Detection of Defined Human Poses for Video Surveillance

Mid-year Report for BTech 451

by

Xu He

Supervisor: Professor Reinhard Klette

Tamaki campus
Department of Computer Science
The University of Auckland
New Zealand
June 2014

Abstract

The project aims at motion detection for video surveillance by using an IPVS camera offered by Compucon New Zealand. Commercial off-the-shelf video surveillance cameras are very capable of recording images in computer hard disks at high resolution and high frame rates. They are also capable of applying motion detection criteria to achieve surveillance objectives while reducing the bandwidth burden for data transmission and recording in computer storage. However, the detection criteria are often still just based on changes in pixel counts and not on the meaning of images or the environment seen or recorded. This is not accurate and leads to a lot of false alarms. Computer data modelling in motion detection comes in at this point. Motion detection is typically used for real-time human detection, human tracking and human activity analysis in video surveillance systems.

This is a one-year project, offered by Compucon New Zealand. It carries weights of three courses taught at the University of Auckland. This project is compulsory for a final-year BTech (Information Technology) honors degree student.

Keywords: Human motion detection, Video surveillance, Human tracking, Background subtraction, Raising-up-hands recognition.

Acknowledgments

I thank my academic supervisor Professor Reinhard Klette for guiding and teaching. I am grateful to Dr Manoharan for giving me the opportunity. I also thank TN Chan for keep pushing me, and for providing a camera. Last not least, I thank Zhengping Wang for providing his code and helping me with silhouette detection.

Xu He
Auckland
August 15, 2014

Contents

Chapter 1

Introduction

This is a mid-year report for a BTech 451 project at the University of Auckland; the final version will be presented in November 2014. This is a one-year project including two semesters, 8-10 hours of work for the first semester, and 16-20 hours of work for the second semester. The project is research oriented on the methodologies of foreground motion and human action analysis for a real-time video surveillance system.

In this chapter, I introduce the project content, the company information and the camera ACM-1511. Then I report about the motivation and goals in this project. At last, I list the structure of the report.

1.1 Project Overview

Currently, video surveillance systems are applied very common in public places. The main purpose to use a video surveillance system is for security and recording. Many commercial and public places rely on security people for watching the recording. It is not efficient, so automatic security analysis is developing in recent years.

Traditional video surveillance system records all the information but it will consume a huge amount of storage space. Modern commercial video surveillance system are very capable of recording video when human motions are being detected, that is in order to reducing the bandwidth of data transmission and video storage.

However the detection are based on changing of pixels, not on the meaning of the video such as people behavior abnormal in front of camera. The detection results are not accurate and leads to lots of false alarms. At this point, we attempt to modeling a real-time system to take the meaningful defined human motions captured by a surveillance camera.

1.2 Company Information

This project is sponsored by TN Chan from the company Compucon New Zealand.

Figure 1.1: Compucon New Zealand

Compucon New Zealand is a computing system manufacturer and a digital technology system integrator, which was established in 1992, as shown in figure 1.1. It has been 100% New Zealand owned since 2011.

Compucon New Zealand is also part of an International Compute manufacturing group of companies founded in 1989 in Sydney. For more information about this company, please visit following link,

www.compucon.co.nz.

1.3 Data Collection

An IPVS camera will be used for recording sample data, the model is ACM-1511, shows in figure 1.2. This is a powerful IP indoor camera, which supports 8 frames per second at 1280 x 1024 resolution, or 30 frames per second at 640 x 480 resolution. We record sample video on a multi-media lab in the University of Auckland.

For more information about this camera, please visit following link,

http://www.acti.com/product/detail/Bullet_Camera/ACM-1511.

Figure 1.2: ACM-1511

1.4 Motivation and Goal

This project will focus on two main area, foreground moving object detection and analyzing defined human action.

Recognizing of people holding up hands is the main goal on this project. At the end of the project, I hope to gain a good understanding of several computer vision techniques including foreground detection, human action analyzing, morphology processing, and how does those knowledge apply to real devices. Also, have a good experience on how to doing an academic research on computer science.

1.5 Structure of Report

The next chapter will take about how we recorded a test video sequence by using a IPVS video surveillance camera. In order to supporting the future work, we will discuss how many situations and problems probably exist in this project.

In chapter 3, we will research and compare the several exist algorithm for silhouettes detection, then we choose one of the appropriate method. After implement the algorithm, we will show some foreground results.

In chapter 4, we will research on a set of method for refining detected silhouettes. The silhouettes results will contain noises, shadows and holes, which are affect hu-

man modeling.

In chapter 5, we will discuss methods of pose understanding, we use detected silhouettes to matching a human model, then provide a bunch of classifier to recognize people raising up his hands.

At last, we will discuss the results and make a conclusion.

Chapter 2

Silhouette Detection

This chapter discuss several method for moving object detection in order to get a human silhouette. The first subsection research on several famous methods. Then, we choose one suitable method for this project.

2.1 Overview about Methods

Currently, moving object detection has been researched over the past years but it is still a challenge problem. Several methods can be used for real time moving human detection of a surveillance system. However, there is no perfect algorithm for those purpose, these methods have their advantages and disadvantages. I briefly review methods as listed in surveys as follows,

The **Optical flow** is a very important method for moving object detection and analyzing. The definition of optical flow is defined by Gibso in 1950. This book [1] discusses the method step by step.

The **Background subtraction** approach is one of the basic techniques used for detection moving foreground objects (e.g. a human). It is widely used for video surveillance and not computational expensive. A non-moving camera is very suitable of using background subtraction approach, especially in our case. In another words, if the camera are moving, the whole processing procedure will fail due to the changing of almost every pixels. This paper [2] has proposed a comparison and description of real-time background subtraction algorithms for a video surveillance system.

All the background subtraction method have similar processing steps. The main processing steps is training a set of video sequence in order to get a background image, then set a threshold to subtract current frame between trained background frame. Also the background image will keeping update as time goes.

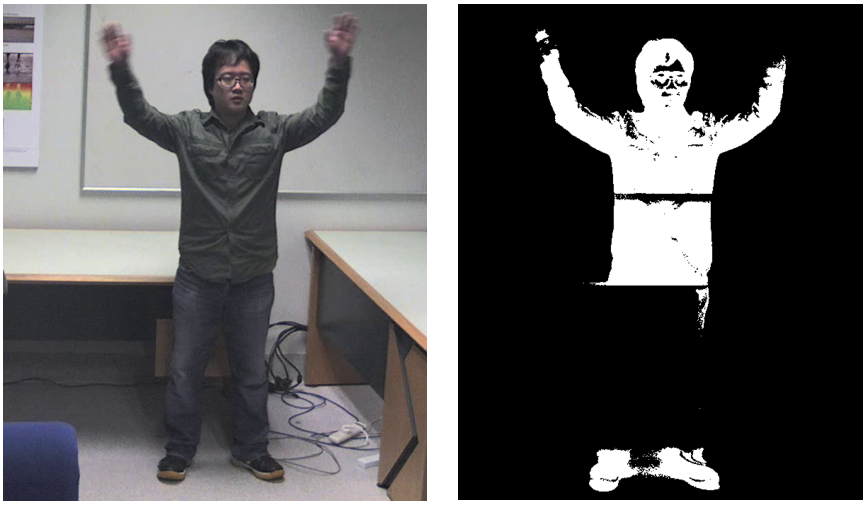

Figure 2.1: A sample using Gaussian mixture model.

To subtraction background, we need to produce an background image first. There are several background modelling method has been presented.

(1) The *Frame differencing* algorithm is a simple and basic method for background subtraction by checking the difference in a set of consecutive frames. We considered the pixel as a foreground if the corresponding pixel have changed apparently by comparing threshold. Frame differencing is very easily to implement and use, but not very suitable for detecting slow moving object. Also, it is hardly to determine a good threshold value for any particular environment and the calculation step is limited by threshold which leads to inaccuracy results. This paper gives a solution using frame differencing [3].

(2) The *Gaussian mixture model* (or a mixtures of Gaussian) is defined by using a small number (say, between 3 to 5) Gaussian distributions for an additive description of background values.

There is a build-in method from OpenCV, BackgroundSubtractorMOG, which is implements Gaussian Mixture-based background and foreground segmentation algorithm described in [4]. This paper proposed a method to improve GMM mode by using a automatic method for adaptive lightning changing. An example by using OpenCV function, see figure 2.1.

(3) The *Median filtering* is a statistical background modeling method widely used in research area. Each background pixel is the median value of each corresponding

pixel in all buffered frames. The background is defined as following equation,

$$B_t = Median(I_t, I_{t-1}, I_{t-2}, ..., I_{t-n}) \tag{2.1}$$

Where I_t is the frame at time t, B_t is the updated background frame. An n value is being used for decide how many n frames buffered to calculate background, thus the larger frame number we stored, the more frames we calculate. The *median filtering* has complexity $O(NlogN)$.

(4) The *approximate median filtering* introduced by McFarlane and Schofield [5], which is a complimentary statistical background modeling method to *median filtering*.

The *approximate median filtering* background pixel is updated by the following equation (2.2).

$$B(x,y,t) = \begin{cases} B(x,y,t-1) + 1, \\ \qquad \text{if } I(x,y,t) > B(x,y,t-1) \\ B(x,y,t-1) - 1, \\ \qquad \text{if } I(x,y,t) < B(x,y,t-1) \end{cases} \tag{2.2}$$

Where $I(x,y,t)$ is the mapping of image's pixel at position (x,y) in time t, and $B(x,y,t)$ is the mapping of background's pixel at position (x,y) in time t. Let $I(x,y,0)$ (in first frame) be the initial value of $B(x,y,0)$. Then, in each time frame, we updated the background pixels by comparing with the previous background pixels.

2.2 The Chosen Method

We use the *approximate median filter* to update background image. And we use the source code provided by Zhengping Wang, a reference of his paper in [6].

First, we training a background image by using *approximate median filter* as mentioned above. Then, estimate the background edges on the subtracted background image and raw occlusion boundaries of a person by using the Sobel operator (will discuss in section 3.2). Then, subtract the raw occlusion boundaries of a person and background boundaries in order to extract the true occlusion boundaries of a person. Finally, we can fill the true occlusion boundaries to get the foreground mask.

We use the following equation (2.3) to subtract out the foreground.

$$F(x,y,t) = \begin{cases} 1 \;\; if|I(x,y,t) - B(x,y,t-1)| > \sigma_t \\ \\ 0 \;\; I(x,y,t) < otherwise \end{cases} \tag{2.3}$$

Where $F(x,y,t)$ is the foreground pixel at position (x,y) in time t, and the initial value of $F(x,y,0)$ is 0, so we considered all the pixels as background pixels at the beginning. A pixel at (x,y) int time frame t is a foreground pixel when the absolute difference between the current value of $I(x,y,t)$ and background value $B(x,y,t-1)$ is larger than a threshold $sigma_t$, which is defined as following equation (2.4).

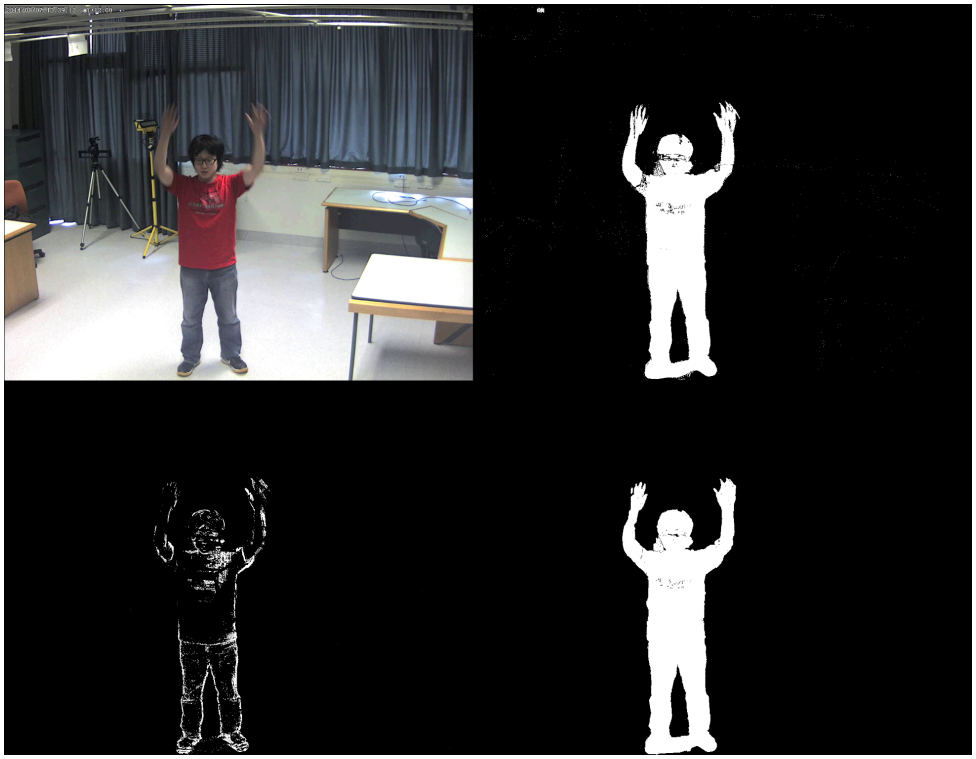

Figure 2.2: A screen shot of front silhouette. *top-left:* current frame, *top-right:* foreground detection result, *bottom-left:* occlusion boundary, *bottom-right:* result silhouette

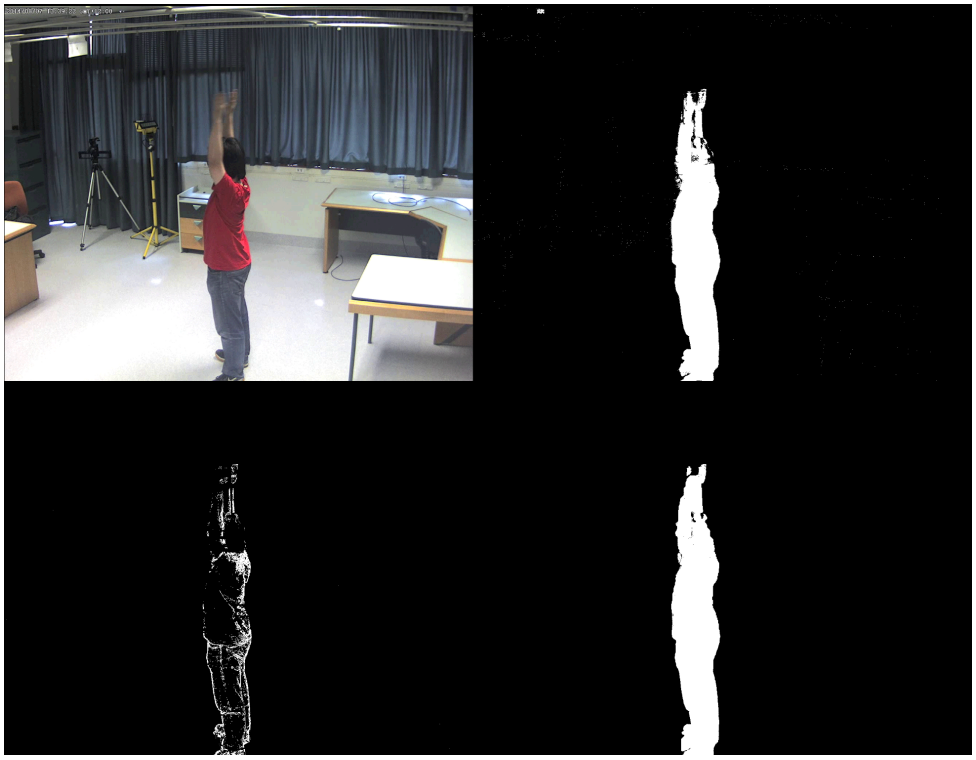

Figure 2.3: A screen shot of side silhouette of a person. *top-left:* current frame, *top-right:* foreground detection result, *bottom-left:* occlusion boundary, *bottom-right:* result silhouette.

$$\sigma_t = \sqrt{\left(\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}(I(x,y,t)-\mu_t)^2\right)/n} \tag{2.4}$$

$$\sigma_t = \left(\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}I(x,y,t)\right)/n \tag{2.5}$$

Where the σ_t is the standard deviation of all input pixels at time frame t, and the μ_t is the mean of all input pixels at time frame t. W and H are width and height of the frame respectively and n is the total number of pixels in the frame.

Figure 2.2 shows a front human body silhouette result by using *approximate median filter* of Zhengping's method, and figure 2.3 2.4 shows two different sides hu-

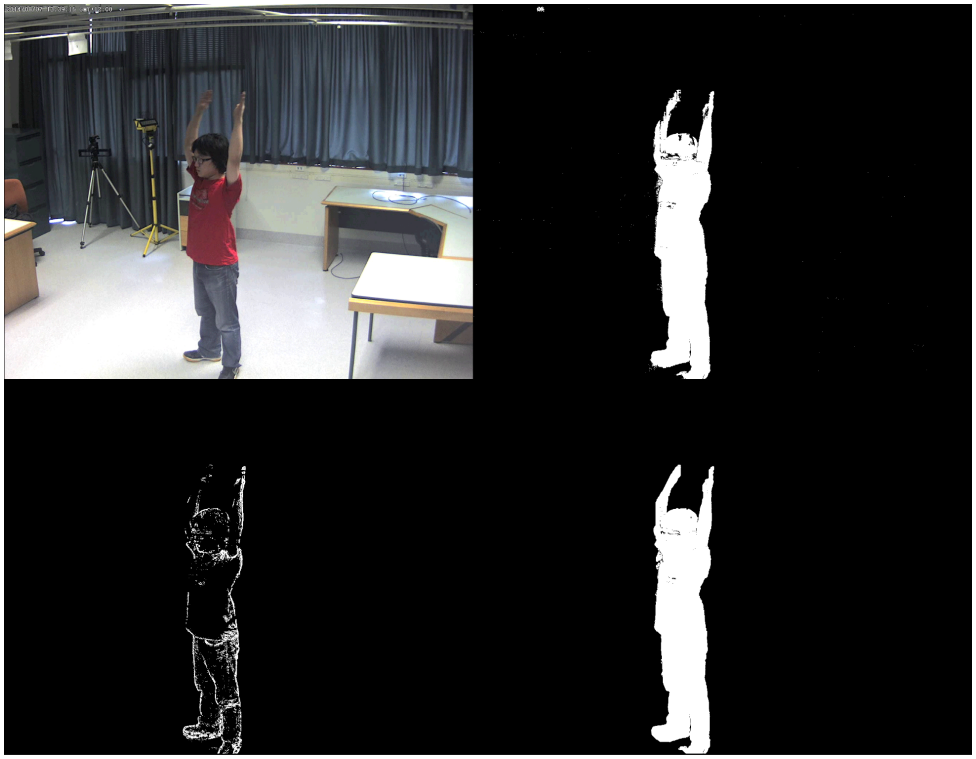

Figure 2.4: A screen shot of side silhouette of a person. *top-left:* current frame, *top-right:* foreground detection result, *bottom-left:* occlusion boundary, *bottom-right:* result silhouette.

man body silhouette.

In conclusion, these figures shown a good results of a human body silhouettes. However, the silhouettes have some little holes and shadows. We will discuss how to shadow removal, holes filling, edges detection and noise removal in the next chapter.

Chapter 3

Refining Detected Silhouettes

In this chapter, we will discuss the methods of refining detected human body silhouettes including shadow removal, holes filling and noise removal. After these process, we will get a meaningful results for future motion recognizing. The silhouettes results influenced by the lightning condition, room environment, background objects and camera position.

3.1 Shadow Removal

Shadow removal is an major problem in video surveillance. Moving shadow can also be detected by the background subtraction algorithm, but we do not need shadow pixel which will affect the further human detection analyzing.

Removing shadow normally can be done in a color image rather than a gray-scaled image, detect shadows in gray-scaled image is more complicated and challenge. An fast and robust shadow removal algorithm based on YUV color space has been reported in these paper [7]. An cast shadow method deal with Gaussian Mixture Models' drawback proposed in [8]. An approach based on RGB color space and K-means clustering algorithm [9].

In this project, we used an new shadow evaluation method proposed by Zhengping Wang in [6]. There are two main steps, candidate shadow detection and shadow evaluation.

Candidate shadow can be detected by forming an Gaussian distribution of detected foreground pixels. Real foreground pixels will close to the top of Gaussian curve, whereas shadow pixels are concentrate on each sides of Gaussian curve.

The candidate shadow can be detected as following equation 3.1,

$$S(x,y,t) = \begin{cases} 1 & |I(x,y,t) - \mu| > \sigma \\ \\ 0 & otherwise \end{cases} \tag{3.1}$$

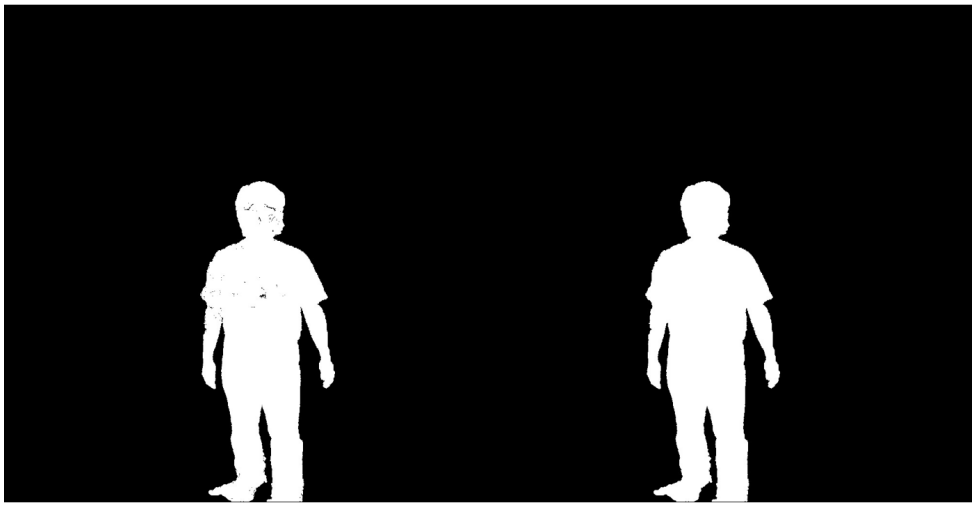

Figure 3.1: left: silhouettes with holes. right: filled silhouettes

Where, $S(x, y, t)$ is candidate shadow, $I(x, y, t)$ is current frame, $B(x, y, t)$ is background image, μ is the mean value of all foreground pixels' intensity, and an standard deviation σ is being used for reduce incorrect detection of shadow pixels.

Shadow evaluation step is used for correcting misclassified pixels. The paper defined a true shadow pixel should not be a pixel which is enclosed or semi-enclosed by pixels on an detected occlusion boundary of a person. Figure 3.2 shows an silhouettes after shadow removal.

3.2 Holes Filling

In the extracted foreground silhouettes, there are a lot of holes shown in the body. Figure 3.1 shows a not saturated foreground silhouettes, which will leads the foreground object is not faithful to the original input frame. The reason is the the intensity value between background and foreground pixels are quite similar.

To fix this problem, we can use a morphology dilation operator to fill small holes in an binary image. However, morphology dilation depends on kernel size, large kernel size is not computational cheap, small kernel size may not deal with large holes.

Here we use another simple method by using an OpenCV function, *cvDrawContours* with parameter CV_FILLED. The more specific steps as follow, we define a hole

as a background region surrounded by a connected foreground region. First, perform edge detector on an image with holes. Then fill the pixels outside the contours, Finally, we can get an filled image by inverse the filled image before.

3.3 Noise Removal

Mathematical morphological erosion operation is a common tool in image processing for noise removal. This is works well for the very small blob, but it can not handle large blob.

The large non human blob can be removed by following algorithm. Count the total number of every blob border pixels. Then set a threshold, if the total number less than threshold, we removed that blob from the frame.

An result image shows in figure 3.2.

3.4 Edge Detection

In this project, edge detection is being used for silhouette detection we mentioned in chapter 2. And also being used for extract contour of the silhouettes.

Edge Detection. Methods for edge detection follow the step-edge model. This means that we can detect an edge in an image by local maximal of absolute values of first-order derivatives or by zero-crossings in second-order derivatives.

Figure 3.2: left: silhouettes with noise and shadow. right: silhouettes after shadow and noise removal

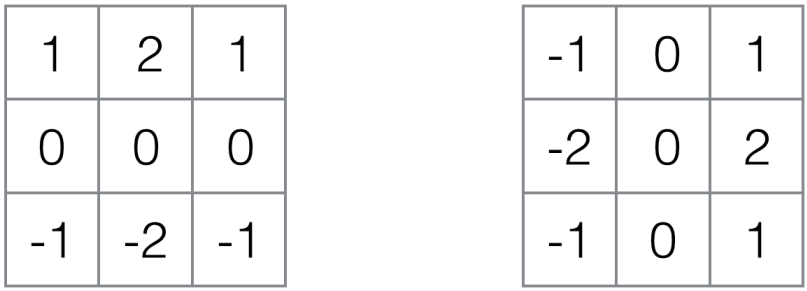

Figure 3.3: Two Sobel operators left: horizontally operator right: vertically operator

The *Sobel operator* is a simple example of a first-order derivative-type edge detector.

Sobel operator contains two 3 x 3 kernel, in figure 3.3, can be separately calculate the convolution between input image and one of the operator. Thus, $G_(x)$ is the horizontal Sobel edge convolution result, and $G_(y)$ is the vertical Sobel edge convolution result. Then the edge map can be calculate as following equation 3.2,

$$E(x,y,t) = \begin{cases} 1 \;\; if|G_(x)(x,y,t)| + |G_(y)(x,y,t)| > \alpha\sigma_(t)/2 \\ \\ 0 \;\; otherwise \end{cases} \tag{3.2}$$

Where $E(x,y,t)$ is the edge map, α is a low threshold in order to extract as many background edges as possible. So, true occlusion-boundaries can be detected by subtracting background boundaries and frame occlusion boundaries.

Chapter 4

Pose Understanding

This chapter discuss methods of human modelling based on human body silhouettes.

4.1 Matching a Human Model

Star skeletonization proposed in [10] and used for analysis human walking and human running, which is an silhouette-based method for analyzing human motion or human behaviors. Advantage of this method is, it is not iterative and computationally cheap, which means very suitable for real-time processing. To estimate the human skeleton, the algorithm extracts the five crucial points includes head, left hand, right hand, left foot and right foot, this method described in [12].

The algorithm following these steps,

1. Estimate the boundary of extracted human silhouettes by using an OpenCV function *findContours*.

2. Find the center of gravity of the target boundary. Support there are number of N boundary pixels, and the centroid of the boundary is (x_c, y_c). We defined the centroid as following,

$$x_c = \frac{1}{N}\sum_{i=1}^{N} x_i, y_c = \frac{1}{N}\sum_{i=1}^{N} y_i$$

Where x_c is the average position of sum of all x boundary pixel, y_c is the average position of sum of all y boundary pixel. (x_i, y_i) is the position of each boundary pixel.

3. In this step, we have to find a signal graph which is plotted position i versus distances $D(i)$. Find the distances $D(i)$ between the center of the gravity (x_c, y_c) and each border pixel point. The order of calculating each pixel should be either clockwise or anti-clockwise.

4. The signal $D(i)$ we got from step 3 is noised and zigzag. The distance function can be smoothed by using a low pass filter in the frequency domain.

5. Taken all local maxima as extremal points. The star skeleton is constructed by connecting center of gravity and maxima points.

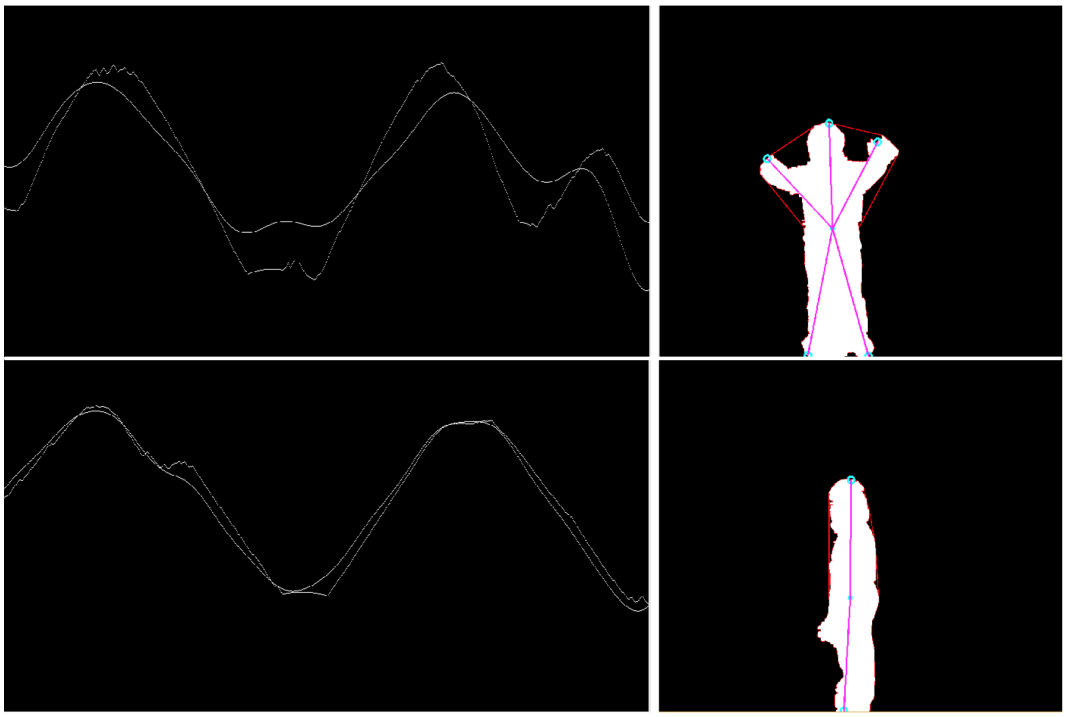

Figure 4.1: A screen shot of two skeleton results. *top-left:* front silhouettes distance signal and smoothed distance signal, *top-right:* front skeleton of human body, *bottom-left:* side silhouettes distance signal and smoothed distance signal, *bottom-right:* side skeleton of human body

4.2 Understanding Poses

For pose understanding, we use the human modelling results, by comparing the extremal points. An classifier needs to be construct to recognize human raising up hands towards different direction. This is the main working for the next semester.

Chapter 5

Conclusions

This chapter describes the project work done so far and planned future work.

During the first semester, I have done the extraction of human body silhouettes from a few test sample videos by using a background subtraction method.

I have done some programming work to refine the detected human silhouettes by a few following steps, such as shadow removing, hole filling, and noise removal. Then I used the extracted human silhouettes for modelling a human body skeleton for future pose understanding.

Currently, the silhouette detection results still have some problems due to lightning changes and the camera's white balance function.

In the next semester I will focus on recognising defined poses, especially people raising up their hands, in several different situations. I will design and implement a classifier to recognise the defined poses, and test the method on different persons. Finally I will also test the method on multiple-people situations as well.

Bibliography

[1] R. Klette: *Concise Computer Vision*. Springer, London, 2014.

[2] M. Hedayati, Wan Mimi Diyana Wan Zaki, Aini Hussain: A qualitative and quantitative comparison of real-time background subtraction algorithms for video surveillance applications. In *Journal of Computational Information Systems*, **8(2)**:493-505, 2012.

[3] N. Prabhakar, V. Vaithiyanathan, A. P. Sharma, A. Singh, and P. Singhal: Object tracking using frame differencing and template matching. Technical Report, Department of Computer Science and Engineering National Institute of Technology Rourkela, Odisha, India, 2012.

[4] P. KaewTraKulPong and R. Bowden: An improved adaptive background mixture model for real-time tracking with shadow detection. In *Video-Based Surveillance Systems*, pp. 135–144, Springer, 2002.

[5] N. J. B. McFarlane, C. P. Schofield: Segmentation and tracking of piglets in images. In *Machine vision and application*, 187-193, 1995.

[6] Z. P. Wang, B. S. Shin, and R. Klette. Accurate silhouette extraction of a person in video data by shadow evaluation. In *Int. J. Computer Theory Engineering*, **6**:476–483, 2014.

[7] O. Schreer, I. Feldmann, U. Golz, and P. Kauff: Fast and robust shadow detection in video conference applications. In *Video/Image Processing and Multimedia Communications 4th EURASIP-IEEE Region 8 International Symposium on VIPromCom*, pp. 371–375, 2002.

[8] B. E. Lee, T. B. Nguyen, and S. T. Chung: An efficient cast shadow removal for motion segmentation. In *Signal processing, computational geometry and artificial vision*, pp. 83–87 , WSEAS, 2009.

[9] A. Chowdhury, U. Chong: Real time shadow removal with k-means clustering and RGB color model. In *International Journal of Multimedia & Ubiquitous Engineering*, pp. 159, SERSS, 2012.

[10] H. Fujiyoshi, A. J. Lipton: Real-time human motion analysis by image skeletoniztion. In *In Applications of Computer Vision Proceedings*, pp. 15–21, IEEE, 1998.

[11] I. Haritaoglu, D. Harwood, and L. S. Davis: Real-time surveillance of people and their activities. In *Pattern Analysis and Machine Intelligence*, pp. 809-830, IEEE Computer Society, 2000.

[12] P. Correa, J. Czyz, T. Umeda, F. Marques, X. Marichal, B. Macq: Silhouette-based probabilistic 2D human motion estimation for real-time applications. In *IEEE International Conference on Image Processing*, pp. 836–839, 2005.

[13] X. Chen, Z. He, D. Anderson, and J. Keller, and M. Skubic: Adaptive silhouette extraction and human tracking in dynamic environments. In *Fuzzy System*, pp.236–243, IEEE Computer Society, 2006.

[14] A. Manzanera: Human motion analysis: tols, models, algorithms and applications. Tutorial presented in ENSTA-ParisTech, Aug–5–2009

[15] M. Dahmane, J. Menuier: Real-time moving object detection and shadow removing in video surveillance. In *International Conference: Sciences of Electronic, Technologies of Information and Telecommunications*, 2005.

[16] H. Kim, R. Sakamoto, I. Kitahara, T. Toriyama, and K. Kogure: Robust silhouette extraction technique using background subtraction. In *MIRU* (Hiroshima, Japan), 2007.