

# **Gesture based PTZ camera control**

*Report submitted in*

*May 2014*

*to the department of*

***Computer Science and Engineering***

*of*

***National Institute of Technology Rourkela***

*in partial fulfillment of the requirements*

*for the degree of*

***Bachelor of Technology***

*by*

***Puneet Sahoo***

*(Roll 110CS0470)*

*under the supervision of*

*Dr. Pankaj K. Sa*



**Department of Computer Science and Engineering**

**National Institute of Technology Rourkela**

**Rourkela – 769 008, Odisha, India**

# **Gesture based PTZ camera control**

## **Final Year thesis**

This report titled “Gesture based PTZ camera control” gives the details of research work carried out at Department of Computer Science and Engineering, NIT Rourkela as a part of Final Year Project of NIT, Rourkela under the guidance of Dr. Pankaj Kumar Sa, Dept. of Computer Science and Engineering.

Puneet Sahoo  
Roll No: 110CS0470

**DECLARATION**

I hereby declare that all the work contained in this report is my own work unless otherwise acknowledged. Also, all of my work has not been previously submitted for any academic degree. All sources of quoted information have been acknowledged by means of appropriate references.

Puneet Sahoo

NIT Rourkela

**CERTIFICATE**

This Final Year Project titled “Gesture based PTZ camera control” has been carried out by Mr. Puneet Sahoo, Student of National Institute of Technology, Rourkela under our supervision.

This research work was jointly carried out by the candidate and us. This report does not contain any classified information and the results can be jointly published in any of the National/International Journals / Conferences.

**Signature of Guide**

Dr. Pankaj Kumar Sa  
Assistant Professor  
Department of Computer Science and Engineering  
National Institute of Technology  
Rourkela 769 008  
Ph. No: +91 9437110444

## **ACKNOWLEDGEMENTS**

I am indebted to many people who played a critical role in completion of this project. First and foremost, I would like to extend my heartfelt thanks to Department of Computer Science and Engineering, NIT Rourkela and its entire staff for providing me with a wonderful atmosphere to carry out this research work. I would also take this opportunity to thank my project guide, Dr. Pankaj Kumar Sa, Assistant Professor, NIT Rourkela for his constant support throughout the project and his valuable suggestions. I am indebted to him for helping me throughout the course of the project, having patience with me and for guiding me at each and every step. I thank each and every one of the student of Dept. of CSE, NIT Rourkela who played a part in this project.

Finally, I thank my family for their constant care, for supporting and motivating me at every step and making sure that I had no difficulties.

## TABLE OF CONTENTS

<b>1</b>	<b>INTRODUCTION .....</b>	<b>1</b>
1.1	Endeavour of the Work.....	1
1.2	Literature Survey.....	2
1.3	Objectives of the work.....	3
<b>2</b>	<b>GESTURE DETECTION FROM A RECORDED VIDEO.....</b>	<b>4</b>
2.1	Image Segmentation .....	4
2.1.1	Background Subtraction.....	4
2.1.2	Frame Differencing.....	5
2.2	Image Morphology.....	6
2.2.1	Image Dilation .....	8
2.2.2	Image Erosion.....	8
2.3	Connected Component Analysis .....	9
<b>3</b>	<b>GESTURE DETECTION FROM LIVE CAMERA FEED.....</b>	<b>11</b>
3.1	Optical flow analysis .....	11
3.1.1	Horn Schunck method of optical flow determination .....	12
3.2	Calculation of Pan, tilt and zoom of IP camera .....	13
<b>4</b>	<b>IMPLEMENTATION.....</b>	<b>15</b>
4.1	Algorithm for detection of person making the gesture from a video shot on a static camera .	15
4.2	Algorithm for motion detection from a live camera feed and consequent determination of camera parameters.....	16
<b>5</b>	<b>RESULTS .....</b>	<b>17</b>
5.1	Background Subtraction.....	17
5.2	Frame Differencing .....	19
5.3	Image dilation, followed by erosion .....	20
5.4	Connected Component Labelling.....	21
5.5	Optical flow analysis on IP camera feed and motion detection .....	22
<b>6</b>	<b>CONCLUSION AND SCOPE FOR FURTHER RESEARCH.....</b>	<b>23</b>
<b>7</b>	<b>REFERENCES .....</b>	<b>24</b>

## LIST OF FIGURES

Figure 2-1 Background Subtraction .....	5
Figure 2-2 Frame difference.....	6
Figure 2-3 Image dilation, image on right showing the dilated image .....	8
Figure 2-4 Image erosion, image on right showing eroded image .....	9
Figure 2-5 Matrix showing labelled components .....	10
Figure 2-6 Labelled objects in an image.....	10
Figure 3-1 Optical flow analysis showing the velocity vectors .....	11
Figure 3-2 AXIS 214 PTZ camera .....	13
Figure 3-3 Block Diagram of the complete process of PTZ determination .....	14
Figure 5-1 Background Frame .....	17
Figure 5-2 Frame at time 't'.....	18
Figure 5-3 Result from background subtraction .....	18
Figure 5-4 Frame at time 't'.....	19
Figure 5-5 Frame at time 't-1' .....	19
Figure 5-6 Result after frame differencing and thresholding .....	20
Figure 5-7 Frame without dilation and erosion .....	20
Figure 5-8 Frame after dilation, followed by erosion .....	21
Figure 5-9 Labelled frame .....	21
Figure 5-10 Match between hand and label 2.....	21
Figure 5-11 Identification of person making the gesture .....	22
Figure 5-12 Motion detected.....	22

## **LIST OF KEYWORDS**

### **Keywords:**

Frame, image, grayscale, morphology, dilation, erosion, labelling, optical, pan, tilt, zoom, structure

### **Abbreviations:**

RGB – Red Green Blue

PTZ – Pan Tilt Zoom

IP – Internet Protocol

## ABSTRACT

Gesture Detection is one of the most popular fields of research among the computer vision community. It has increased importance now because of its potential application in the fields of indoor surveillance, object tracking, traffic surveillance, etc. Gesture detection and recognition enables greater communication between machines and human beings. Research in this field in future can lead to non-textual inputs, a generation where there will be no need for keyboards, mouse or even switchboards.

There are many possible techniques to detect motion in the real world. Tracking devices like gloves and body sensors have been used in the past and are still used for detecting fast and subtle motions. Some vision based systems are also used, which function based on properties like texture and colour. However, several factors are associated with these gesture sensing technologies, like accuracy, cost, comfort, etc. Handling these devices might be cumbersome to the user.

This work essentially deals with detection of gesture, first from a recorded video and then from a live camera feed. First image segmentation techniques like frame differencing and background subtraction have been applied to study the motion occurring from one frame to the other. Morphological operations like dilation and erosion are then applied to filter the segmented image. Then connected component analysis is performed on the resultant image and the person making the gesture is detected.

The work has then been extended to observe and detect motion from a live camera feed. Optical flow is used here for motion detection, since frame differencing has some observable disadvantages like noise detection and inaccuracy. A live feed is taken from an IP PTZ camera and optical flow technique is applied to it. Once the gesture is detected from the feed, we compute the pan, tilt, and zoom required to focus the person making the gesture. Then the camera is fed with computed values of the parameters and the camera focuses on the person, in real time.

## CHAPTER – 1

### INTRODUCTION

#### 1.1 Endeavour of the Work

**Gesture detection** is a topic of computer science with the aim of identifying, detecting or recognizing human gestures by utilising mathematical algorithms. Gestures can originate from any bodily motion or state but, like face, hand, leg, etc. [1]. Immense research has been carried on for years for identification of sign language. Gesture detection has wide field of applications, like an indoor surveillance system, automated homes, remote control through gestures, etc. Gesture detection provides a way of raising the level of interaction between machines and human beings to another level.

While technologies like tracking gloves and body sensors exist for detection of motion and gesture, they are cumbersome to use and have cost and accuracy related issues [2]. This introduces the point of implementing techniques like image segmentation and connected component analysis for detection of person making the gesture. The project is then further extended to detect motion from a live camera feed in real time and detect motion from it using optical flow technique. The pan, tilt, and zoom of the camera are then calculated to direct the camera to focus on that particular person.

## 1.2 Literature Survey

Extensive Literature Survey was carried out before the commencement of the work and during the course of the work, which involved survey of various image processing techniques from Research journals and online materials. The major activities of Literature Survey are described below.

The paper titled "Human Activity Analysis: A Review" by J. K. Aggarwal and M. S. Ryoo, was surveyed to study about various kinds of human activities and the importance of their recognition. The paper gave us a general idea about activity recognition.

"Gesture Recognition: A Survey" by S. Mitra and T. Acharya was studied thoroughly. We studied about the various kinds of gestures and their uses. We had a general idea about the application of gesture recognition in different fields and the different kinds of gesture sensing technologies used nowadays. The paper described how the field of gesture recognition has come into prominence over the years and is a widely researched branch of computer science today.

The book "Digital Image Processing" by R. Gonzalez and R. Woods was referred to study about the basics of image processing, like image morphology, the definition of basics like frames and color scales, etc. The majority of the formulae used for implementation were studied from this book. Topics like image dilation erosion, morphological closing of an image, and general topics like video frame segmentation, RGB and grayscales of color, etc. were studied from this book extensively

The paper "Determining Optical Flow" by Berthold K.P. Horn and Brian G. Schunck, gave us the complete insight about the optical flow. The paper presents detailed description about the calculation of optical flow and the constraints associated with it. We studied how optical flow overcomes the difficulties posed by basic image segmentation techniques like frame differencing and background subtraction. We studied about the various factors that influence the calculation of optical flow and the result that we obtain

from it. Optical flow technique learnt from this paper was implemented for calculation of velocity vectors and for subsequent determination of motion in a video frame.

### **1.3 Objectives of the work**

1. Segmentation of video frames
2. Image Morphology
3. Connected Component analysis
4. Optical flow analysis
5. Controlling camera parameters

## CHAPTER – 2

### GESTURE DETECTION FROM A RECORDED VIDEO

#### 2.1 Image Segmentation

Image segmentation is the process of dividing an image into segments, or set of pixels, so that we can extract useful features from the image. The aim of image segmentation is to make the image easier to analyse and useful for feature extraction. Generally, image segmentation is used for identifying objects from a picture or detecting boundaries and edges [3]. Thus, image segmentation primarily has two objectives: first is to decompose the image in such a way so that we can extract and process only the required features of an image, and second is to change the representation of the image in such a way that is useful for further processing.

The motivation for applying image segmentation techniques to the current problem are due to the fact that motion in a video causes change in properties of the pixels from one frame to another. While the pixels corresponding to the static portion of the video remain unchanged, the pixels corresponding to the moving portions (the object of interest) vary in intensity from frame to frame. This enables us to identify the moving portion separately from the video frame.

##### 2.1.1 Background Subtraction

Background subtraction is the process of separating the foreground objects from the background in a video frame. Since the foreground object is the object of our attention, this method is useful for separating out the moving objects in a video from the still background. The mathematical formula for background subtraction can be given as

$$DIFF[i, j] = I_t[i, j] - I_1[i, j]$$

Where  $DIFF$  represents the difference between the current frame (at time 't') and the background frame.



**Figure 2-1 Background Subtraction**

### **2.1.2 Frame Differencing**

Frame Differencing is the technique to compute the difference between two consecutive video frames. If there is a change in pixels from one frame to the next frame, this surely implies that there is some change occurring in the video between consecutive frames. Thus, this technique is useful for identification of the moving object (or body part) in the video. Mathematically, frame difference can be represented as

$$DIFF[i, j] = I_t[i, j] - I_{t-1}[i, j]$$

Where DIFF represents the difference between the current frame (at time 't') and the previous frame (at time 't-1').



**Figure 2-2 Frame difference**

## 2.2 Image Morphology

Morphological image processing is a collection of non-linear operations concerned with the morphology of features in an image [4]. Morphological operations functions are dependent only on the order of the values of the pixels, not on their numerical values.

Therefore, binary images can easily be processed using these operations. Morphological operations can also be applied to grayscale images.

Morphological techniques operate on an image with a small element called a structuring element. The structuring element is overlapped at all pixel locations of an image and then it is compared with the neighbours of that particular pixels. Then operations are applied to compute whether the element adjusts completely within the neighbourhood, or whether it just intersects it.

A morphological operation when applied on a binary image results in a new binary image in which a pixel's value is non zero only if the operation that has been applied is successful at that location in the input image.

The structuring element is a small binary image, i.e. a set of pixels, each with a value of 0 or 1:

- The matrix dimensions of the element specify the size of the structuring element.
- The pattern of 1s and 0s determine the shape of the structuring element.
- The origin of the structuring element is usually one of its pixels, but the origin can be present outside the element too.

When the structuring element is placed in a binary image, each of its pixels is overlapped with the corresponding pixel of the neighbourhood under the structuring element. The structuring element completely fits the image if, for each of its pixels set to 1, the corresponding image pixel is also 1. Similarly, a structuring element intersects an image if, at least for one of its pixels set to 1 the corresponding image pixel is also 1.

Zero-valued pixels of the structuring element are ignored, i.e. they indicate points where the corresponding image value is irrelevant.

### 2.2.1 Image Dilation

Dilation of a binary image gradually enlarges the boundaries of regions of foreground pixels, resulting in the areas of foreground pixels growing in size while holes within those regions becoming smaller. Thus dilation grows and thicken the foreground objects in an image.

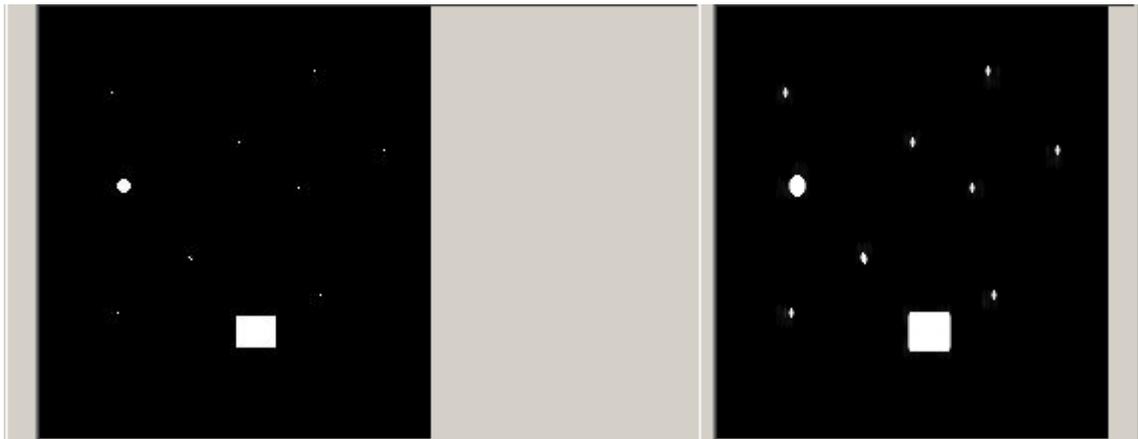


Figure 2-3 Image dilation, image on right showing the dilated image

### 2.2.2 Image Erosion

Erosion of binary image results in the foreground pixels shrinking and holes expanding. Dilation, followed by erosion, enlarges the background, but retains the boundary shape. Erosion thins out the objects in an image.

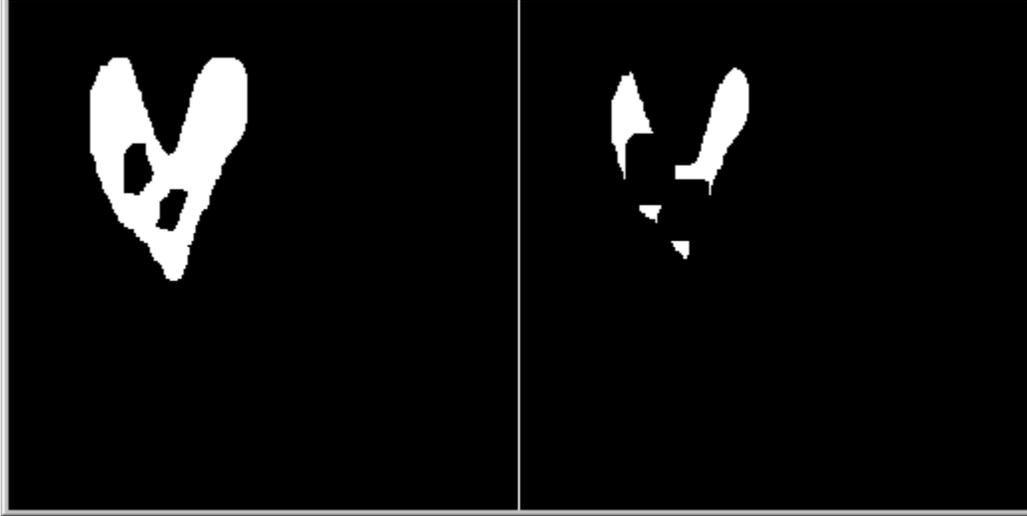


Figure 2-4 Image erosion, image on right showing eroded image

### 2.3 Connected Component Analysis

Connected component labeling functions by scanning an image pixel by pixel (from top to bottom and left to right) in order to identify connected pixel regions, *i.e.* neighborhood regions of pixels which share the same set of intensity values  $V$  [5].

Connected component labeling operates on binary and graylevel images and different measures of connectivity are possible. The connected components labeling operator scans through an image row-wise until it comes to a point  $p$  (where  $p$  is the pixel that is labeled during the scanning process) for which the intensity value is 1. When that point is reached, the four neighbors of  $p$ , top, bottom, left and right pixels, are examined which have already been encountered in the scan earlier. Based on this information, the labeling of  $p$  occurs as follows:

- If all four neighbors of  $p$  are 0, we assign a new label to  $p$ , else
- if only one neighbor has intensity value=1, we assign its label to  $p$ , else
- if more than one of the neighbors have intensity value=1, one of the labels is assigned to  $p$  and the equivalences are noted down.

After the scan is completed, we sort the equivalent label pairs into equivalence classes and we assign a unique label to each class. Finally, a second scan is made through the image, and in this step, each label is replaced by the label assigned to its equivalence classes.

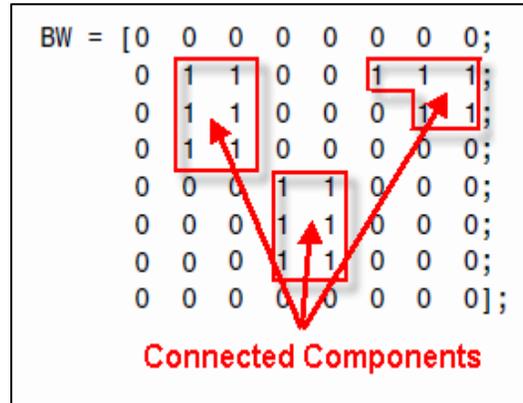


Figure 2-5 Matrix showing labelled components

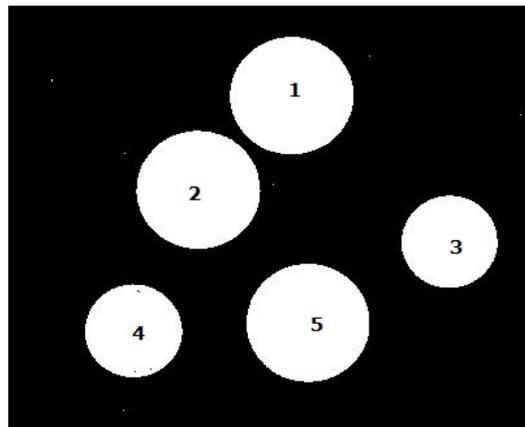


Figure 2-6 Labelled objects in an image

## CHAPTER – 3

### GESTURE DETECTION FROM LIVE CAMERA FEED

#### 3.1 Optical flow analysis

Optical flow or optic flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene. It is the distribution of velocities of brightness patterns in an image [6]. Optical flow arises from the relative motion of objects and the viewers and is useful for detection of moving objects in an image.

Two basic assumptions are made for computation of optical flow:

- Image brightness constancy: The brightness of any observed object point remains constant over time.
- Surface reflectance does not contain highlights.



Figure 3-1 Optical flow analysis showing the velocity vectors

### 3.1.1 Horn Schunck method of optical flow determination

This is a global method, where calculation of a velocity vector can be based on the entire image. The length of the velocity vector gives the magnitude of velocity, and the direction determines the direction of motion. The Horn-Schunck [7] method imposes another constraint apart from the universal image brightness constancy, the smoothness constraint. The smoothness constraint imposes the rule that the neighborhood points in an image should move in similar manner. So, the optical flow field should vary smoothly from one frame to another without much discontinuity. Even if the object changes its position, its reflectivity or illumination is assumed to remain unchanged over time. It tries to minimize the error:

$$E^2 = \iint (\nabla I \cdot \mathbf{v} + I_t) + \alpha^2 \left( \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 \right) dx dy$$

Where the first part is the image brightness constraint and the second part is the image smoothness constraint.

In the first part of the equation,  $\nabla I$  is the spatial gradient,  $\mathbf{v}$  is the optical flow vector and  $I_t$  is the temporal gradient. The first part of the equation essentially tells us that if we apply the optical flow vector to the spatial gradient, it'll be cancelled out by the temporal gradient. This is consistent with the image brightness constancy that we have assumed.

The second part of the equation tells us that the optical flow vectors should have minimal discontinuity. Hence, we try to minimize the square of magnitude of flow vectors along all dimensions.

The above equation, when worked out iteratively, results in the fact that optical flow vectors for any pixel at a moment of time are computed using two components: the average of the flow vector in the previous iteration and the spatial and temporal gradients of the pixel of interest. Since the method is iterative, optical flow vectors are generated for all pixels and all the empty regions will also be filled. The optical flow vectors can then be

drawn as lines, the length of the lines representing the magnitude of the velocity and the orientation giving the direction of motion of the object.

### 3.2 Calculation of Pan, tilt and zoom of IP camera

An IP camera is a digital camera, generally used for recording videos for surveillance purposes. Unlike normal CCTV cameras, IP cameras can receive and transmit information from computer networks. IP cameras have the added advantage that they can be easily interfaced through popular scripting softwares and hence, can be controlled and commanded as per requirement. An IP camera has generally three parameters associated with it: Pan, Tilt and Zoom.



**Figure 3-2 AXIS 214 PTZ camera**

The pan of the camera refers to its rotation of the camera head in the horizontal plane. Tilt determines the movement of the camera in vertical plane, and zoom refers to the distance relative to the camera at which the object of interest is located. An AXIS 214 PTZ Network camera was used for this project, which had the following PTZ value ranges:

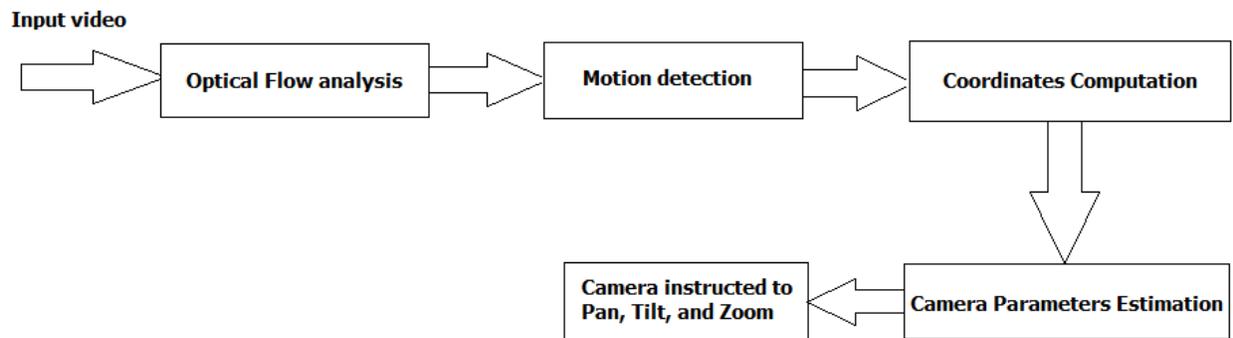
Pan: -170 to 170 degrees

Tilt: -90 to 30 degrees

Zoom: 18x optical zoom

Once the coordinates of the detected motion in a frame are determined by application of optical flow, the coordinates can then be used to compute the degree by which the camera needs to be panned, tilted and zoomed to focus it on the person making the gesture, hence fulfilling the motive of the project.

The block diagram of the complete process of determination of PTZ of the camera can be illustrated below:



**Figure 3-3 Block Diagram of the complete process of PTZ determination**

## CHAPTER - 4

### IMPLEMENTATION

#### 4.1 Algorithm for detection of person making the gesture from a video shot on a static camera

1. Partition the video into separate frames.
2. Convert the frames from RGB scale to grayscale for easier processing
3. Apply background subtraction to every frame to extract the foreground objects from the video in every frame.
4. Apply frame difference to every consecutive frame to recognize the moving object between frames.
5. Apply image dilation, followed by image erosion for every frame to enlarge the background and fill the holes inside the detected foreground objects, all the while retaining the object boundary.
6. Assuming that the group of pixels obtained after background subtraction are non-overlapping and distinct, apply connected component labelling on them and label them as separate labels.
7. Compare the pixel location of the moving object obtained from frame differencing with the labels generated now, and find out the label in which they exist.
8. The label containing the pixels corresponding to the moving object is the person making the gesture.

## **4.2 Algorithm for motion detection from a live camera feed and consequent determination of camera parameters**

1. Obtain an image snap from the IP PTZ camera and convert it to grayscale.
2. Apply Horn Schunck optical flow method to the frame.
3. Determine the velocity vectors for the motion occurring in the frame.
4. Calculate the threshold velocity, the velocity which marks the minimum velocity to be perceived as a gesture.
5. Segment the object of interest from the frame based on the calculated threshold velocity.
6. Calculate the area of every segmented object and draw a bounding box around it.
7. Calculate the center of the bounding box to give the coordinates of the detected motion.
8. Estimate the pan, tilt and zoom of the camera based on the computed coordinates and the range of the camera parameters.
9. Direct the camera to point and focus at that the area where the gesture is being made.

## CHAPTER - 5

### RESULTS

All the results shown below have been generated using MATLAB 2013a software. AXIS 214 PTZ Network camera has been used for recording live camera feed and then focusing it on the person of interest.

#### 5.1 Background Subtraction

The figures below show two frames taken from a video shot on a static camera. The first frame illustrates the background, and the second frame shows the background with people present in it. Their resultant background difference, after thresholding, results in the third frame, which shows only the two people, with background frame subtracted and eliminated.



**Figure 5-1 Background Frame**



**Figure 5-2 Frame at time 't'**



**Figure 5-3 Result from background subtraction**

## 5.2 Frame Differencing

The figures below show two consecutive frames taken from a video and their resultant frame difference.



**Figure 5-4 Frame at time 't'**



**Figure 5-5 Frame at time 't-1'**



**Figure 5-6 Result after frame differencing and thresholding**

### **5.3 Image dilation, followed by erosion**

Dilation, followed by erosion, applied on an unfiltered frame.



**Figure 5-7 Frame without dilation and erosion**



Figure 5-8 Frame after dilation, followed by erosion

#### 5.4 Connected Component Labelling

Connected component labelling applied to label the two people in the video separately and then a match is made between the moving hand and the labels to detect the person making the gesture.



Figure 5-9 Labelled frame

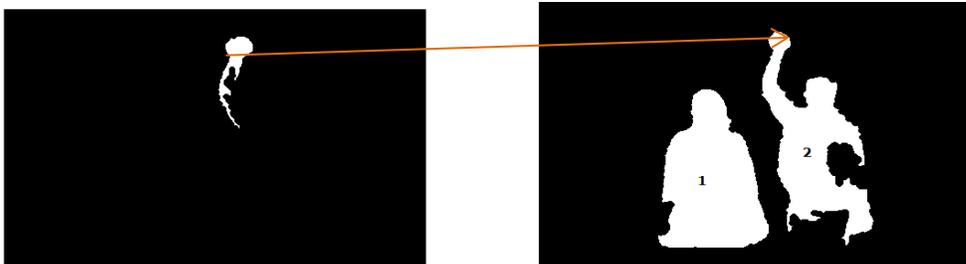


Figure 5-10 Match between hand and label 2

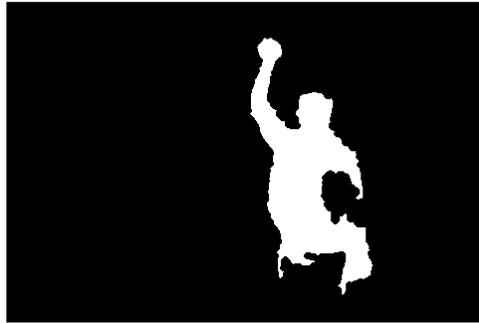


Figure 5-11 Identification of person making the gesture

## 5.5 Optical flow analysis on IP camera feed and motion detection

Optical flow technique applied on a live feed obtained from an IP camera. The bounding box shown indicates the region in which motion was detected.



Figure 5-12 Motion detected

## **CHAPTER – 6**

### **CONCLUSION AND SCOPE FOR FURTHER RESEARCH**

Several techniques were involved during the process of motion detection and gesture recognition. Techniques like frame differencing and background subtraction were used for segmenting the image and separating out the foreground objects and the moving objects. Morphological operations like dilation and erosion were applied to fill the holes in the binary representation of frames and to have distinct boundaries. Finally, connected component analysis was used to identify the person making the gesture.

The disadvantage associated with frame differencing is that it detects even slightest of motions in the video and doesn't recognize the direction of motion. Hence, Horn Schunck method of optical flow was used for motion detection.

The procedure was then applied to a real time camera feed which supports PTZ. Motion was detected and the camera was directed to point and focus at that particular person.

This work can be further extended for real time monitoring of a room, or for surveillance purposes. The work can also be used for classroom monitoring. Using better techniques for estimation of pan, tilt and zoom, we can get even more accurate results.

## REFERENCES

- [1] J. K. Aggarwal and M. S. Ryoo, "Human Activity Analysis: A Review," *ACM Computing Survey*, 43 (3), 2011.
- [2] S. Mitra and T. Acharya, "Gesture Recognition: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics – Part C*, 37 (3), 2007.
- [3] R. Gonzalez and R. Woods, "Digital Image Processing", Addison Wesley, 1992.
- [4] <https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing.html/topic4.htm>
- [5] <http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm>
- [6] Berthold K.P. Horn and Brian G. Schunck, "Determining Optical Flow", *Artificial Intelligence*, vol 17, pp 185–203, 1981.
- [7] Peter O'Donovan, "Optical Flow: Techniques and applications", *Artificial Intelligence*, 2005.
- [8] [http://www.borisfx.com/avid/bccavx/classic\\_features.php](http://www.borisfx.com/avid/bccavx/classic_features.php)
- [9] <http://areshmatlab.blogspot.in/2010/05/low-complexity-background-subtraction.html>
- [10] <http://vip.bu.edu/projects/vsns/background-subtraction/>
- [11] <http://www.mathworks.in/help/images/labeling-and-measuring-objects-in-a-binary-image.html>
- [12] <http://www.codeproject.com/KB/GDI-plus/ImageDilation/DilationProject.PNG>
- [13] <http://www.cis.rit.edu/class/simg782.old/talkmorphimages/Erosion1.gif>