecture

Computational Efficiency This parameter is utilized when any computation is speeded up by using multiple agents to receive data parallelly [1].

Reliability Reliability is achieved when agents with interconnected and redundant capabilities are found dynamically; hence facilitating backup and reducing component failures.

Robustness The system which fuse data that comes from a number of sources is more robust then those where the data comes from a single source . This refers to both data from homogeneous as well as heterogenous agents.

Maintainability MAS being composed of multiple single agent;makes component maintenance easier to due to its modularity.

Responsiveness When a complex task is divided among many single agents, the response time for each reduces as it requires to propagates the modularity instead of the whole system.

The main challenge for proper functioning of a Multi-agent system lies in the coordination between the individual agents. This involves reliable data communication between agents as

further discussed in the next section.

1.3 Data Transmission

Video, audio, image, text and various types of data are required to be transferred from agent to agent depending on the type of application the MAS is involved in. Data transfer is physical transmission of data from one point to another point. Reliable data transmission involves available network bandwidth. Bulk data transfer results in slow system response, jitter, data loss and higher storage requirements at the receiver end. As a solution to this problem, research considers developing faster network devices, error detection and correction algorithms, data compression techniques, higher capacity storage devices. Various methods are reported in literature for different data communication. Picture Transfer Protocol (PTP) is used to transfer the image data between devices [3]. Real-time Transfer Protocol (RTP) is used to transfer audio and video data over the IP network; RTP is mainly used for entertainment systems which involves video teleconference application, live streaming, streaming media and television services. Data transfer between agents in a MAS usually requires a common language, also known as an Agent Communication Language (ACL). Some widely used ACLs are KQML and FIPA-ACL [4]. Although ACLs were popular in software agent systems, their use was limited in robot systems where low level data like video transfer was necessary. This gave rise to the use of backchannels for image and video data. However, optimizing data dimension before transferring it is equally important and is one of the main factors discussed in this work. The research mainly deals with optimizing bulk video data transfer between agents. This is done by exploiting the concept of a panoramic mosaic. There are many situations in which background of video is same but there are changes in foreground of video. For example video lecture in which background are same, while the position of lecturer changes in the foreground. Similar is the case for stage shows, fixed surveillance cameras and many more. These scenes can be dealt with by separating the static and dynamic parts of a scene and appropriately providing both the information without redundancy in a panorama.

The proposed data dimension optimization technique involves scene panorama creation, dy-

namic contents segmentation, data packet encoding at sender side and data reconstruction at receiver side. These are discussed in details in chapter 4. Chapter 2 gives related work corresponding to each of the steps involved in data preprocessing prior to sending, and post processing after receiving. Experimental Results and Observations are presented in chapter 5. This includes the demonstration of the data encoding and reconstruction phases in different indoor and outdoor scenes. The amount of transferring time and storage requirement reductions are discussed. Chapter 6 briefs the conclusion of the thesis work and presents some directions for future work.

Chapter 2

Literature Survey

As discussed in the previous chapter, the thesis aims to focus on data dimension reduction techniques prior to transmission between agents in a multi agent system using a panoramic mosaic. In view of the above the thesis first discusses some agent communication mechanisms. Following this panorama generation techniques and data encoding techniques are presented. These are elaborated in the following sections.

2.1 Agent Communication Mechanisms

In multi-agent system, agents interact and communicate with each other. For communication or interacting purpose a communication language is required. Agent Communication Language or ACL is such type of language. ACL is a language with defined semantics, pragmatics and syntax which is the basis of communication between designed and developed software agents. The most popular ACLs are KQML and FIPA-ACL used in many applications. Knowledge Query and Manipulation Language (KQML) is a language and protocol for communication between different software agents. FIPA-ACL is based on speech act theory. For exchanging the message between two agents a common meeting point is required, through which messages are received and delivered. Most of the physical robot team do not use ACL, which is developed by MAS community. While they are used with an ad-hoc solution, they define their own protocol which specifies to their system. Following these a two way communication strategy using ACL as well as backchannels was used mainly due

to permit low level bulk data transfer like video. This solution realizes both these goals by uniting the strengths of ACLs and the methods used by the robot community. Backchannels, has been implemented on the RETSINA MAS [27], which uses KQML. The messages are sent over backchannel and agreed upon at the ACL level. The meaning of message and content which is exchanged is specified by referencing a user defined format description library. Library contains detail of how to parse message and message semantic description, for example video. In some applications translator agent is used to convert the message into human readable form or converts message into agent communication language. The main purpose of ACL is to provide a facility of communication between agents in MAS. ACL must support the flow control, establishment and termination of backchannels [28].

2.1.1 Establishment of Backchannels

The establishment of backchannels begins when one agent send or receive message from other agent. If initiator want to become a sender than server send a request of "client line" to other agents. If initiator want to become a server than server send a request of "server line" to other agents. In both the cases request indicate which type of data is transferred. Detail information related to establishment is shown in table 2.1.

Communication Act	Description
Backchannel request	Request <line type> protocol <message type> Server name <name>, reference name <name>
Accept request	Accept <line type> reference number <n> Server name <name>
Connection status	Connection <status> Reference number <n> , server name<name>

Table 2.1: Establishment of Backchannel

Backchannel are differentiating easily using reference number, which is unique for the server. All connection regarding backchannel must be specifying the reference number and server. "Line Type" is either client type or server type. Communication "status" may be accepted, failed, terminated and requested.

2.1.2 Flow of Backchannel

Backchannel provide a facility of message or data transfer between different agents. For transferring the information flow of backchannel is necessary. It is important to start and stream the flow of data, repeat message, frequency of transmission, status to conform the connection of backchannel and respond total number of transmission which is sent.

2.1.3 Termination of Backchannel

Either client or server may be initiator, so the ability to terminate the connection over the backchannel is equally for both client and server. There are three protocol to terminate the connection of backchannel.

Handshake Protocol

In handshake method first, agent 1 send a connection close request message through the ACL to agent 2. Agent 2 received message from agent 1 and close the TCP connection and send a response message to agent 1 that connection is closed.

Warning Method

In this method Agent 1 sends warning message that the line will be closed in ACL. Warning message is sent at 5 and 30 second. During that time Agent 2 close the TCP connection and inform Agent 1 that connection is terminated. If within time duration Agent 2 does not want to terminate the connection, then Agent 1 terminate the connection and sends a message to Agent 2.

Cold Shoulder Method

In this method either of the agent close the TCP connection without informing the other agent hence leading to loss of data. Message in the ACL is sent to inform the other agent that the connection is already terminated.

The connection flow and backchannel library structure for the data transmitted between the agents, used in the present work is further described in chapter refmeth.

2.2 Image Panorama

The word panorama is widely used in photography, painting, film, drawing and 3-D model. The word came in the dictionary in 18th century by Robert Barker an Irish painter. Word panorama is derived from the motion picture term panning. Panorama is a Greek word; it is a combination of word "**Pan**" means "All" and "**horama**" which means "view". Huge amount of definitions are available for "Panorama". Some of them are:

Definition 1: According to British Dictionary "panorama is a mental picture of a series of images or events".

Definition 2: "Panorama is presentation of images or a complete survey of images".

Definition 3: "A large and unbroken view of a scene or of a sequence of events is called as a panorama."

Definition 4: "A large extended picture or series of pictures of a scene, unrolled before spectators a part at a time so as to appear continuous [6]."

Panorama is generated by stitching images on the basis of some common information or common pixels. The panorama is created through feature detection and feature matching techniques. Both of these techniques are used in many computer vision applications like object detection, tracking and registration. For creating a panoramic image registration and feature detection of images are required. There are several automated softwares which exist to create a panorama in an offline process. Adam et al. [7] align the camera image by viewfinder alignment work. Camera images are aligned with the help of calculating the histogram of gradients. These alignments are further refined using feature points. Currently this approach is being used in mobile phones. Another technology is used for generating a high evolution panorama by triggering the automatic capturing of high evolutionary images. This approach is however not used in real time and it requires offline processing [8]. Creation of panoramic mosaic in real time from frames with great registration differences is a problem

which was addressed by different researchers. In this context the two substeps i.e. feature detection and registration are discussed in the subsections below.

2.2.1 Feature Detection

Feature detection is the basic component required in most application of image processing and computer vision. There is no exact definition of feature detection, definition of feature points mainly depends on the type of application or problem. Interest point in an image is feature point; these feature points is used as an initial point for many vision algorithms. Repeatability is the property of feature detection. Features selected should be robust and should uniquely represent the region of interest subject to illumination and affine transformation. Some common feature types are given below:

Edge

Edges are nothing but points on the boundaries between the image regions. Edges have 1-D structure, it can be the arbitrary shape or it may include junctions. Edges in images are set of points with strong gradient magnitude. There are many methods for edge detection, which are either search based or zero-crossing based. A simple edge model is given in equation 2.1 in which f is a one-dimensional image which has one edge at $s=0$ [9].

$$f(s) = \frac{I_r - I_l}{2} \left(\operatorname{erf} \left(\frac{x}{\sqrt{2}\sigma} \right) + 1 \right) + I_l \quad (2.1)$$

σ is blur scale parameter of the edge, and edge left side intensity is $I_l = \lim_{x \rightarrow -\infty} f(s)$, edge right side intensity is $I_r = \lim_{x \rightarrow \infty} f(s)$.

Corner or Interest Point

Corner or interest points are used interchangeably. It is a 2-D structure which is used in many computer vision algorithms. A corner also defines a point at which different edge direction intersect in a local neighborhood point. The weighted SSD between two patches are represented by s [11].

$$s(x, y) = \sum_u \sum_v w(u, v) (I(u + x, v + y) - I(u, v))^2 \quad (2.2)$$

Blobs or regions of interest

In image processing blob detection is the method which is basically used for detecting regions in an image which differ in some properties like color, brightness. All points in blob detection are same in some sense to each other. There are two classes of blob detectors one is the differential method which is based on the derivatives of a function with respect to position. Other method is based on local extreme which is based on local maxima and local minima of the function. Blob detection is mainly based on Laplacian of the Gaussian. Suppose the input image is $f(i, j)$ then the Gaussian kernel is given in equation 2.3 [10].

$$g(i, j, t) = \frac{1}{2\pi t^2} e^{-\frac{i^2 + j^2}{2t^2}} \quad (2.3)$$

2.2.2 Image Registration

Registration is the determination of a geometrical transformation that align points in one view of an object with corresponding points in another view of that object or another object [12]. Image registration is the process of aligning images from two or more data sets. Image registration involves integrating the images for extracting the information which is impossible to obtain from a single image. Image registration is used in many applications like medical imaging, remote sensing and cryptography and many more. An effective image registration method is based on the situation dependent and can be complex and time consuming process. Effective image registration requires selection of a point transformation model to provide a reference point between images, and a method is required for comparing information to identify the parameters required to properly align the images. For image registration one image is taken as a reference image, one image is taken as a source image that will be converted into the reference image. For automatic image registration two approaches are their; one is intensity based and other is feature based. In Intensity based registration image alignment is based on their relative intensity pattern and in features based registration image alignment is based on feature detection, extraction and feature matching. Image registration is classified based on the transformation models. First category of transformation model is a linear transformation which allows the scaling, affine, rotation translation

and other transformations. A second category of transformation model includes nonrigid transformation which is capable of wrapping the target image to align with the reference image. Nonrigid or elastic transformation includes basic functions like compactly supported transformation, thin plate and multiquadrics, physical communication models and deformation models. Figure 2.1 shows the image registration in which image (A) is source or input image, image (B) is reference image according to reference image source image is aligned, image (C) show the target image corresponding to the reference image.



(a) Input Image



(b) Reference image



(c) Target Image

Figure 2.1: Image Registration

For panorama creation, image registration is important, without image registration a perfect panorama is not possible. Mosaic construction or panorama creation is used in various applications like video surveillance, remote sensing, satellite photography, virtual environment. There are several different approaches for creating a panorama, which is classified into direct methods and feature based methods. Direct methods are iterative methods which discover the parameter set by using the information of all pixels for reducing the sum of square

difference. For direct method a good initial parameter values are required. While on other hand feature based methods are commonly focus on image tracking [13] [35]. Some direct and feature-based methods are discussed here. Segmentation based methods are based on detecting the matching areas of the images. It consists method which are based on color or gray images. Segmentation based algorithms can be classified into images based, region based, fuzzy based and histogram thresholding . In histogram thresholding based algorithm valleys and peaks in the image are identified [14]. It works for simple images but fails if input images are varied by some complex transformation (affine, rotation). Another method is a phase correlation method to determine the overlapping areas in pairs of images. This method is based on cross power spectrum between the pair of images. But some disadvantage of this method is that it deals with the huge computation burden because computation is performed repeatedly on each pixels of image [15]. Feature based methods work by identifying the features which are not affected by camera. These features, in the images include corner, edge and ridges, Harris corner detector method, FAST detector, SUSAN detector [16], Scale Invariant Feature Transform. SIFT is another feature detection and extraction technique for color images which is proposed by David G. Lowe. SIFT method and deals with geometric transform. SIFT gives a better accuracy because it gives thousand of feature point for a single image, but a lot of feature point's increase the computational burden [17], [29]. For feature extraction Lowe used a modified K-D algorithm which is also called Best-Bin First (BBF) search method. Speeded- Up Robust Features (SURF) is an efficient algorithm for feature detection and extraction which is used in many applications. It provides a better balance between common point and feature points [18]. A fully automatic feature-based approach to combine a sequence of images into a single 2D seamless mosaic (panorama) was proposed in [33]. Removing seam after stitching or photometric adjustment is another subtask of panorama creation. There are two kinds of image stitching methods: transition smoothing and optimal seam finding [19].

Figure 2.2 shows that first images are taken as an input, then features are extracted from input images and matched with the other image pair. The images are then registered using the matching pair of points and is stitched along one of the matching point to create a panorama post color matching or blending to adjust illumination differences between the

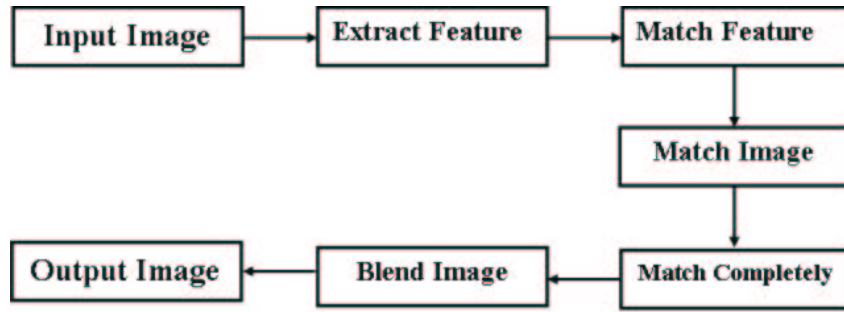


Figure 2.2: Image Stitching using Harris corner detection

two images.

Once panorama of the scene is created, the next task is to transfer it to the Agent. However in this case a single panorama consists of a sequence of panoramas. This is achieved by recording a single panorama and the changes to it over the time frames. Each time frame has a subimage which contains the region which differs from the base panorama. These subimages are encoded in the main panorama using image steganography techniques. These can be reconstructed by imposing the corresponding subimage in the base panorama for each time frame. The next section discusses common steganographic techniques reported in literature.

2.3 Data Distribution

A lot of image steganographic techniques have been used by different researchers. Different steganography techniques vary according to the number of channels used for data hiding (one or all three), particular pixels selected (all pixels, random pixels or edge pixels for example) and number of bits used in each pixel (one or two). Some of them are discussed here. In S.Shanmuga Priya et. al's [20] article data hiding is performed by using a pair of pixels, in which one bit of information is stored in the LSB of first pixel and second pixels stored another bit of information. In [21] the message is inserted in last two bits of pixel. 0 is

inserted when last two bits of pixel value are 00/01. Otherwise by adding and subtracting into pixel value we can insert 0. Similarly 1 is inserted when last two bits of pixel values are 10/11. Steganographic techniques also use RGB color images. In RGB image there are three channels red, green and blue. One channel used as a pointer which decides embedding capacity in the other two channels. Here LSB on any one of the channels decide how the data is hidden into other two channels. When last 2 bits of pointer channel are 00, no information is hidden, if last two bit are 01 data is hidden only on second channel, if bits are 10 it denotes that data is hidden only on first channel and if last bits are 11 , data is hidden on both the channels [22]. In Shweta Singhal et.al's [23] paper blue channel is used to store the message. Here one byte of blue factor is changed with message bit. OPAP, Hamming and Zig-Zag methods have been used by some researchers [24],[26]

This chapter summarizes agent communication languages, panorama creation and image steganographic techniques. The techniques used for each of them in the thesis for video data communication through encoded panoramic mosaic are next illustrated in chapter 4.

Chapter 3

Problem Statement

This work is done as a part of a project which involves a multi agent system which analyzes the environment. The system is a heterogeneous system with two agents. Agent 1 is a robot which is equipped with a pan-tilt camera, seven IR sensors and three ultrasonic sensors and a SBC. The sensors are connected to the SBC using a configured wireless network. The SBC gathers the sensor data and processes it before sending them to Agent 2. Agent 2 is a computer which communicates with Agent1 using ACL and backchannel. Agent 1 perceives the environment and sends the gathered data to Agent 2 which performs a semantic parsing of the data. The semantic output is send back to Agent 1 on the basis of which it performs certain actions. For example if the environment consists of a human being then the agent tries to get a full view of the face and store it for expression analysis. A thorough review of MAS demonstrates the use of multiagents for different applications. These involve the use of agent communication language as well as backchannel in case of bulk data. One important factor observed while carrying out the work was the effect of data dimension on the system performance [5]. Larger amount of environment data resulted in higher accuracy but at the cost of response time and storage space. Thus this thesis particularly focuses on setting an optimal trade off between the two. The idea stems from the fact that environmental video data captured continuously by the sensor will consist of a lot of image frames in which background of frames are same, with major or minor differences in the foreground of frames (moving object). Thus the idea of using a panorama which encompasses the dynamic content of several frames is exploited here. The main contributions of the work are:

- Segmenting the dynamic regions through a sequence of frames, and creating a panorama of the environment and using steganographic techniques to encode the dynamic scene regions.
- Reconstructing the original frame sequences from the encoded panorama received from Agent 1.
- Generating a model using full cubic polynomial coefficients which can give the approximate depth for any point on the panoramic mosaic.

Chapter 4

Methodology

This chapter describes the various phases for data dimension reduction. A brief description of the multiagent system and its communication flow is also provided.

4.1 Overview of Our Approach

A heterogeneous system with two agents is used in the thesis work. In general the system attempts to provide a high level environmental sensing. One agent has camera, and obstacle detection sensors like IR and UltraSonic attached to it. The agent collects the data from the sensors and transfers them to another agent which does high level parsing on the scene information and sends semantic information about it to the first agent. Agent 1 then navigates based on the information it gets from Agent 2 using its obstacle detection sensors. In the current work, Agent 1 perceives the environment and processes the gathered data before transmitting it to Agent 2. The received data is reconstructed at the Agent 2 before semantic processing is done over it. The process is further elaborated in the following sections. Figure 4.1 and 4.2 give the general system structure and communication flow respectively.

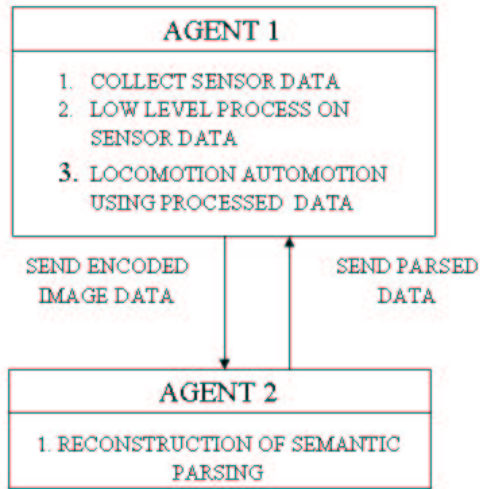


Figure 4.1: Action performed by agent 1 and agent 2

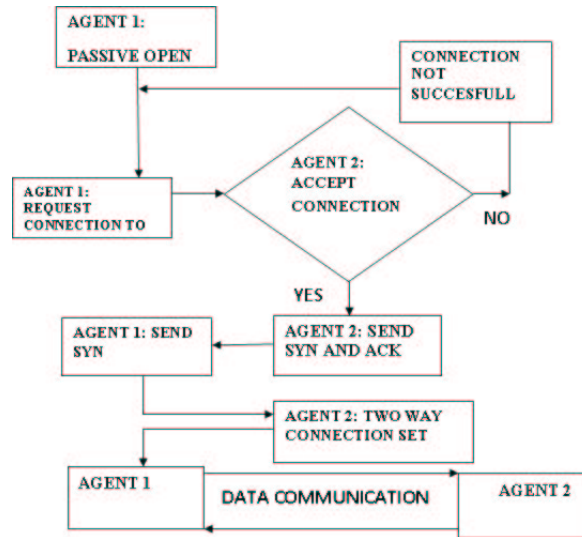


Figure 4.2: Flowchart of communication between agent 1 and agent 2

For establishing the communication path between Agents one Agent should be in passive form. Here Agent 1 is in passive open and sends a connection request to Agent 2. If

Agent 2 receives a request from Agent 1 then connection is established and agent 2 sends an acknowledgment (ACK) message to agent 1. Agent 1 receives the ACK message from agent 2 and two way communications is established. Now both agents are sending and receiving the data. Agent 1 is equipped with a pan-tilt camera which is programmed to exhibit a pan-tilt motion continuously and grab a picture. These images are to be send in Agent 2 continuously for semantic parsing. However, the current work creates a panorama of the scene at the t_0 instance. The dynamic areas of the scene are extracted using frame differencing. These are then encoded in the panorama using steganographic techniques before transferring it. At the receiver end the encoded dynamic areas are extracted and individual images are reconstructed. The approach is shown in Figure 4.3.

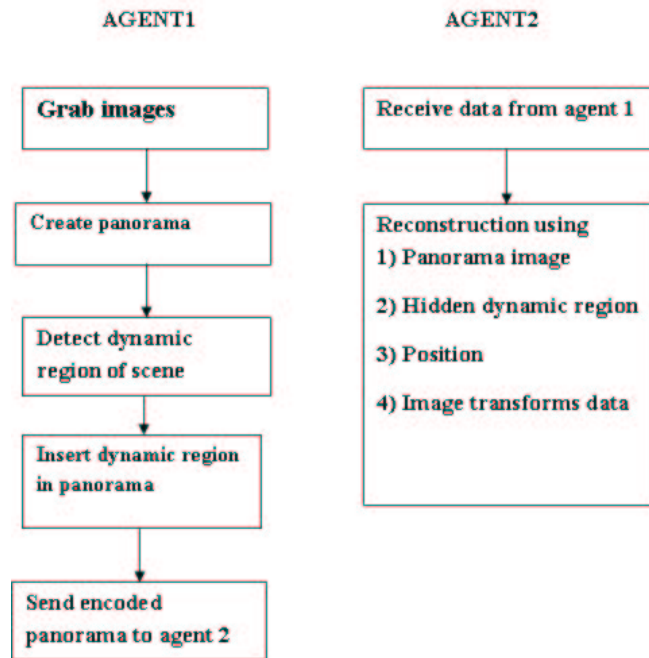


Figure 4.3: Block diagram of approach

4.2 Panorama Creation Approach

For successfully constructing a panorama from two or more individual images, two adjustments are required. First is geometric adjustment and second is photometric adjustment [29]. Geometric adjustment transforms the images into the same coordinate or computing surface [29]. This requires a reference image using which all other images are transformed so that at the end all images are on the same surface [30]. The block diagram for panorama creation is shown in the figure 4.4.

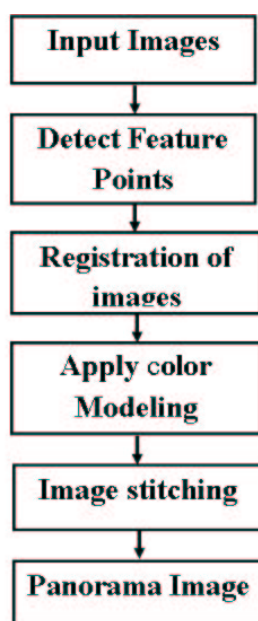


Figure 4.4: Block diagram for Panorama creation

Figure 4.4 clearly shows that panorama creation involves feature point detection, image registration, color modeling and image stitching in sequence. This process is further explained in the following subsections.

4.2.1 Feature Point Detection

The first step for panorama creation is detection of the feature points from the images. This research uses SURF (Speeded Up Robust Feature [31]) for panorama creation which

computes Hessian matrix as shown in equation 4.1 .

$$H(p, \sigma) = \begin{bmatrix} L_{xx}(p, \sigma) & L_{(xy)}(p, \sigma) \\ L_{xy}(p, \sigma) & L_{(yy)}(p, \sigma) \end{bmatrix} \quad (4.1)$$

Where $L_{xx}(p, \sigma)$ is image convolution of second derivative $\frac{d^2}{dx^2}g(\sigma)$. 20 Feature points of image using SURF features are shown in figure 4.5 which is represented by a cross (+) symbol.



Figure 4.5: Feature point of images using SURF Feature

Matched Feature is used for matching the feature points between the two images. For panorama creation some common part in images are necessary. This common part is determined using matching points between the images. Similarity is computed as a ratio between the number of point-to-point correspondences that can be established for detecting points and the mean number of points detected in two images using equation 4.2 [34].

$$S_{1,2} = \frac{c(f1, f2)}{Mean(n1, n2)} \quad (4.2)$$

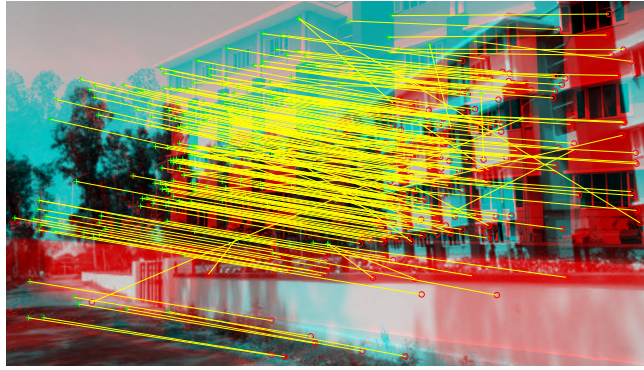
$C(f1, f2)$ denote the number of corresponding couple and $n1, n2$ the number of detecting points respectively. Once feature detection points and matching point of both the images are obtained the two images are merged and matching point position is stored for further processing. The figure 4.6 shows two input images and matching point between images.



(a) Input Image 1



(b) Input Image 2



(c) Matching point between Image1 and Image2

Figure 4.6: Feature Matching

Image registration follows matching feature point detection.

4.2.2 Image Registration

From the mathematical model of imaging geometry a pixel point p_l maps to a 3D world point P_w as shown in equation 4.3

$$\begin{aligned} p_l &= A.P_w \\ p_r &= A'.P_w \end{aligned} \tag{4.3}$$

Here A is considered as the camera parameter matrix for camera position of the left image, while A' is used for denoting parameter matrix for camera position for the right image.

Camera parameter matrix includes the intrinsic and extrinsic camera parameters. From equation 4.3 P_w can be written as

$$P_w = A^T A^{-1} A^T p_l \quad (4.4)$$

" A^{-1} here denotes a pseudoinverse operation". For image point p_r ,

$$p_r = A' P_w \quad (4.5)$$

or

$$p_r = A' A^T A^{-1} A^T p_l$$

$$p_r = H p_l, \text{ where } H = A' A^T A^{-1} A^T \quad (4.6)$$

H is thus the required transformation matrix between the right and left image points and needs to be solved. This can be done using 8 pairs of matched feature points between left and right images.

Image registration using the above method is shown in figure 4.7 in which one image is target image and all other images are registered with reference to target images. Once the registration is completed all images are in same surface.



(a) Input Image 1

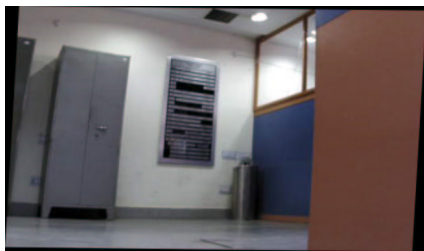


(b) Input Image 2



(c) Target Image

Figure 4.7: Image Before Registration



(a) Input Image 1



(b) Input Image 2

Figure 4.8: Image After Registration

Once all the images are on the same surface then the other factor is color differences of

image which affect the quality of the panoramic image. For that color balancing is used.

4.2.3 Color Balancing

For photometric adjustment, color blending and seam removal technique is used. This research uses a color balancing technique for photometric adjustment. The fundament frequency components of the different images are replaced with that of the reference image. Figure ?? shows the result of color balancing approach. Here, one image is taken as input image and other as reference image. According to the reference image, changes are applied to the input image.



(a) Input Image



(b) Reference Image



(c) input Image after color balancing

Figure 4.9: Color balancing between images

4.2.4 Image Stitching

Photo stitching or image stitching are used for producing a high resolution image or panorama image. Image stitching is the process in which multiple images with overlapping areas are joined along a matched feature point.

4.3 Moving Object Detection and Representation

Let $F_1^1, F_2^1, F_3^1, \dots, F_k^1$ be the individual images with which panorama is created at t_1 using k different pans of the camera. At instance t_2 the set is represented by $F_1^2, F_2^2, F_3^2, \dots, F_k^2$. The two sets may differ due to the presence of moving regions/objects or lighting changes. The latter is already handled using color balancing. Dynamic content or moving object information is retrieved by computing the difference between F_i^j and F_i^{j-1} . Position (in the image frame) and content (entire bounding box) of these dynamic regions are stored for future use.

The figures below show three images at time t_1 and t_2 . Their difference is calculated and shown. Figure 4.10 shows the image at time t_1 . Figure 4.11 shows the images at time t_2 , and figure 4.12 shows differences between images.



(a) Image F_1^1



(b) Image F_2^1



(c) Image F_3^1

Figure 4.10: Image at time t_1



(a) Image F_1^2



(b) Image F_2^2

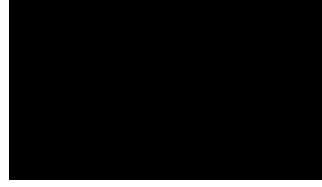


(c) Image F_3^2

Figure 4.11: Images at time t_2



(a) Difference Image 1



(b) Difference Image 2



(c) Difference Image 3

Figure 4.12: Image difference

The detected moving object is stored along with its bounding box information.

4.4 Encoded Panorama

Individual images in panorama are registered with reference to the first image. Hence for individual image reconstruction the transform value is required. Also the image boundaries in the panorama, the dynamic content and position are required. For reconstruction later at the receiver end, the dynamic content is inserted in the panorama using steganography techniques, whereas rest of the information is send in a structure. Suppose sizes of images are $R \times C$ pixels. Positions of images in panorama are calculated using equation 4.7 .

$$\begin{cases} P_i^s = JP_{i-1,i} - MP_i, P_i^e = JP_{i-1,i} + (C - MP_i) & i > 1 \\ P_i^s = 0, P_i^e = JP_{i,i+1} + (C - MP_i) & i = 1 \end{cases} \quad (4.7)$$

where P_i^s represents the starting position of image i and P_i^e represents last position of image i in the panorama. $JP(i-1, i)$ is the joining point between image $i-1$ and i in the panorama , MP_i gives the point in image i which corresponds to $JP(i-1, i)$ in the panorama. In

this research every channel of the panorama is used to store the subimages containing the bounding box of the dynamic regions . Subimage pixel bits are distributed to every pixels last two bits in the corresponding three channels. Suppose first pixel values of bounding box information of RGB channels are 234,243,213 then 4 pixels of the panorama will be required to hide the same. This is further shown in the example below:

Suppose panorama pixel values are

$p_1 = 194,192,123$, $p_2 = 213,123,222$, $p_3 = 111,123,213$, $p_4 = 198,231,098$

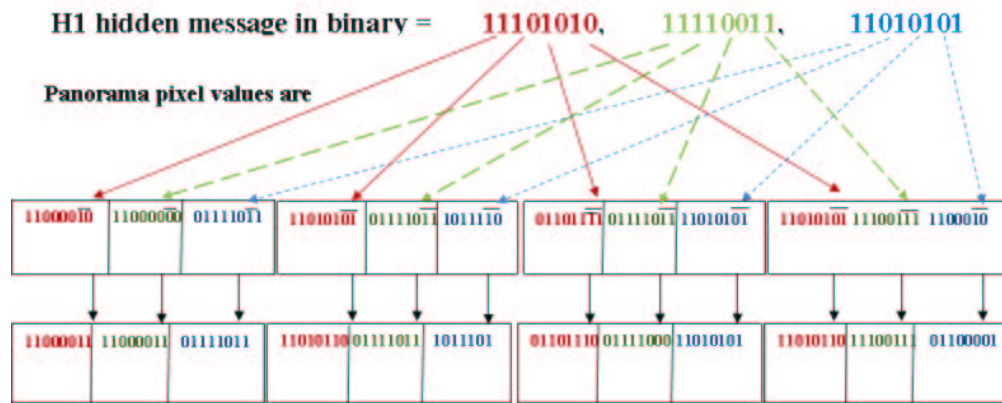


Figure 4.13: Example of data hiding

After hiding panorama pixel values are $p_1 = 195,195, 123$, $p_2 = 214,123,221$, $p_3 = 110,120, 213$, $p_4 = 214,231,97$. Example shows that maximum variation in pixel values of panorama is 3, which is not affected so much in original image. Once the panorama is encoded, it is ready to be transferred from agent 1 to agent 2 .

4.5 Data Communication

Starting and ending points of each image in the panorama and their transformation matrices are send along with the encoded panorama to the receiver agent. The panorama is send using backchannel whereas other information is send in the structure as shown in the figure 4.14 . This is beneficial when compared to Multi- agent system which uses KQML [37] for

communication as shown below

Suppose agents are transferring the position of moving object from agent1 to agent 2.

```
( tell
: sender Agent 1
: receiver Agent 2
: language default-language
: ontology default-ontology
: reply-with nob
: in-reply-to nob
: forward-to nob
: content
(:Position of moving object at t1
:Position of moving object at t2
:Position of moving object at t3))
```

Similarly if agents are communicating and sending the transform value of images from agent 1 to agent 2 then the message has following format [1].

```
( tell
: sender Agent 1
: receiver Agent 2
: language default-language
: ontology default-ontology
: reply-with nob
: in-reply-to nob
: forward-to nob
: content
(:transform value of  $F_2^2$  to  $F_1^1$  at t2)
:transform value of  $F_2^3$  to  $F_1^1$  at t3))
```

Transform value of f2 is 3x3 matrix (1.0492, -0.0899, 0, 0.0897, 0.9571, 0, 432.3960 8.4703 1.0000). Transform value of f3 is (-2.6466, -0.3919, 0, 6.2972, 3.7595, 0, -213.8597, -611.1318,

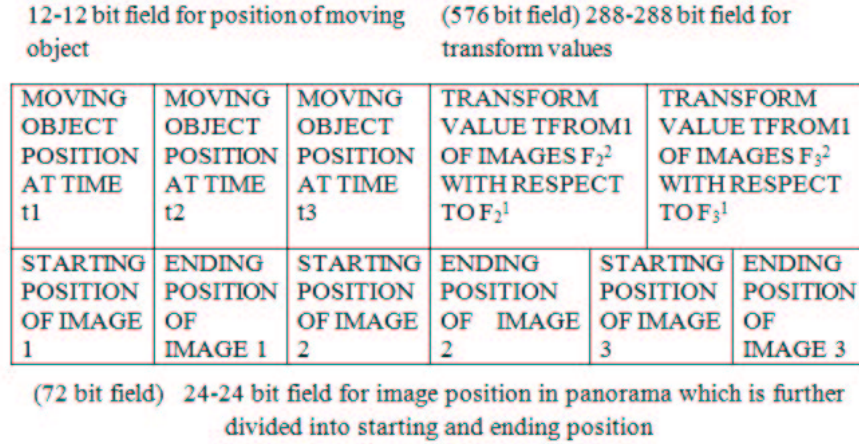


Figure 4.14: Structure which send with panorama to agent 2

1.0000).

Structure attached with panorama is 684 bit long. Panorama size is 1031x2751 pixels, maximum number of columns in panorama is 2751, and thus any column is represented by 12 bit. First 36 bit field is used for position of moving object which is further divided according to different images. Next 576 bit field is used for transform value of images with respect to image F_1^1 . Transform values are in 3x3 matrixes each value is represented in 4bytes. So 288 bit field is used for transformation value. Last field is Image position in panorama, so again starting and ending position of image is not greater than 2751 and thus represented in 12 bit each.

4.6 Image Reconstruction from Panorama

Image Reconstruction includes decomposing the panorama into the individual images using starting and ending point and transformation values. The moving regions (decoded from the panorama) are then superimposed using alpha blending [40]. The image boundary information helps extract the images which are then transformed using the corresponding

transformation matrix. The dynamic region is then blended using its position and information.

4.7 Depth Mapping in Panoramic Image

One more idea discussed in this thesis is related to measuring the depth of any point in the panoramic image. The calculating depth of the image is the main problem in computer vision system, and important in many applications like 3-D reconstruction, robotics and scene understanding. Depth estimation in a single monocular image is a very difficult task, for that a global structure of images with prior knowledge about image scene is required. There are several algorithms for depth estimation reported in literature. Some of them include a method where supervised learning is used for estimating the 1-D distance of obstacle, driving a remote control car [41]. Another computes depth from a single image mapping. This method is mainly based on capturing depth and depth relationship between image using (Markov Random Fields) MRF [42]. Training data are collected using a 3-D distance scanner, which consist set of images and ground-truth depthmaps corresponding to the images. By using the training set data MRF calculated a predicted depth. This model use L_2 (Gaussian)factor model for capturing the depth. Depth is also computed using structured light. This model is based on a pair of images which are used to test the accuracy of stereo algorithm which deal with ground-truth correspondence. This approach used a pair of cameras and light projector. By using a structured light sequence of each camera, the label (unique code) of each pixel is calculated.

The model used in the thesis exploits a full cubic polynomial coefficient mapping between the x and y disparities and depth of a particular point. A set of training data is first collected where disparities of various points in a pair of transformed images is recorded. This is shown in equation 4.8. This is repeated three times using three different transformation angles for the same set of points. Hence for the same depth value, different transformation of the same points give different disparity values. Once the three set of coefficients are obtained, given any point, the depth values χ can be computed from the disparity x and y . The same three transformations used for the training data is used during testing. This gives three different

χ values which are averaged to give the final reading.

$$\chi = \alpha_0 + \alpha_1x + \alpha_2y + \alpha_3xy + \alpha_4x^2 + \alpha_5y^2 + \alpha_6x^2y + \alpha_7xy^2 + \alpha_8x^3 + \alpha_9y^3 \quad (4.8)$$

This model is utilized to know the approximate depth of any point in the panorama at runtime. Given any point as input to the system, the image to which it belongs can be easily recovered using the start and ending points of the images in the panorama. The pan given to the camera while capturing the image is already known. Agent 1 can be panned to the same angle and moved by the computed depth to reach the point selected from the panorama. This mechanism works as moving the agent to any point selected by the user from the scene panorama.

Chapter 5

Experiments and Result

The experimental result shows panorama creation, moving object detection and reconstruction of images from the encoded panorama. Three different results are shown using indoor and outdoor images. Also, The depth estimation model is tested using a few points. The calculated and actual depth of these points are shown in the table.

5.1 Outdoor Scene 1

Three images are taken as input for creating a panorama. These input images are shown in below figure 5.1.



(a) Image F_1^1



(b) Image F_2^1



(c) Image F_3^1

Figure 5.1: Three input images for panorama creation of an outdoor scene

Figure 5.2 shows the feature point corresponding to the input images.

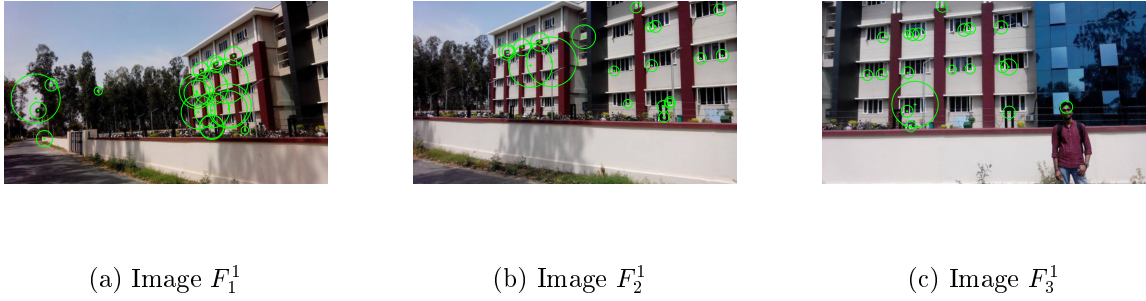


Figure 5.2: Feature detection for outdoor scene 1

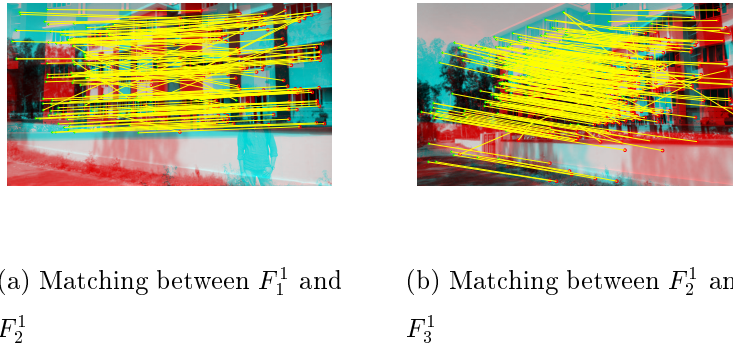


Figure 5.3: Matching Points for outdoor scene 1

Matching points and image panorama at t_1 is shown in figure 5.3 and 5.4 respectively.



Figure 5.4: Outdoor Panorama 1 at time t_1

Scene changes from t_1 at t_2 and t_3 are extracted and shown on the panorama created at

t_1 (figure 5.5).



Figure 5.5: Bounding Box of moving object at time t_2 and t_3 superimposed on Outdoor 1 panorama of t_1 .

Bounding box information and other related information with moving object are hidden into panorama created at time t_1 . Size of panorama is $1031 \times 2751 \times 3 = 8508843$ pixels, and size of moving object is $124 \times 244 \times 3 = 90768$ pixels. Each pixel of moving object uses 4 pixel values of panorama for hiding the value, so 90638 pixel data is hidden into $90638 \times 4 = 362552$ pixels of panorama, thus maximum $8508843 / 362552 = 24$ moving object can be hidden in the panorama of three images. Reconstructed image from panorama at time t_2 needs the position of bounding box, adjustment value, image position and bounding box information from panorama. After extracting the image F_2^2 from panorama dynamic region is super imposed at its position. Figure 5.6a shows boundary positions of F_2^2 image (rectangle in yellow color) and position of bounding box (circle in purple color). Figure 5.6b shows extracted image at time t_2 . Figure 5.6c shows the image after applying inverse transform and superimposing dynamic content at appropriate position.



(a) Extracted image boundary and bounding box position

(b) Extracted image at time t_2

(c) Original image transform and dynamic region superimposed

Figure 5.6: Image Reconstruction for Outdoor scene 1

5.2 Indoor Scene 1

Three images are taken as input for creating an indoor panorama. These input images are shown in figure 5.7.



(a) Image F_1^1

(b) Image F_2^1

(c) Image F_3^1

Figure 5.7: Three input images for panorama creation in an Indoor Scene

Figure 5.8 shows the feature point corresponding to the input images.



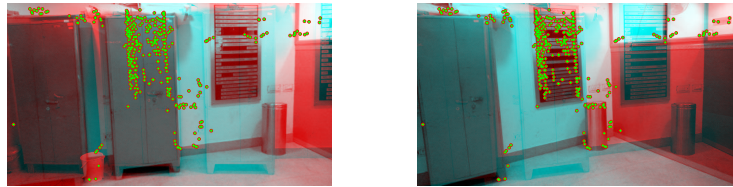
(a) Image F_1^1

(b) Image F_2^1

(c) Image F_3^1

Figure 5.8: Feature Detection for Indoor Scene

Matching points and image panorama at t_1 is shown in figure 5.9 and 5.10 respectively.



(a) Matching between F_1^1 and F_2^1

(b) Matching between F_2^1 and F_3^1

Figure 5.9: Matching Points for Indoor Scene



Figure 5.10: Indoor Panorama at time t_1

Scene changes from t_1 at t_2 and t_3 are extracted and shown on the panorama created at t_1 (figure 5.11).

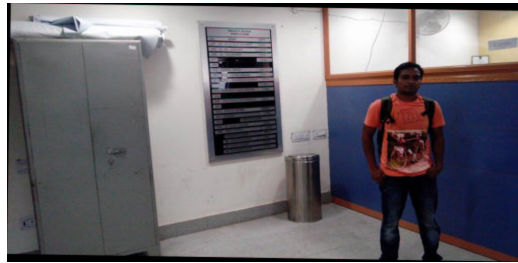


Figure 5.11: Bounding Box of moving object at time t_3 superimposed on Indoor panorama of t_1 .

Bounding box information and other related information with moving object are hidden into panorama which is created at time t_1 . In this result Size of panorama is $735 \times 1926 \times 3 = 4246830$ pixels, and size of moving object is $107 \times 251 \times 3 = 80571$ pixels. Each pixel of moving object uses 4 pixel values of panorama for hiding the value, so 80571 pixel data is hidden into $80571 \times 4 = 322284$ pixels of panorama, means maximum $4246830 / 322284 = 52$ moving object can be hidden in the panorama of three images. Reconstructed image from panorama at time t_3 needs the position of bounding box, adjustment value, image position and bounding box information from panorama. After extracting the image F_3^3 from panorama, dynamic region is super imposed at its position. Figure 5.12a shows boundary positions of F_3^3 image (rectangle in yellow color) and position of bounding box (circle in purple color). Figure 5.12b shows extracted image at time t_3 after applying inverse transform and superimposing dynamic content at appropriate position.



(a) Extracted image and bounding box position



(b) Original image transformed and dynamic region super imposed

Figure 5.12: Image Reconstruction for Indoor Scene

5.3 Outdoor Scene 2

Four images are taken as input for creating a panorama as shown in figure 5.13.



(a) Image F_1^1



(b) Image F_2^1



(c) Image F_3^1



(d) Image F_4^1

Figure 5.13: Four input images for panorama creation of an outdoor scene

Figure 5.14 shows the feature point corresponding to the input images.



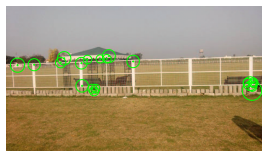
(a) Image F_1^1



(b) Image F_2^1



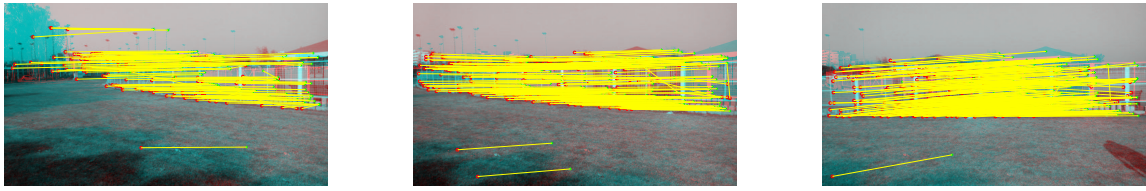
(c) Image F_3^1



(d) Image F_4^1

Figure 5.14: Feature detection in Outdoor Scene 2

Matching points and image panorama at t_1 is shown in figure 5.15 and 5.16 respectively.



(a) Matching between F_1^1 and F_2^1

(b) Matching between F_2^1 and F_3^1

(c) Matching between F_3^1 and F_4^1

Figure 5.15: Matching Points in Outdoor Scene 2



Figure 5.16: Panorama at time t_1 for Outdoor Scene 2

Scene changes from t_1 at t_2 , t_3 and t_4 are extracted and shown on the panorama created at t_1 (figure 5.17).

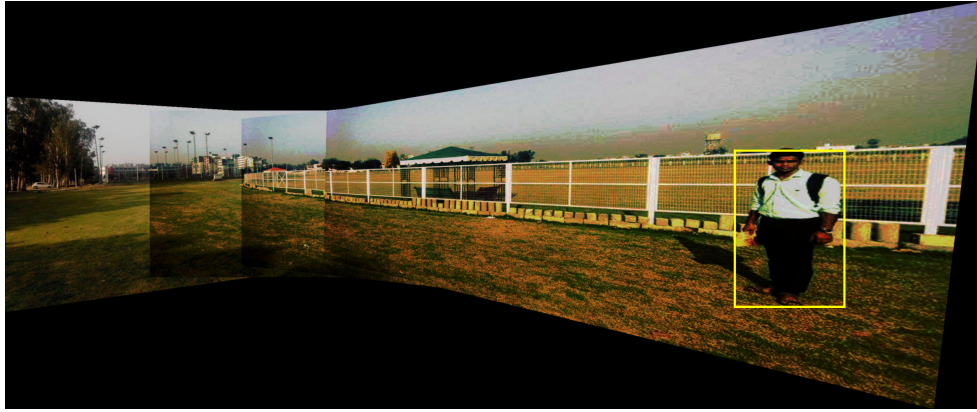
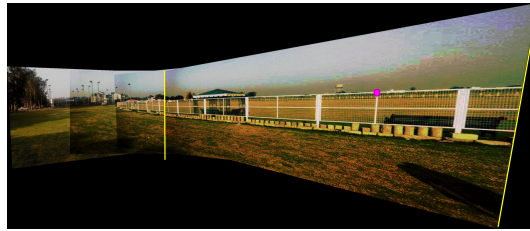


Figure 5.17: Bounding Box of moving object at time t_4 superimposed on Outdoor 2 panorama of t_1 .

Bounding box information and other related information with moving object are hidden into panorama which is created at time t_1 . Size of panorama is $1351 \times 3255 \times 3 = 13192515$ pixels, and size of moving object is $113 \times 270 \times 3 = 91530$ pixels. Each pixel of moving object uses 4 pixel values of panorama for hiding the value, so 91530 pixel data is hidden into $91530 \times 4 = 366120$ pixels of panorama, means maximum $13192515 / 366120 = 36$ moving object can be hidden in the panorama of four images. Reconstructed image from panorama at time t_4 needs the position of bounding box, adjustment value, image position and bounding box information from panorama. After extracting the image F_4^4 from panorama dynamic region is super imposed at its position. Figure 5.18a shows boundary positions of F_4^4 image (rectangle in yellow color) and position of bounding box (circle in purple color). Figure 5.18b shows extracted image at time t_4 after applying inverse transform and superimposing dynamic content at appropriate position.



(a) Extracted image and bounding box position

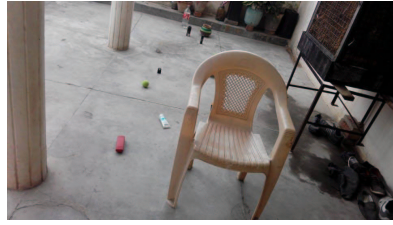


(b) Original image transformed and dynamic region super imposed

Figure 5.18: Image Reconstruction for Outdoor Scene 2

5.4 Depth Measurement using depth model

Result 4 deal with the depth measurement in images. For this method two images are taken from the same distance, but different camera position. Input images are shown in figure 5.19, in which yellow dot represents point form which depth is measured.



(a) First Image



(b) Second Image

Figure 5.19: Input Image for Depth Measurement

Table 5.1 shows the difference between the measured and calculated depth of the points shown in the figure using the given transformation values and coefficients obtained in equation 4.8.

Measured Depth (cm)	Calculated Depth	Error
152	149.23	2.77
129	130.05	1.05
116	114.25	1.75
100	104.05	4.05
93	92.22	0.78

Table 5.1: Depth measurement Entry

Chapter 6

Conclusion and Future work

This thesis argues that besides using faster network devices, advanced error detection and correction algorithms, data compression techniques, higher capacity storage devices to deal with bulk data, it is equally important to optimize data dimension before transfer. The work presented here involves data communication between a MAS system with two heterogenous agents. Environmental data gathered by Agent 1 is processed to reduce data dimension, before its transfer to Agent 2 where original data is reconstructed back. This involves the use of a panorama of the agent environment. Instead of transferring video data in bulk, the sender agent processes the frames to create a panorama. The sender updates the panorama only when there are changes in the scene. A single panorama hence encompasses the full 360 degree view at the t_0 instant as well as the dynamic information of the next t_1 to t_n instances. The dynamic content is encoded in the panorama using steganographic techniques. The value for n is set according to the amount of dynamic content of the scene. The panorama is refreshed when the Agent changes position. This reduces the data dimension transferred to other agents, which reconstructs the encoded panorama into individual image frames from t_1 to t_n . The amount of transfer is trimmed from $k \times n$ (k images per panorama and n panoramas) to a single panorama. Also if k images have a dimension $m \times n \times k$, a single panorama can be mapped as $\frac{m.k}{2} \times \frac{n.k}{2}$. The value of k depicts the number of pans used to acquire the 360 degree view and depends on the scene. The model is highly effective when there are minimal changes in the surrounding scene at fixed intervals. Experimental Results demonstrate the use of the technique in different indoor and outdoor scenes. It is observed

that there is a reasonable improvement in communication speed as well as storage space using the approach (normally 3 image information requires $1920 \times 1080 \times 3$ pixels but using these approach 1031×2751 pixels of panorama and 252 bit structure is used for same purpose in result 1). The quality of reconstructed data can be further improved by using other panorama creating techniques. The work currently uses a SURF feature matching for registration and image stitching. The color balancing technique, used to map the illumination differences between images is preferred over conventional seam removal techniques due to two reasons:- first being the speed efficiency of the former. Secondly, the panorama is further utilized for scene segmentation and semantic parsing which remains unhampered by presence of minor seams, if any. The depth model using full cubic polynomial coefficient gives an approximate depth mapping for any pixel.

Future work in this area may include comparing the current application performances in terms of communication speed and reconstruction quality while using sparse models instead of SURF or other key point features. Also, the experimental results shown in the thesis deal with a single moving object, may further be increased. Future work can consider mapping the transformations between the same moving object from frame to frame. Thus instead of using the bounding box of the moving object in each frame, a single Gaussian mixture model (denoting the moving object) and a list of transformation mappings may be used. The depth model proposed in the thesis can be improved by considering machine learning algorithms like neural network, support vector machine and bayesian classifiers.

Bibliography

- [1] Nazarko, Sergiy. "*Evaluation of data fusion methods using Kalman filtering and transferable belief*" *Measurement and Control* 203 (1989): 205.
- [2] Berna-Koes, Mary, Illah Nourbakhsh, and Katia Sycara. "*Communication efficiency in multi-agent systems*" *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*. Vol. 3. IEEE, 2004.
- [3] Ray, Dip Narayan, Somajyoti Majumder, and Sumit Mukhopadhyay. "*A Behavior-based Approach for Multi-agent Q-learning for Autonomous Exploration.*" arXiv preprint arXiv:1110.1796 (2011).
- [4] Bordini, Rafael H., Mehdi Dastani, and Michael Winikoff. "*Current issues in multi-agent systems development*" *Engineering Societies in the Agents World VII*. Springer Berlin Heidelberg, 2007. 38-61.
- [5] Ferber, Jacques. *Multi-agent systems: an introduction to distributed artificial intelligence*. Vol. 1. Reading: Addison-Wesley, 1999.
- [6] <http://dictionary.reference.com/browse/panorama>
- [7] A. Adams, N. Gelfand, and K. Pulli, "*View Finder Alignment,*" *Computer Graphics Forum (Proceedings of Eurographics)* 27, 2008, pp. 597-606
- [8] X. Wang, M. Tico, and K. Pulli, "*Panoramic Imaging System for Mobile Devices,*" Poster at the 36th international conference and exhibition on Computer graphics and interactive techniques (SIGGRAPH 2009), 2008.

- [9] Gauglitz, Steffen, Tobias Hllerer, and Matthew Turk. "*Evaluation of interest point detectors and feature descriptors for visual tracking.*" International journal of computer vision 94.3 (2011): 335-360.
- [10] Schmid, Cordelia, Roger Mohr, and Christian Bauckhage. "*Evaluation of interest point detectors.*" International Journal of computer vision 37.2 (2000): 151-172.
- [11] Gauglitz, Steffen, Tobias Hllerer, and Matthew Turk. "*Evaluation of interest point detectors and feature descriptors for visual tracking.*" International journal of computer vision 94.3 (2011): 335-360.
- [12] West, Jay, et al. "*Comparison and evaluation of retrospective intermodality brain image registration techniques*" Journal of computer assisted tomography 21.4 (1997): 554-568.
- [13] Brown, Lisa Gottesfeld. "*A survey of image registration techniques.*" ACM computing surveys (CSUR) 24.4 (1992): 325-376.
- [14] Cheng, Heng-Da, and Ying Sun. "*A hierarchical approach to color image segmentation using homogeneity.*" Image Processing, IEEE Transactions on 9.12 (2000): 2071-2082.
- [15] Zhang, Xiaoyong, Masahide Abe, and Masayuki Kawamata. "*An efficient subpixel image registration based on the phase-only correlations of image projections*" Communications and Information Technologies (ISCIT), 2010 International Symposium on. IEEE, 2010.
- [16] Pazzi, Richard W., et al. "*A novel image mosaicking technique for enlarging the field of view of images transmitted over wireless image sensor networks*" Mobile Networks and Applications 15.4 (2010): 589-606.
- [17] Lowe, David G. "*Distinctive image features from scale-invariant keypoints*" International journal of computer vision 60.2 (2004): 91-110.
- [18] Karthik, R., A. AnnisFathima, and V. Vaidehi. "*Panoramic view creation using invariant moments and SURF features*" Recent Trends in Information Technology (ICRTIT), 2013 International Conference on. IEEE, 2013.

- [19] Li, Yanfang, et al. "*Automatic image stitching using sift*" Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on. IEEE, 2008.
- [20] M.-T. Sun and A. Reibman, eds, "*Compressed Video over Networks*" Marcel Dekker, New York, 2001.
- [21] Apostolopoulos, John G., Wai-tian Tan, and Susie J. Wee. "*Video streaming: Concepts, algorithms, and systems*" HP Laboratories, report HPL-2002-260 (2002).
- [22] Conklin, Gregory J., et al. "*Video coding for streaming media delivery on the Internet*" Circuits and Systems for Video Technology, IEEE Transactions on 11.3 (2001): 269-281.
- [23] Zhang, Xiaoyong, Masahide Abe, and Masayuki Kawamata. "*An efficient subpixel image registration based on the phase-only correlations of image projections*" Communications and Information Technologies (ISCIT), 2010 International Symposium on. IEEE, 2010.
- [24] Sirsikar Sumedha, Salunkhe Jagruti "*Analysis of data hiding using Digital Image Signal Processing*" International Conference on Electronic Systems Signal Processing and Computing Technologies, 2014.
- [25] K. Ma, W. Zhang, X. Zhao, N. Yu, F. Li, "*Reversible data hiding in encrypted images by reserving room before encryption*", IEEE Trans. Inf Forensic Secur. 8(3) (2013) 553-562.
- [26] Agham, Vinit, and Tareek Pattewar. "*A novel approach towards separable reversible data hiding technique.*", Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on. IEEE, 2014
- [27] K. Sycara, M. Paolucci, M. van Velsen, and J. Giampapa, *The RETSINA MAS infrastructure*, special joint issue of Autonomous Agents and MAS, vol. 7, no. 1 and 2, July 2003.
- [28] Berna-Koes, Mary, Illah Nourbakhsh, and Katia Sycara. "*Communication efficiency in multi-agent systems*" Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on. Vol. 3. IEEE, 2004.

- [29] Juan, Luo, and Oubong Gwun. "*SURF applied in panorama image stitching*" Image Processing Theory Tools and Applications (IPTA), 2010 2nd International Conference on. IEEE, 2010.
- [30] D. G. Lowe, "*Distinctive image features from scale-invariant key points*", Int. J. Comput. Vis., vol. 60, no. 2, pp. 91-110, Jan. 2004.
- [31] Grimson, W. Eric L. "*Computational experiments with a feature based stereo algorithm*" Pattern Analysis and Machine Intelligence, IEEE Transactions on 1 (1985): 17-34.
- [32] M.-T. Sun and A. Reibman, eds, "*Compressed Video over Networks*"
- [33] Leone, A., et al. "*A fully automated approach for underwater mosaicking*" OCEANS 2006. IEEE, 2006.
- [34]] Prez, Patrick, Michel Gangnet, and Andrew Blake. "*Poisson image editing*", ACM Transactions on Graphics (TOG). Vol. 22. No. 3. ACM, 2003.
- [35] Yu, Yinan, et al. "*A novel algorithm for view and illumination invariant image matching*", Image Processing, IEEE Transactions on 21.1: 229-240, 2012.
- [36] Agarwala, Aseem, et al. "*Interactive digital photomontage*" ACM Transactions on Graphics (TOG). Vol. 23. No. 3. ACM, 2004.
- [37] Koes Berna, Mary, Nourbakhsh Illah, and Sycara Katia. "*Communication efficiency in multi-agent systems.*" Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on. Vol. 3. IEEE, 2004.
- [38] D. J. Rossi and A. S. Willsky, "*Reconstruction from projections based on detection and estimation of objects parts I and II: Performance analysis and robustness analysis*", IEEE Trans. Acoust. Speech Signal Process. ASSP-32, pp.886-906, 1984
- [39] Gan, Lijie, and Zhong Qu. "*Research on 3-D Reconstruction With a Series of Cross-Sectional Images*" Innovative Computing, Information and Control, 2006. ICICIC'06. First International Conference on. Vol. 1. IEEE, 2006.

- [40] M. Uyttendaele, A. Eden, and R. Szeliski, "*Eliminating ghosting and exposure artifacts in image mosaics*", in Computer Vision and Pattern Recognition (CVPR'01), pp. 509-516, IEEE Computer Society, 2001.
- [41] J. Michels, A. Saxena, and A.Y. Ng. "*High speed obstacle avoidance using monocular vision and reinforcement learning*", In ICML, 2005.
- [42] Saxena, Ashutosh, Sung H. Chung, and Andrew Y. Ng. "*Learning depth from single monocular images*", Advances in Neural Information Processing Systems. 2005.

List of Publications

1. Sachin Meena and JhiliK Bhattacharya “Encoded Scene Panorama based Information Flow in a Multi Agent System,” accepted at Fourth International Conference on Advances in Computing, Communications and Informatics (ICACCI-2015), Kochi, 2015 will be published by “Springer as a special volume in the prestigious Advances in Intelligent Systems and Computing Series”.
2. Sachin Meena and JhiliK Bhattacharya “Video compression using steganographic Techniques”, accepted at International Journal of Innovations & Advancement in Computer Science (IJIACS-15), Vol. 4 ISSN (2347 – 8616) 2015.

Video URL

<https://www.youtube.com/watch?v=CXI-srRyDRo&feature=youtu.be>