國立交通大學

資訊科學與工程研究所

碩 士 論 文

在影帶中基於粒子濾波器的多重部分人體追蹤

Particle Filter-based Multi-part Human Tracking in Video Sequences

研 究 生：李金翰

指導教授：李錫堅 教授

中 華 民 國 九 十 五 年 十 月

在影帶中基於粒子濾波器的多重部分人體追蹤

Particle Filter-based Multi-part Human Tracking in Video Sequences

研 究 生：李金翰　　Student：Jin-Han Li

指導教授：李錫堅　　Advisor：Hsi-Jian Lee

國 立 交 通 大 學

資 訊 科 學 與 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

September 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年十月

在影帶中基於粒子濾波器的多重部分人體追踪

學生：李金翰　　　　　　　　　　指導教授：李錫堅博士

國立交通大學資訊工程研究所碩士班

摘　　要

本論文之研究目的在於如何利用現有的技術再加強去追踪人體。在一個監控環境中，我們希望建立一個機制去追踪影帶中的人體進而利用相對資訊來做應用。隨著時代的進步，使得視訊訊號數位化處理更為普及化，因此使得智慧型視訊監控系統(VASM)更為矚目。也因為此系統更能切合大眾的需求，因此智慧型監控系統在居家安全控管上扮演很重要的角色。

以往在傳統上為偵測入侵者必須要人工以手動的方式一一監看監視系統，因此我們希望可以配合影像處理以及物體追踪技術，以自動化的方式提供智慧型的監控系統，對居家安全有更一層的保障。

在提出的系統中，分成四個部分：偵測出影像中的移動物體，對前景物體分割成三區塊並同時分別以三區塊做追蹤的動作，利用三區塊的資訊去判斷異常，利用異常的資訊調整區塊 ：

第一個部分，影帶中前景人物的偵測。由於我們的系統目標是希望能夠更加有效的進行人體追蹤，故在前景偵測的部分我們已假設在追蹤過程前都已經明確知道，如果便是配合我們前景偵測的技術去找出影帶中的移動人體，進而利用偵測出來後的資訊我們開始做追蹤，因為一些基本背景偵測的問題我們都已假設完整被解決了，例如影子問題或是前景區域破碎的問題都已經解決了，另外我們的

系統是以追蹤單人區塊為主，故不去考慮多人追蹤之間的互動，每個人體都是以一個個體獨立去做追蹤的。

第二個部份，人物的身體部位切割。當抽取出移動人物後，為了更加有效進行追蹤人體，我們將人做三區塊的切割，依照人體的比例大致上的切割出我們想要的位置，根據身體部位的不同，因此我們將人物分為三個主要部位。這一部份描述我們為何將人物切成三個身體部位，以及如何分別對此三區塊同時做追蹤。

第三個部份，異常偵測。當我們將人體切割成三區塊之後，我們會分別對此三區塊做區塊的追蹤，那這一部份主要是介紹如何利用這三區塊彼此之間的資訊定義出一自定的限制，以便在測量物體時可以利用此一限制來判斷目前是那塊區塊發生異常。

第四個部份，區塊調整。在第三個部分的步驟結束之後，我們可以得知目標是那一區塊發生異常，因此我們判斷出來要將那一區塊做調整的動作，也就是重新的將異常區塊根據一些簡單的規則以及人體中大致的比例將異常區塊重新拉回相對位置，以及重新做追蹤的動作，這個部分就是討論將區塊調整的規則。

實驗的部份，我們測試了許多影帶並運用我們的追蹤機制。實驗結果發現我們所提出來的機制有著不錯的效果，我們可以利用這個機制來協助來應用在智慧型視訊監控系統(VASM)以及入侵者偵測的運用上。

Particle Filter-based Multi-part Human Tracking in Video Sequences

Student：Jin-Han Li　　　　　Advisors：Dr. Hsi-Jian Lee

Department of Computer Science and Information Engineering
National Chiao Tung University

ABSTRACT

The purpose of this thesis is to construct a specific person tracking system when given a image with any person. In the surveillance environment, with the progress of the times to process the video signals is more popular .So we think that the VASM is great and satisfies the need of the popular. In tradition we need watch the surveillance system in order to detect the invader non-automatically. So we combine the technology of the image process and the technology of tracking target to support the automatic machine for surveillance automatically

In our system, we have four part: detection of moving target in video sequences, decomposing the human body into three block and track each block , abnormal detection, state correction.

In the first stage, human detection, we use the foreground detection procedure to detect moving persons in a video. In the foreground detection we assume have solve the shadow problem and while tracking we have know the position of the target we track. In addition, our system is based on the tracking of only one part so we don't consider the interaction between different blocks, each body is tracked independently.

In the second stage, human body decomposition, after detection and separation

of the moving suspect in a video sequence, we need to cur the human body into three blocks. Since the weights for matching the body parts may differ, we decompose the human body into three parts. Here, we propose how to cut the human body with three parts.

In the third stage, abnormal detection, after cutting the human body into three part we will track the block each so this section to introduce how to define the constraint of the information of three parts. Then when we test some samples, we can detect the abnormal of any block.

In the forth stage, state correction, after step three, we will know which part occurs abnormal, so we determine which block need to adjust. In the other words, we use some simply rule to adjust the relative position of abnormal block, and track the new position continuously.

In the experiments, we test cases of human tracking , body part segmentation and abnormal detection and state correction. The experimental results show that our system is very effective for human tracking.

Acknowledgements

I am in hearty appreciation of patient discussion and proper guidance received from my advisor, Dr. Hsi-Jian Lee, not only in the course of this thesis study, but also in every aspect of my personal growth.

I think to Mr. Shen-Zheng Wang and Shan-Lung Zhao for their suggestions and encouragement. Appreciation is given to the colleagues of the Document Processing and Character Recognition Laboratory at National Chiao Tung University for their assistance on this thesis.

Finally, I would like to express my deep gratitude to my family and my friends for their help and encouragement in those working days. I would like to dedicate my dissertation to them.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 Motivation

The purpose of this thesis is to construct a specific person tracking system when given a image with any person. In the surveillance environment, with the progress of the times to process the video signals is more popular .So we think that the VASM is great and satisfies the need of the popular. In tradition we need watch the surveillance system in order to detect the invader non-automatically. So we combine the technology of the image process and the technology of tracking target to support the automatic machine for surveillance automatically

1.2 Problem Definition

Track human beings precisely and correct missing targets under complex background.

1.2.1 Why do we need to track an object?

Tracking can be applied to visual surveillance for invader detection.

1.2.2 Why tracking is difficult?

(1). Motion variation

(2). Occlusion

(3). A non-rigid object

1.2.3 A human is a connected object. Track several body parts simultaneously.

Advantages: Use spatial relationship to correct possible tracking human parts.

Disadvantages: complicate

1.3 Survey of Related Research

1.3.1 Moving Object Detection

Background subtraction is a popular method for foreground segmentation, especially under those situations with a relatively stationary background. It attempts to detect moving regions in an image by differencing between the current image and a reference background image in a pixel-by-pixel manner. However, it is extremely sensitive to changes of dynamic scenes due to lighting and extraneous events. Yang and Levine [1] proposed an algorithm to construct the background primal sketch by taking the median value of the pixel color over a series of images based on the observation that the median value was more robust than the mean value. The median value, as well as a threshold value determined using a histogram-based procedure based on the least median squares method, was used to create the difference image. This algorithm proposed by Yang and Levine could handle some of the inconsistencies due to lighting changes, noise, and so on.

Some statistical methods to extract change regions from the background are inspired by the basic background subtraction methods described above. The statistical approaches use the characteristics of individual pixels or groups of pixels to construct more advanced background models. The statistics of the backgrounds can be updated dynamically during processing. Each pixel in the current image can be classified into foreground or background by comparing the statistics of the current background model. Stauffer and Grimson [2] presented an adaptive background mixture model for real-time tracking. In their work, they modeled each pixel as a mixture of Gaussians and used an online approximation to update it. The Gaussian distributions of the adaptive mixture models were evaluated to determine the pixels most likely from a background process, which resulted in a reliable, real-time outdoor tracker to deal with lighting changes and clutter.

Elgammal, et al. [3] present a non-parametric background model and a background subtraction approach. The background model can handle situations where the background of the scene is cluttered and not completely stationary but contains small motions such as tree branches and bushes. The model estimates the probability of observing pixel intensity values based on a sample of intensity values for each pixel. It could adapt quickly to changes in the scene which enables very sensitive detection of moving targets. The implementation of the model runs in real-time for both gray level and color imagery. Evaluation shows that this approach achieves very sensitive detection with very low false alarm rates.

1.3.2 Object Tracking

Object tracking plays an important part for the surveillance system. Many research studies had been proposed for tracking.

Tracking over time typically involves matching objects in consecutive frames using features such as points, lines or blobs. Tracking methods are divided into four categories [4]. We survey three categories of them.

- Region-based tracking

Region-based tracking algorithms track objects according to variations of the image regions corresponding to the moving objects. Many methods maintain the background image dynamically [5-6]. McKenna et al. [7] propose an adaptive background subtraction method in which color and gradient information are combined to deal with shadows and unreliable color cues in motion segmentation. Tracking is then performed at three levels of abstraction: regions, people, and groups. Each region has a bounding box and regions can merge and split. A person is composed of one or more regions grouped together under the condition of geometric structure constraints on the human body, and a human group consists of one or more people grouped together. Therefore, using the region tracker and the individual color appearance model, perfect tracking of multiple people is achieved, even during occlusion.

- Active contour-based tracking

Active contour-based tracking algorithms track objects by representing their outlines as bounding contours and updating these contours dynamically in successive frames[8-11]. Peterfreund [12] explores a new active contour model based on a Kalman filter for tracking non-rigid moving targets such as people in spatio-velocity space.

In contrast to region-based tracking algorithms, active contour-based algorithms describe objects more simply and effectively and reduce time complexity. Even under disturbance or partial occlusion, these algorithms may track objects continuously. However, the tracking precision is limited at the contour level. A further difficulty is that the active contour-based algorithms are highly sensitive to the initialization of tracking, making it difficult to start tracking automatically.

- Feature-based tracking

Feature-based tracking perform recognition and tracking of objects by extracting elements, clustering them into higher level features and then matching the features between images. Feature-based tracking algorithms can further be classified into three subcategories according to the nature of selected features: global feature-based algorithms[13-14], local feature-based algorithms[15-16], and dependence-graph-based algorithms[17].

Jang et al. [18] propose a method combining above three algorithms. The method uses an active template that characterizes regional and structural features of an object is built dynamically based on the information of shape, texture, color, and edge features of the region. Using motion estimation based on a Kalman filter, the tracking of a non-rigid moving object is successfully performed by minimizing a feature energy function during the matching process.

However, there are several serious deficiencies in feature-based tracking algorithms.

- The recognition rate of objects based on 2-D image features is low, because of the nonlinear distortion during perspective projection and the image variations with the viewpoint's movement.
- The stability of dealing effectively with occlusion, overlapping and interference of unrelated structures is generally poor.

1.4 Assumptions

In this thesis, to concentrate on the methods for solving our proposed problems, we make following assumptions.

1. Objects have already been detected and different body parts have been segmented.
2. Each person is tracked independently. Interaction among people is not taken into consideration in this study.
3. Input images are color with resolution of 640×480.
4. The frame rate of videos is 30 frames/sec.
5. Cameras are stationary.

1.5 System Description

In this thesis, we develop a system to search the similar suspects when given a suspect image. The main modules include human detection, human body part segmentation and feature measurement, feature selection, and human searching.

In the module of human detection, we introduce our proposed method to detect foreground person in videos. After obtaining the moving suspect, we segment the suspect's body into several parts and extract the features of the corresponding parts.

In the module of feature selection, we use training videos to select those discriminate features. By using our proposed model and training videos from many features, we can judge which discriminate features can be used to find the similar suspects in videos more effectively. Finally, we can use these discriminate features to search the similar suspects in videos. The system flow diagram is shown in Fig.1.5.1. The main modules of the system are described as follows:

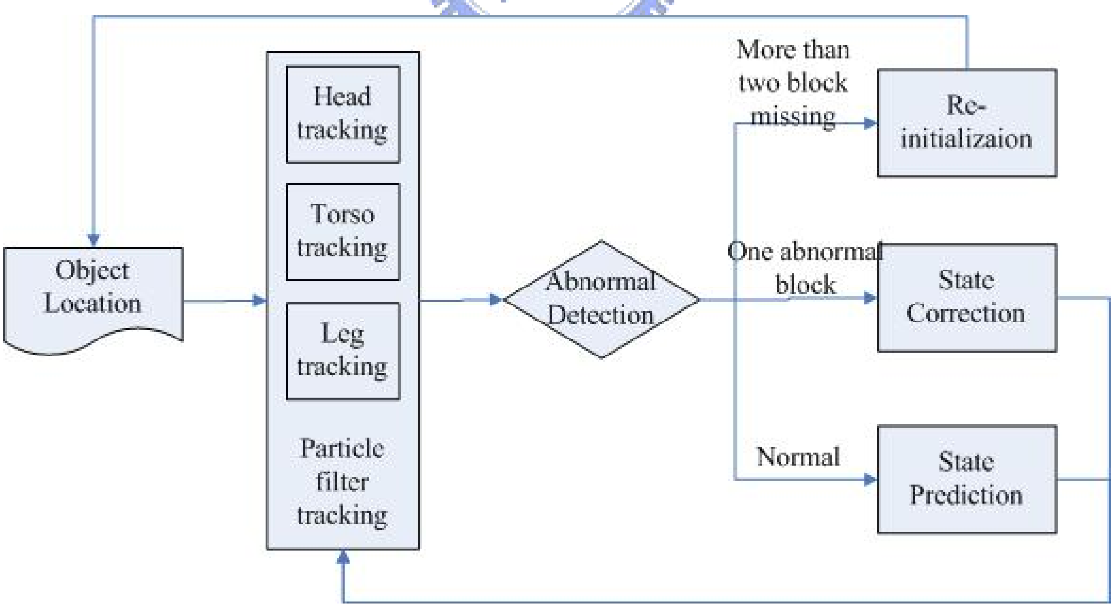


Fig.1.5.1 The system flow diagram

1.5.1 Object Location

We propose a modified frame difference method to detect the foreground person in videos. The color space we used can suppress shadows. However since the foreground person regions may be connected, we also propose a method to split the connected persons.

1.5.2 Human Body Decomposition and Three-parts tracking

After the suspect in a video was extracted, we need to segment the body into several parts according to part significances. Additionally we need extract features of different body parts for human searching.

1.5.3 Abnormal detection

Not all features we extracted are discriminate; hence we propose a model to select those discriminate features. In the model, we use each feature to match the training persons with the suspects in training videos to test the discriminative capability of each feature.

1.5.4 State Correction

With the discriminate features, we determine the combination of features to measure the similarity between the video suspects and the target suspect. According to the similarities of frames, we decide which suspects are similar to the target person.

1.6 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 describes the method of Particle filter for object tracking. Chapter 3 describes the module of Three-part Human Tracking and Consistency Check.Chapter 4 describes experimental results and their analyses. Finally, chapter 5 presents some conclusions and suggestions for future work.

CHAPTER 2 Particle Filter for Object Tracking

It is hard to obtain the information of background, so we can not use the method of background subtraction to detect the moving persons. The third method, frame differencing, can be applied to find the moving regions in a video sequence. In this study, we propose a modified method of frame difference to resolve the situation. Without available information about background, we use our modified frame difference method to obtain moving persons in videos.

2.1 Definitions and Formulations

Definition of particle filters:

Particle filters have been applied to solve non-linear and non-Gaussian estimation problems successfully.

The basic idea of a particle filter is how a probability density propagates.

Color features have been used to represent objects effectively.

A target is tracked with a particle filter by comparing its observation with the observation of the object model.

2.2 Particle Filter

The key idea of a particle filter is to approximate the probability distribution by a weighted sample set

$$S = \left\{ s^{(n)}, \pi^{(n)} \mid n = 1 \ldots N \right\}$$

Each sample is called a particle, represents one hypothetical state of the object, with a corresponding discrete sampling probability π, where

$$\sum_{n=1}^{N} \pi^{(n)} = 1$$

A set $\{(s^1, \pi^1), (s^2, \pi^2), \ldots, (s^N, \pi^N)\}$ is generated; these measurement points are called "particles". Each element of the set is weighted in terms of the observations and N samples are drawn with replacement, by choosing a particular sample with probability

According to these $\{\pi^{(n)}\}$, the estimated state of an object can be determined from expectation of $\{s^{(n)}\}$ at each time step.

$$E(S) = \sum_{n=1}^{N} \pi^{(n)} s^{(n)}$$

We use a weighting function (Epanechnikov kernel) to calculate the color distribution $P_y = \{P_y^{(u)}\}_{u=1...m}$ at location y, Where I is the number of pixels in the region, and δ is the Kronecker delta function The parameter f is a normalization factor,

$$P_y^{(u)} = f \sum_{i=1}^{I} k\left(\frac{\|y - x_i\|}{a}\right) \delta[h(x_i) - u]$$

$$f = \frac{1}{\sum_{i=1}^{I} k\left(\frac{\|y - x_i\|}{a}\right)}$$ ensures that $\sum_{u=1}^{m} P_y^{(u)} = 1.$

2.3 Color Spaces

Feature Selection: Feature selected may make the tracking process more successfully. In our experiments, color histograms have been adopted because they are robust and modest time consuming. The colors of the states are transformed from RGB to YCbCr.

Color space transform of the transform action:

$$\begin{cases} Y = 0.299 * R + 0.587 * G + 0.114 * B \\ C_b = -0.16874 * R - 0.33126 * G + 0.5 * B \\ C_r = 0.5 * R - 0.41869 * G - 0.08131 * B \end{cases}$$

$$\begin{cases} if \ \ C_b < 0 \ then \ C_b = 0 \\ if \ \ C_b > 0 \ then \ C_b = 255 \end{cases} \quad and$$

$$\begin{cases} if \ \ C_r < 0 \ then \ C_r = 0 \\ if \ \ C_r > 0 \ then \ C_r = 255 \end{cases}$$

The color histograms are computed in three color channels of Y, Cb, Cr respectively, and then discretized into *m*-bins. In our experiments, we let m = 8, so the YCbCr space are divided into (8+8+8) bins.

2.4 Color Histogram Smoothing

YCbCr histogram:

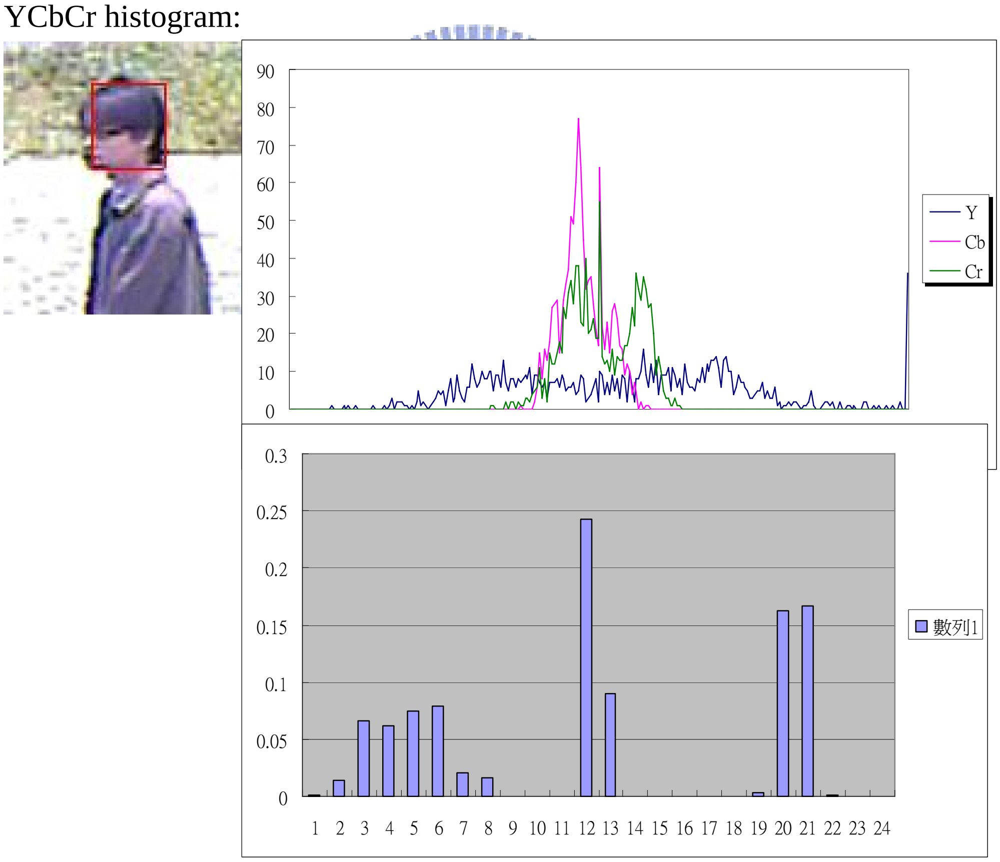


Fig. 2.1 The YCbCr histogram graph (a)(b)

(c)

(a) A sample box (red box), (b) The histogram graph of (a) (c) in the YCbCr color space (c) 8-bins histogram of Y, Cb, Cr of (b) after normalization.

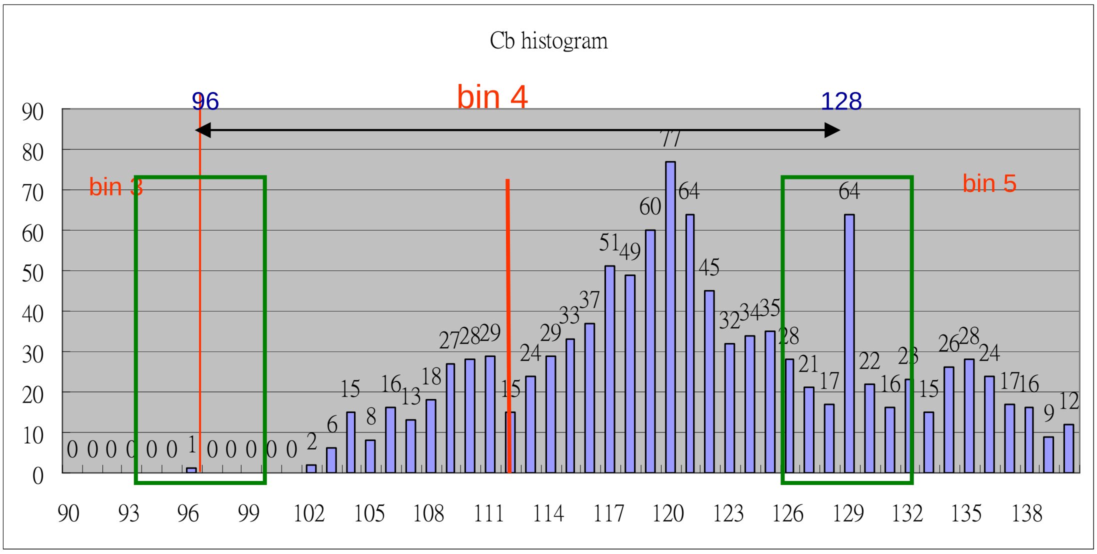


Fig. 2.2 The histogram of Cb

In Fig. 2.2, while reducing 256 levels into 8 bins, we observe the level 129 is sensitive and we don't want this level change its bin frequently in continuous image. We propose a specific distribution function to smooth color histograms.

How to smooth the histogram, we use the Gaussian distribution function as kernel function $K(d)$ in following. We use this mask locally to convolve with each bin.

$$K(d) = \frac{1}{\sqrt{2\pi k}} e^{\frac{d^2}{2k^2}}$$

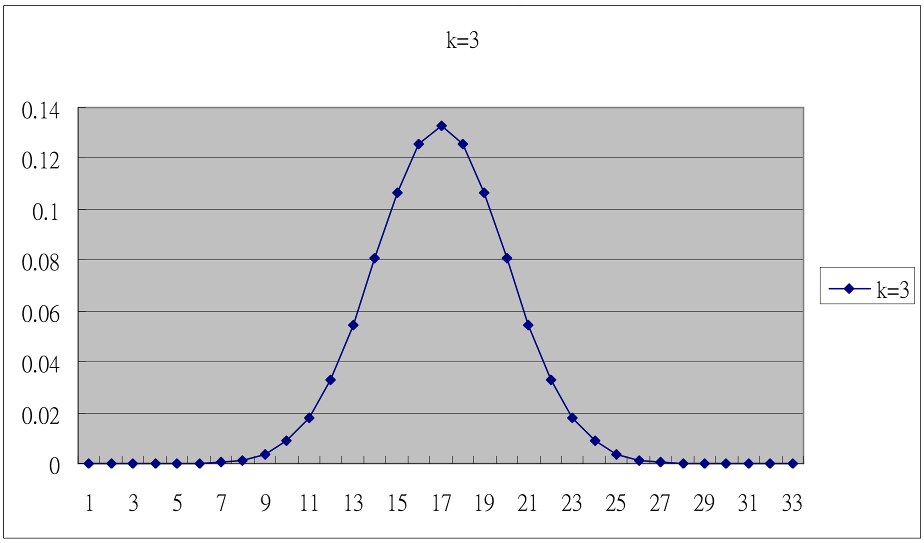
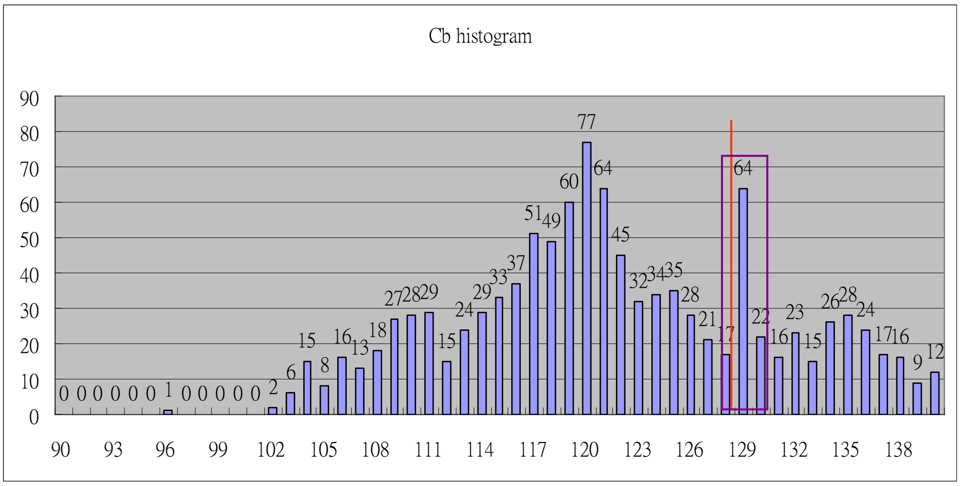


Fig. 2.3 Gaussian distribution function with k =3.

Example of histogram smoothing : We shift one level left to represent another color histogram, take the histogram of Cb color to describe. Observing the difference in Fig. 2.5 and Table. 2.1.



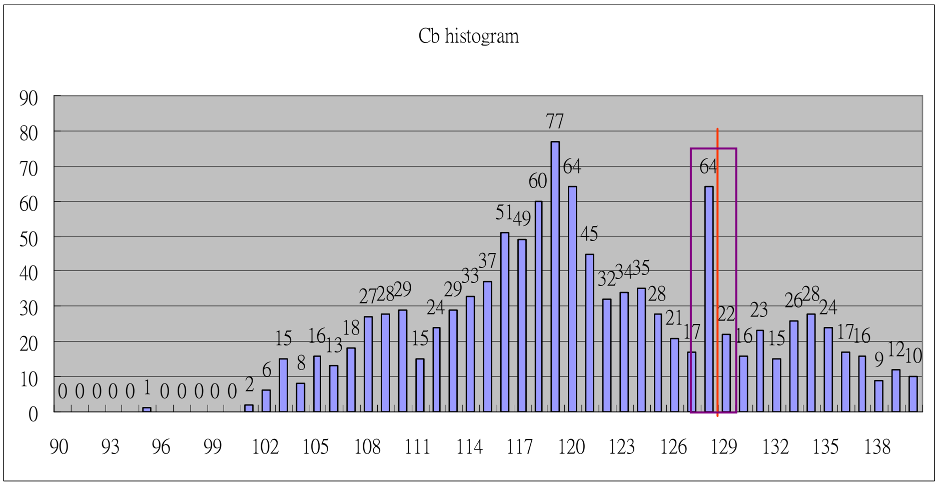


Fig. 2.4 Cb histogram bin in detail (a) is Cb original histogram diagram, (b) the histogram after shifting one level left of (a).

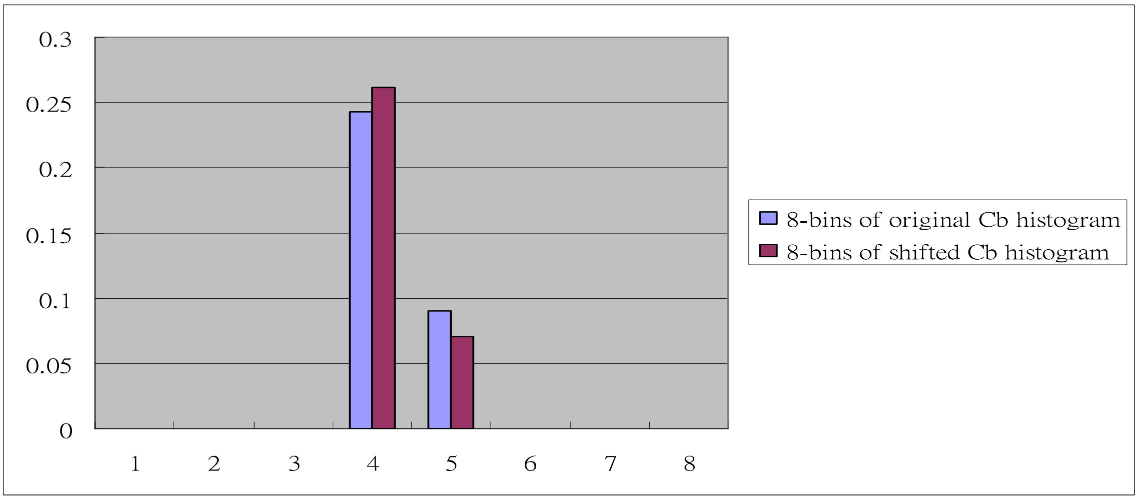


Fig. 2.5 The diagram of original histogram and shifted histogram.

	Original histogram		Shifting histogram		Weight of Cb
Without smoothing	bin 4 = 0.242832	bin 5 = 0.090203	bin 4 = 0.261947	bin 5 = 0.071087	0.07
With smoothing	bin 4 = 0.299165	bin 5 = 0.034168	bin 4 = 0.309746	bin 5 = 0.023588	0.09

Table 2.1 The histogram value in bin 4 and bin 5 with the process of smoothing,

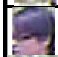
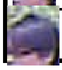
Original state 		
	Weight of original histogram	Weight of smoothing histogram
 Particle 1	0.506121	0.400781
 Particle 2	0.006238	0.003949
 Particle 3	0.118476	0.203282
 Particle 4	0.369166	0.391989

Table 2.2 Example to describe the result of smoothing

We observe the weight of particle 1 is larger in original histogram comparatively so it may make next state go up, after smoothing we lower the weight to reduce the influence of particle 1 . After histogram smoothing, we raise the weight of the similar particle to the original state, and decrease the weight of the unlike particle to the original state.

2.5 Color-based Particle Filter

Suppose that a target is bounded by a rectangle. K. Nummiaro Proposed a particle filter using seven parameters as a state. We assume the scale change is not fast so we ignore the scale change parameter. Each state is described as $S = \{X, Y, W, H, \dot{X}, \dot{Y}\}$,where (X, Y) represents the center of the box, (W, H) are the size of the rectangle, $(\dot{X}, \dot{Y})$are the speed of center.

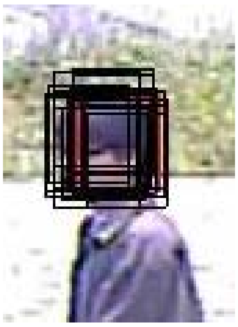


Fig. 2.6 An example of the measurement box (black boxes)

The sample set is propagated according to a dynamic model. Where *A* defines the deterministic component of the model, and *W* is a multivariate Gaussian random variable. In our applications, we use a first order model for *A* describing a region moving with constant velocity

$$S_t = AS_{t-1} + W$$

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Gaussian random variable W

$$W = \begin{bmatrix} N(0,\sigma_X^2) & 0 & 0 & 0 & 0 & 0 \\ 0 & N(0,\sigma_Y^2) & 0 & 0 & 0 & 0 \\ 0 & 0 & N(0,\sigma_{\dot{X}}^2) & 0 & 0 & 0 \\ 0 & 0 & 0 & N(0,\sigma_{\dot{Y}}^2) & 0 & 0 \\ 0 & 0 & 0 & 0 & N(0,\sigma_W^2) & 0 \\ 0 & 0 & 0 & 0 & 0 & N(0,\sigma_H^2) \end{bmatrix}$$

$, where\ \sigma_X^2 = 10,\ \sigma_Y^2 = 10,\ \sigma_{\dot{X}}^2 = 2,\ \sigma_{\dot{Y}}^2 = 2,\ \sigma_W^2 = 2,\ \sigma_H^2 = 2$

Feature update:

$$p_t = p_{t-1} \times 0.9 + \overline{q}_t \times 0.1$$

2.6 Particles Initialization

In the initialization stage, we assign to all particles the same rectangle position but different speeds. Because we do not know where the target moves toward, we broaden the search region in all possible directions.

Fig. 2.7 Broaden the search region in all directions initially.

CHAPTER 3 HUMAN BODY DECOMPOSITION AND FEATURE MEASUREMENT

After detecting the suspect in the videos, the next step is to decompose the body of the suspect and to extract the features of the suspect. In this chapter, we use our method to segment the body of the suspect into three parts because the different parts have different weights. When the query suspect was given, the user can assign different weights to different body parts. Sec. 3.1 introduces how to decompose the body of the suspect into three parts. Then we propose a method to extract the features in different

3.1 Body Part Decomposition

In this section, we introduce our method to decompose the body of the detected person. When the query suspect was given, the user can manually divide the body of the query suspect into three parts and obtain the ratio of the three parts. We then decompose the bodies of the suspects in a video according to the three ratios of the parts in the query suspect. We decompose the body of detected person into three parts: head part, upper-body part and lower-body part. When tracking a human, some researchers only track the head. During the tracking process, we may lose the tracking objects. Tracking error samples:

- In Fig. 3.1, the target is lost due to head motion.
- In Fig. 3.2, the target is lost due to the human motion irregularly.
- In Figs. 3.3 and 3.4, the target may be affected by background.

Frame 285 Frame 279

Fig. 3.1 Head moving may occur while tracking.

Frame 3489 Frame 3351

Frame 3522

Fig. 3.2 Person walks back and fort

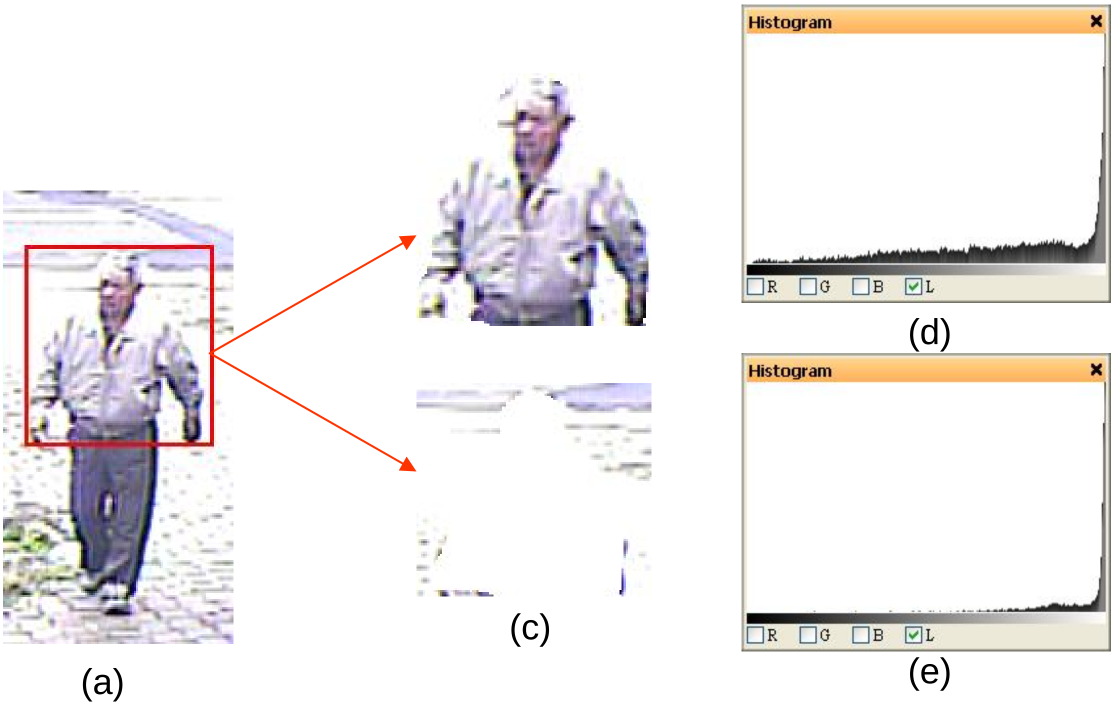


Fig. 3.3 The histogram between background and foreground is similar (a) the original block (b) tracking part in the red box. (c) Background part in the red box. (d) (e) the histogram graphs of (b) (c)

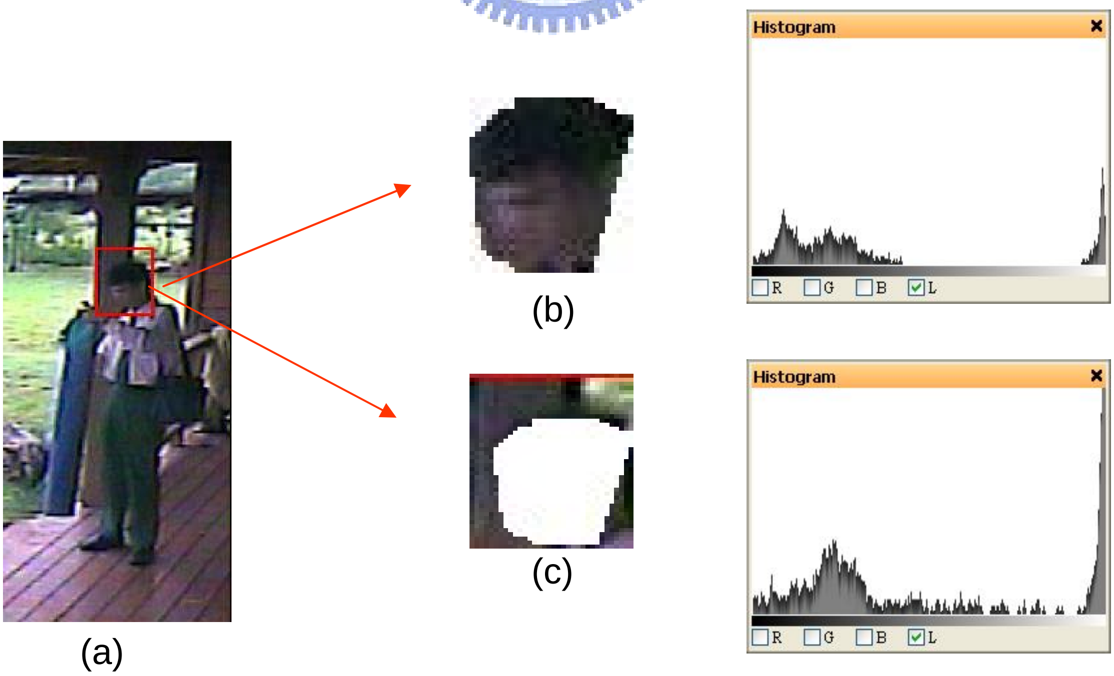


Fig. 3.4 The histogram between background and foreground is similar(2)

To track a human body, spatial information from different body parts can provide more reliable knowledge. The decomposition has already been completed by the module of foreground object detection. When we detect an unreliable tracking result while tracking, we can adjust its position according to their spatial relationship with other body parts.

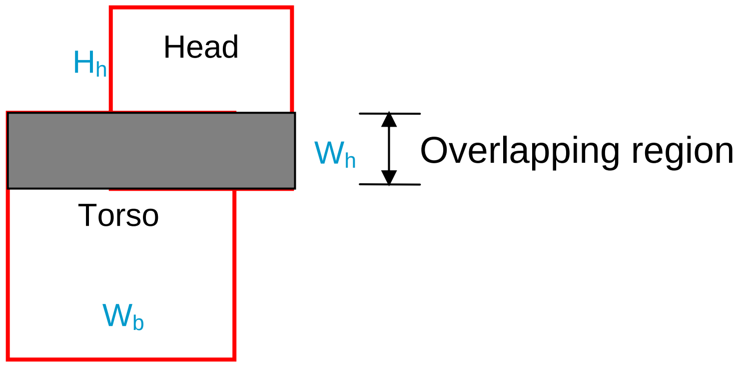
Fig. 3.5 Example of three parts on tracking objects

$$\begin{cases} Y = 0.299 * R + 0.587 * G + 0.114 * B \\ C_b = -0.16874 * R - 0.33126 * G + 0.5 * B \\ C_r = 0.5 * R - 0.41869 * G - 0.08131 * B \end{cases} \quad \begin{cases} if\ C_b < 0\ then\ C_b = 0 \\ if\ C_b > 0\ then\ C_b = 255 \end{cases} \quad and \quad \begin{cases} if\ C_r < 0\ then\ C_r = 0 \\ if\ C_r > 0\ then\ C_r = 255 \end{cases}$$

3.2 Color Feature Consistency Checking

3.2.1 Overlapping Detection

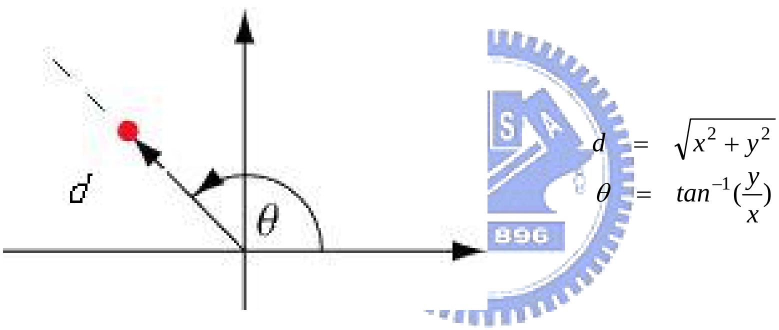
During tracking, one block may be covered by another one. If the height of an overlapping region is more than 1/3 of the height of the head block, an overlapping occurs between the head and torso. The same condition happens between the torso and leg blocks.



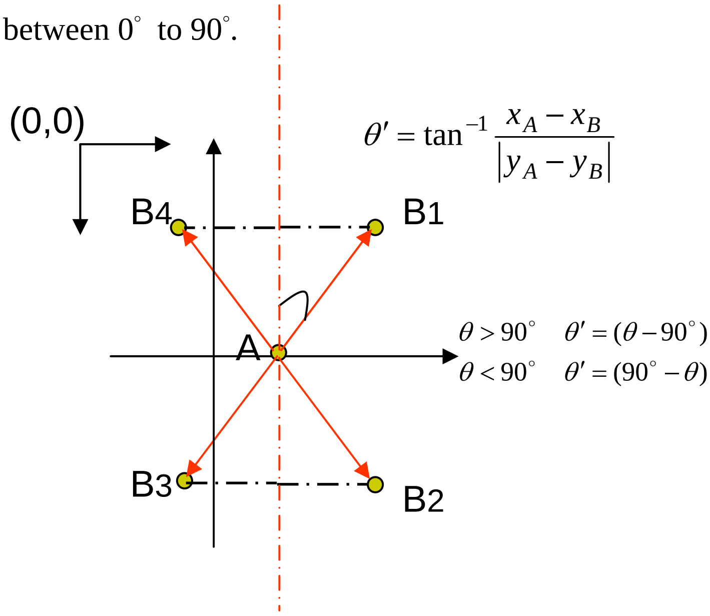
If Overlap region > (1/3) Hh
==>overlapping detect

3.2.2 Incorrect State Detection

We define angle and distance constraints to detect abnormal states.

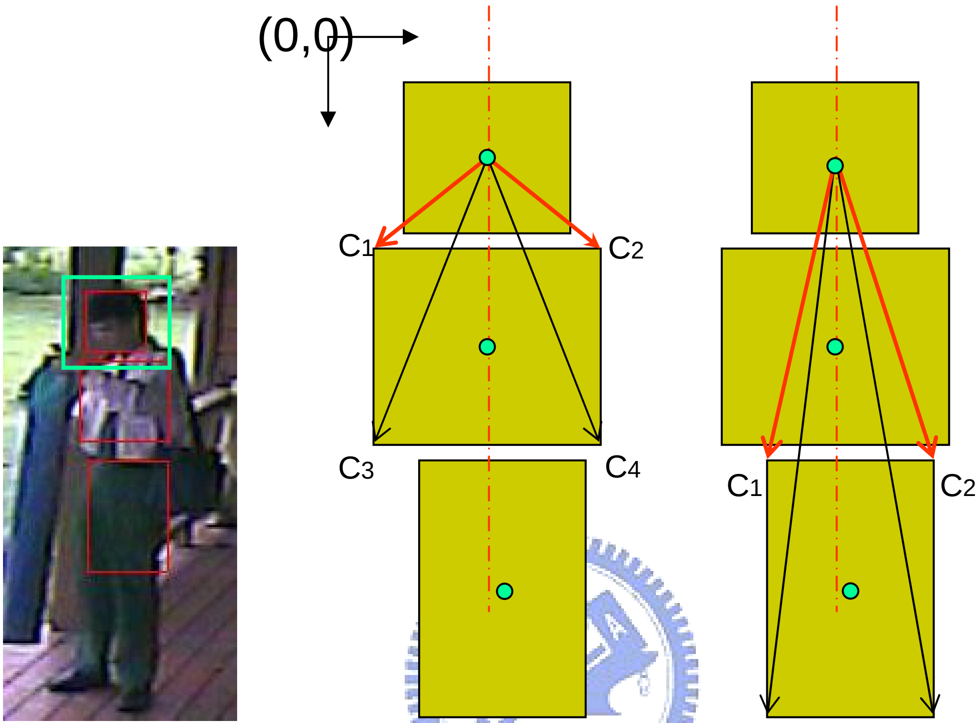


Angle constraint: The angle from point *A* to *B* is defined as : We restrict the angle between 0° to 90°.



The allowable angle range, *Ang (A,B)*, from block *A* to block *B* is defined as the minimum and maximum angle from the center of block *A*, *Center*(*A*), to the four corners of block *B* (*C*1(*B*), *C*2(*B*), *C*3(*B*) and *C*4(*B*)). That is :

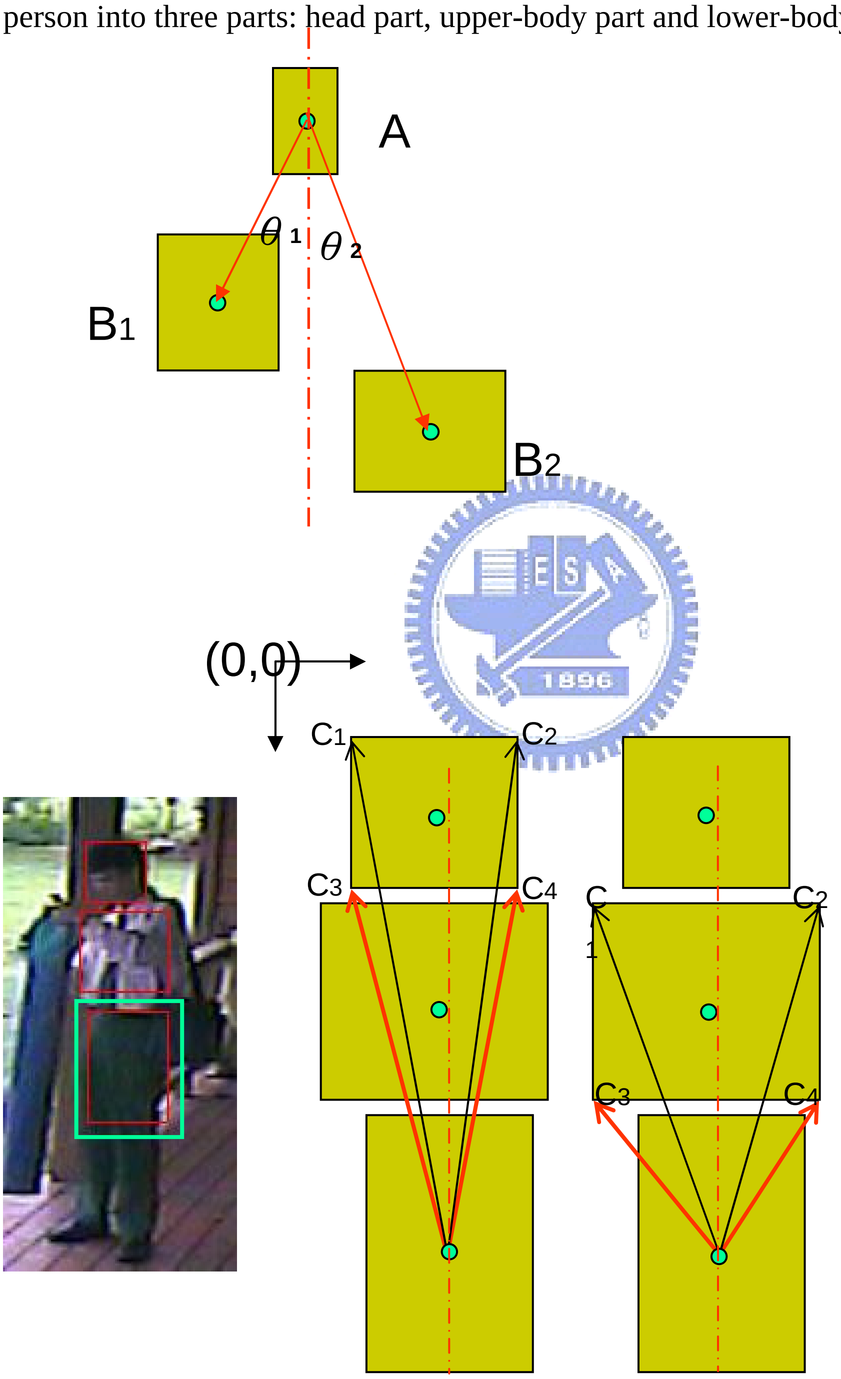
$$(\min_i \theta(Center(A), C_i(B)), \max_i \theta(Center(A), C_i(B))),$$
$$where\ i = 1,2,3,4$$



	Angle_max	Angle_min
Head to torso	**43 degree**	**-54 degree**
Head to leg	**13 degree**	**-19 degree**

Fig. 3.6 Angle constraint on the head block.

This range can be obtained from the training images. We define an angle range from a body part *A* to another part *B*, *Ang* (*A*,*B*)**.**For a specific part *A* in the test frame, we compute the angles $\theta 1$ and $\theta 2$ from the center of part *A* to the centers of another two part *B1* and *B2*. If $\theta 1$ and $\theta 2$ are located within the angle range *Ang (A, B*1*)* and *Ang (A, B*2*)*, respectively, we say that part *A* satisfies the angle constraint. In this section, we introduce our method to decompose the body of the detected person. When the query suspect was given, the user can manually divide the body of the query suspect into three parts and obtain the ratio of the three parts. We then decompose the bodies of the suspects in a video according to the three ratios of the parts in the query suspect. As shown in Fig. 3.1.1, we decompose the body of detected person into three parts: head part, upper-body part and lower-body part.

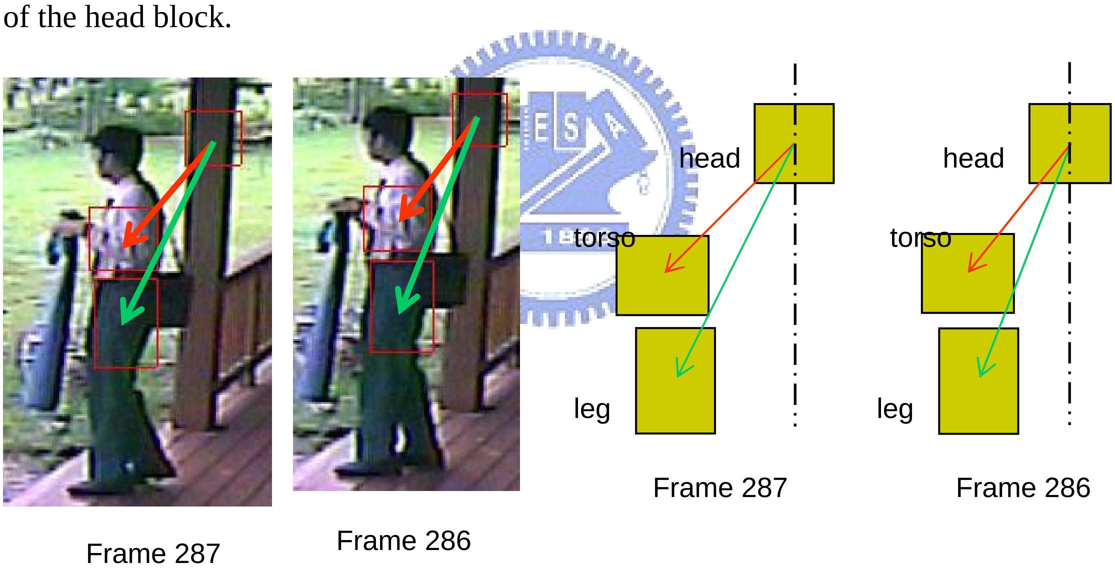


	Angle_max	Angle_min
Leg to head	13 degree	-7 degree
Leg to torso	30 degree	-31 degree

Fig. 3.7 Angle constraint of the leg block.

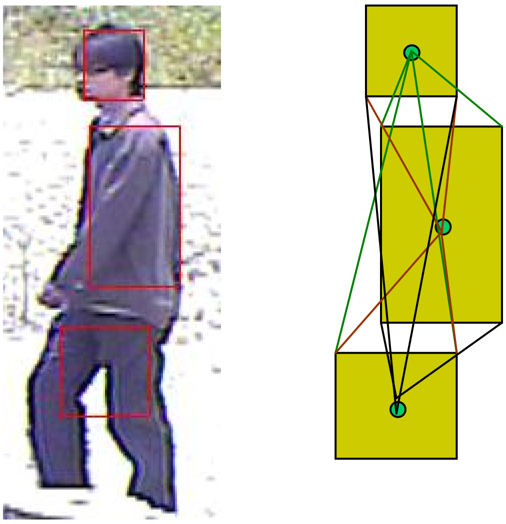
Abnormal examples :

In frame 287, the angles from head to torso and leg both violate the angle constraint of the head block.



	Frame 286	Frame 287
Angle of head to torso	42.5 degree	44.5 degree
Angle of head to leg	25.3 degree	27 degree

Fig. 3.8 Abnormal example of the head block



	Angle_max	Angle_min
Head to torso	17.3 degree	-47.5 degree
Head to leg	12.0 degree	-7.3 degree
Torso to head	26.5 degree	-3.1 degree
Torso to leg	34.1 degree	-4.8 degree
Leg to head	3.8 degree	-8.8 degree
Leg to torso	5.5 degree	-45.0 degree

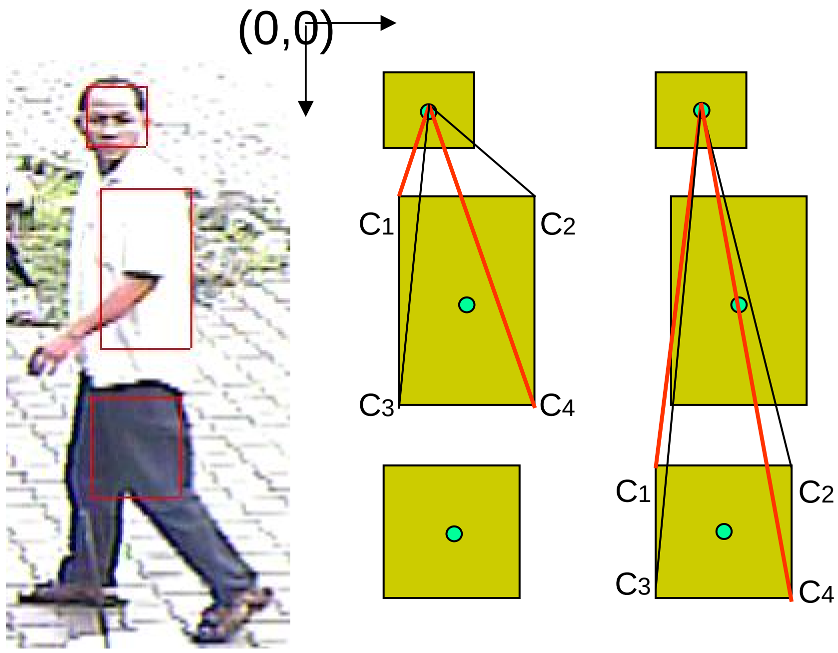
Fig. 3.9 Another example using angle detection.

	Frame 343	Frame 344
Angle of leg to head	-33.0 degree	-34.4 degree
Angle of leg to torso	-44.2 degree	-46.6 degree

Fig. 3.10 Abnormal example of the leg block.

The angle constraint is used to detect horizontal abnormality, while the distance constraint is used to detect vertical abnormality. The allowable distance range *Dist* (*A*,*B*) from block *A* to block *B* is :

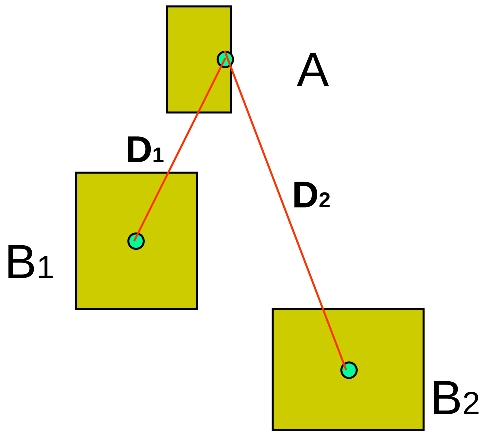
$$(\min_i Dist\ (Center(A), C_i(B))\ ,\ \max_i Dist\ (Center(A), C_i(B))),$$
$$where\ i = 1,2,3,4$$



	Distance_max (pixels)	Distance_min (pixels)
Head between torso	120.88	37.643
Head between leg	202.20	130.91

Fig. 3.11 Distance constraint of the head block.

For a specific part *A* in the test frame, we compute the distances *D*1 and *D*2 from the center of part *A* to the centers of another two part *B*1 and *B*2. If *D*1 and *D*2 are located within the distance range *Dist* (*A*, *B*1) and *Dist* (*A*, *B*2), respectively, we say that part satisfies the distance constraint.



Abnormal example:

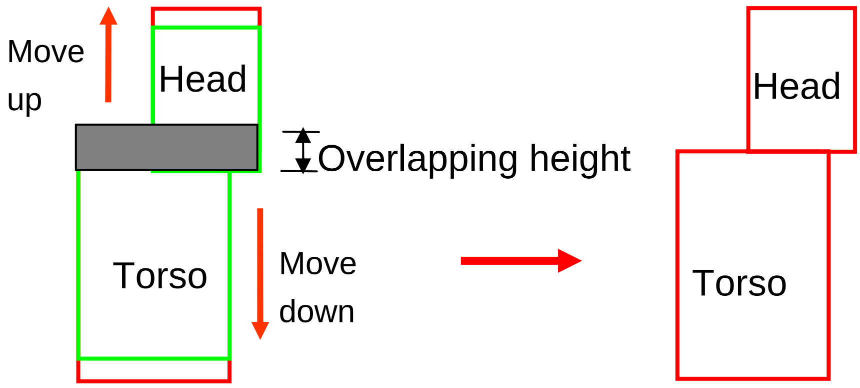
In frame 832 the distance between the center of the torso block and another block violates the distance constraint.

3.3 State Correction

The content of the dress of a person is another characteristic to discriminate different people. The contents of a dress can be described by the edge distribution. The edge distribution is also a kind of texture representation. In this study, we use the gradient (edge) as the texture feature when comparing the dresses of two suspects.

3.3.1 Overlapping Correction

Fig. 3.12 Example of overlapping adjustment between the head block and torso blocks.



Move the head block up d pixels and torso block down

d pixels, where d = Overlapping height/2. In this section, we introduce our method to decompose the body of the detected person. When the query suspect was given, the user can manually divide the body of the query suspect into three parts and obtain the ratio of the three parts. We then decompose the bodies of the suspects in a video according to the three ratios of the parts in the query suspect.

3.3.2 Position Adjustment

We propose to adjust the position in horizontal direction and vertical direction separately.

X coordinate adjustment:

Where A is the abnormal block and $B1$ and $B2$ are the other two normal block

$$x_A = \frac{(x_{B_1} + x_{B_2})}{2}$$

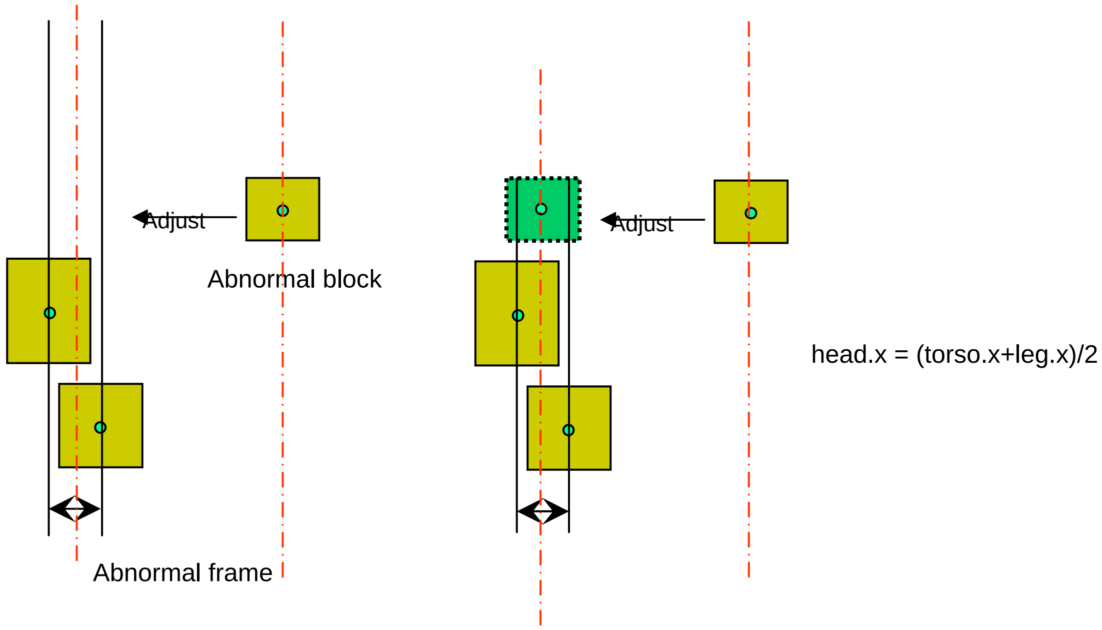


Fig. 3.13 State adjustment of the head block, where
the green rectangle is the result of adjusted state.

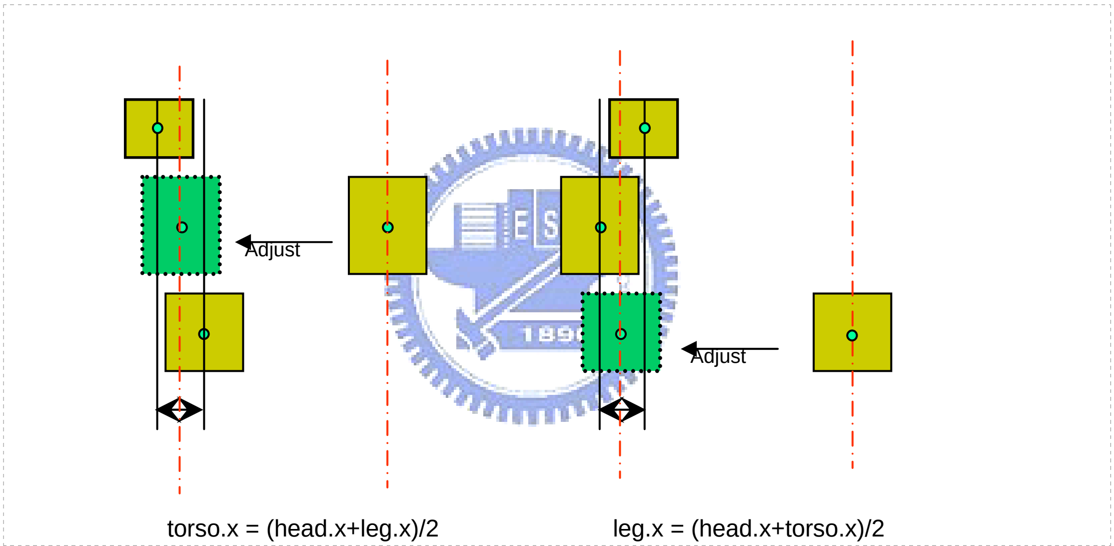


Fig. 3.14 Adjustment of the torso block and leg blocks.

Y coordinate adjustment:

Assume that the distance between head and torso is 5~25 pixels and that between torso and leg is 10~35 pixels. We adjust the *y* coordinate of abnormal blocks according to the allowable distances.

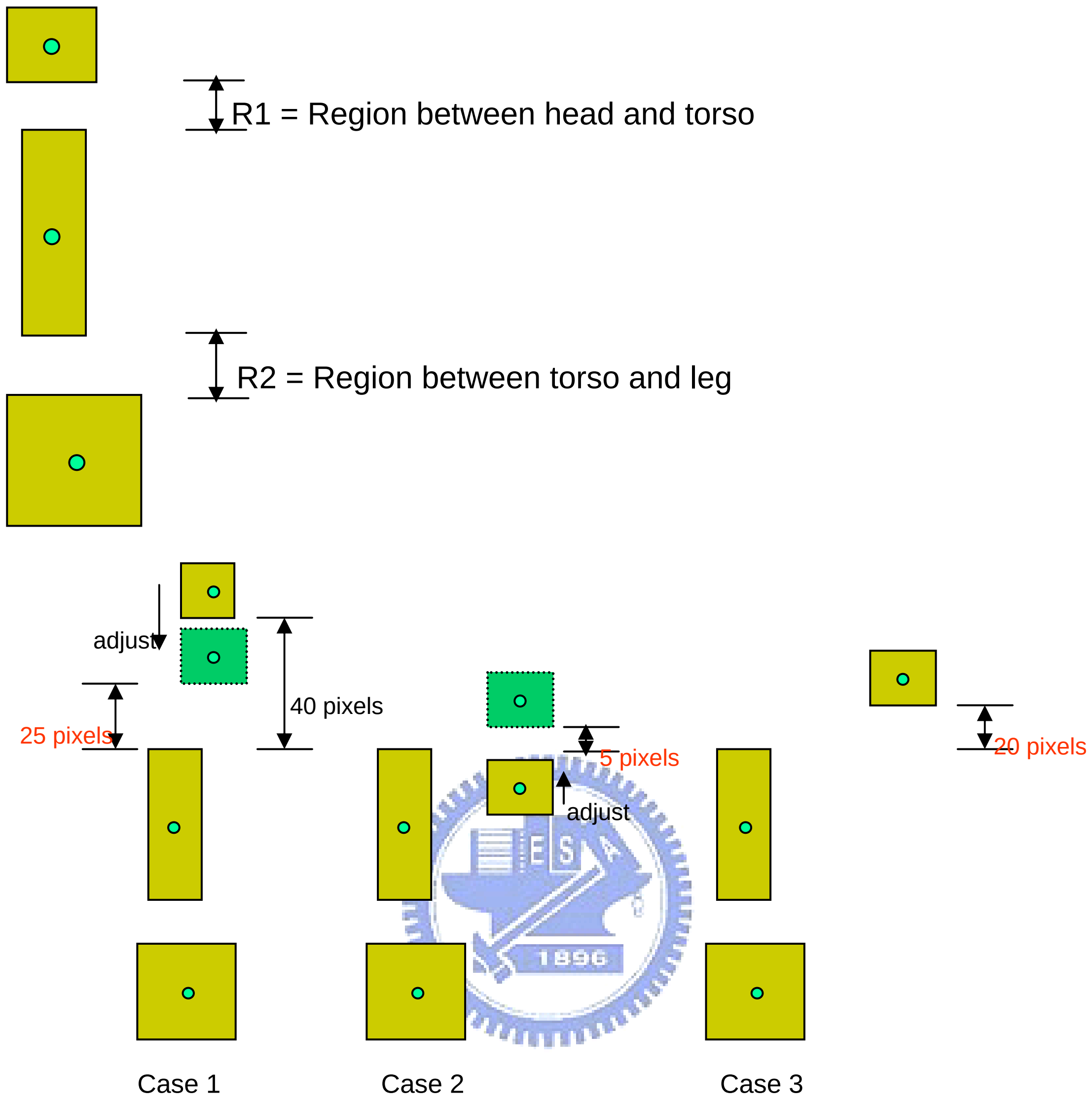


Fig. 15 Three conditions of head block

Rule for adjusting the head block :

Case 1 : if the region between head and torso is bigger than 25 pixels, then move head block to the position where the region between head and torso is 25 pixels.

Case 2 : if the region between head and torso is smaller than 5 pixels, then move head block to the position where the region between head and torso is 5 pixels.

Case 3 : if the region between head and torso is within 5~25 pixels, then keep the y coordinate of head block.

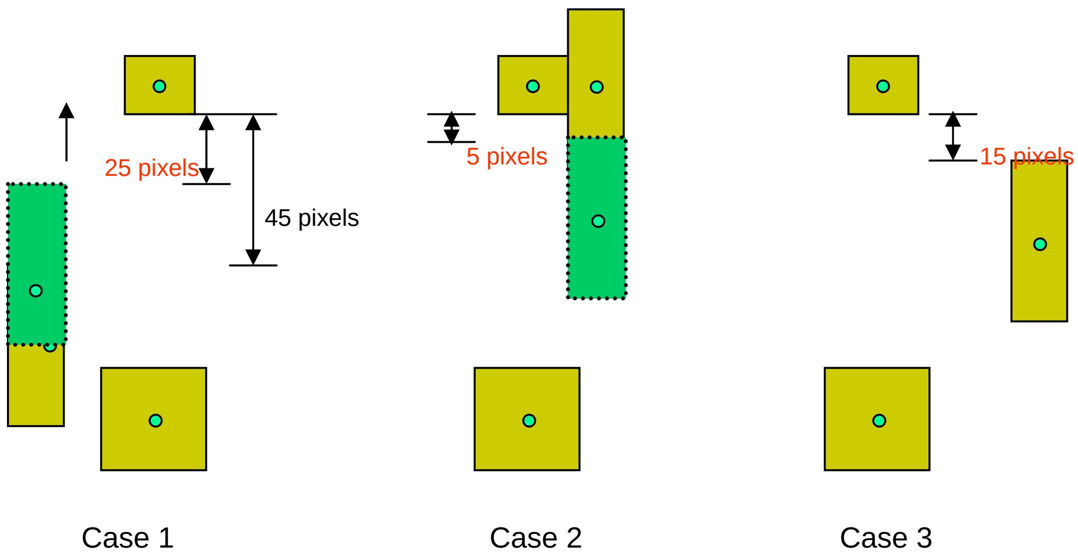


Fig. 16 Three conditions of torso block

Rule for adjusting the torso block :

Case 1 : if the region between head and torso is bigger than 25 pixels, then move torso block to the position where the region between head and torso is 25 pixels.

Case 2 : if the region between head and torso is smaller than 5 pixels, then move torso block to the position where the region between head and torso is 5 pixels.

Case 3 : if the region between head and torso is within 5~25 pixels, then keep the y coordinate of head block.

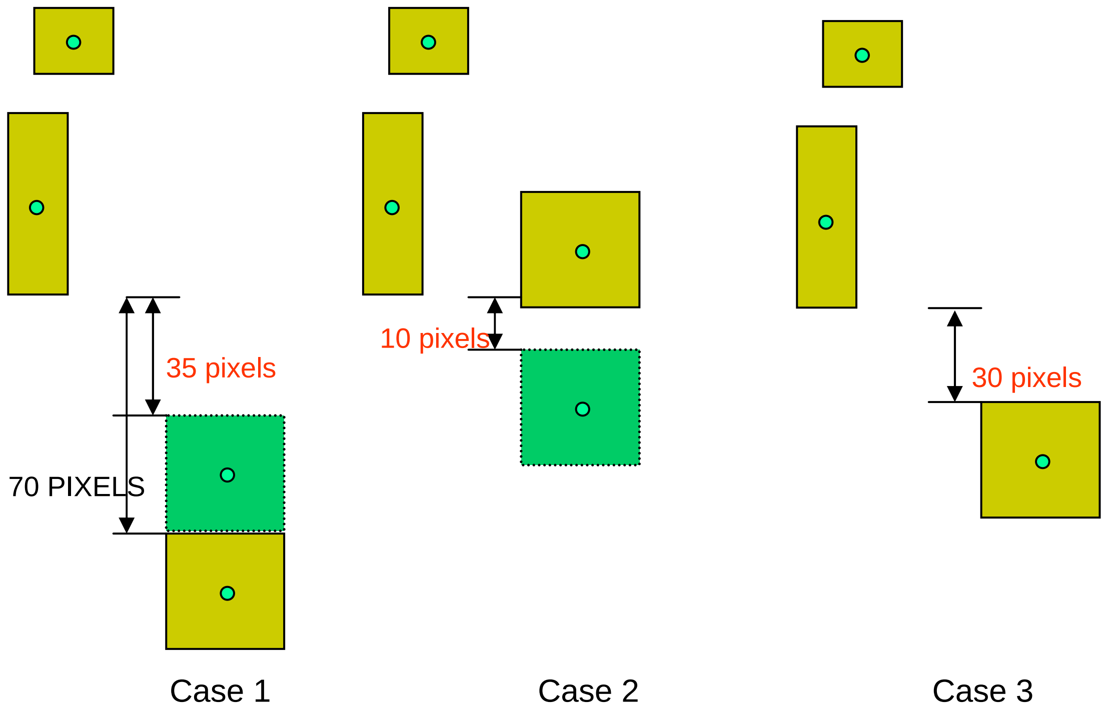


Fig. 17 Three conditions of leg block

Rule for adjusting the leg block :

Case 1 : if the region between head and torso is bigger than 35 pixels, then move leg block to the position where the region between head and torso is 35 pixels.

Case 2 : if the region between head and torso is smaller than 10 pixels, then move leg block to the position where the region between head and torso is 10 pixels.

Case 3 : if the region between head and torso is within 10~35 pixels, then keep the y coordinate of head block.

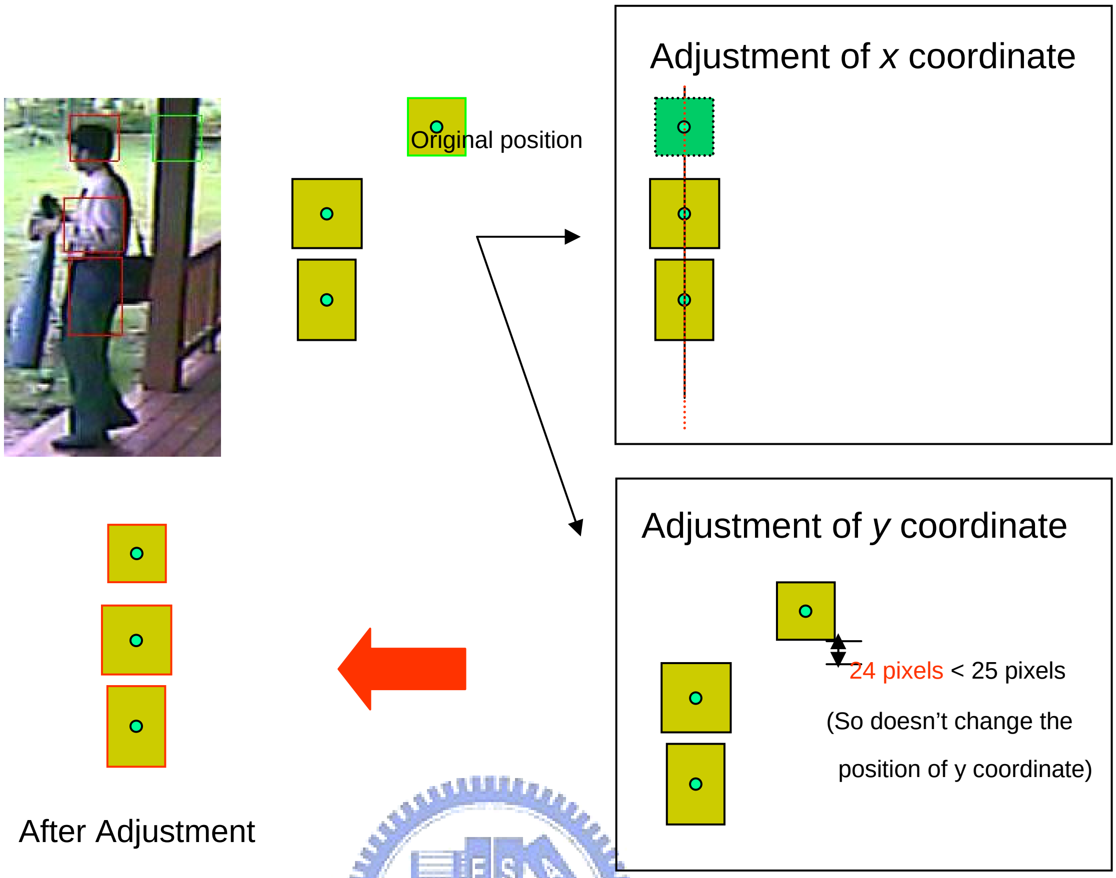


Fig. 3.18 Example of adjustment the head block

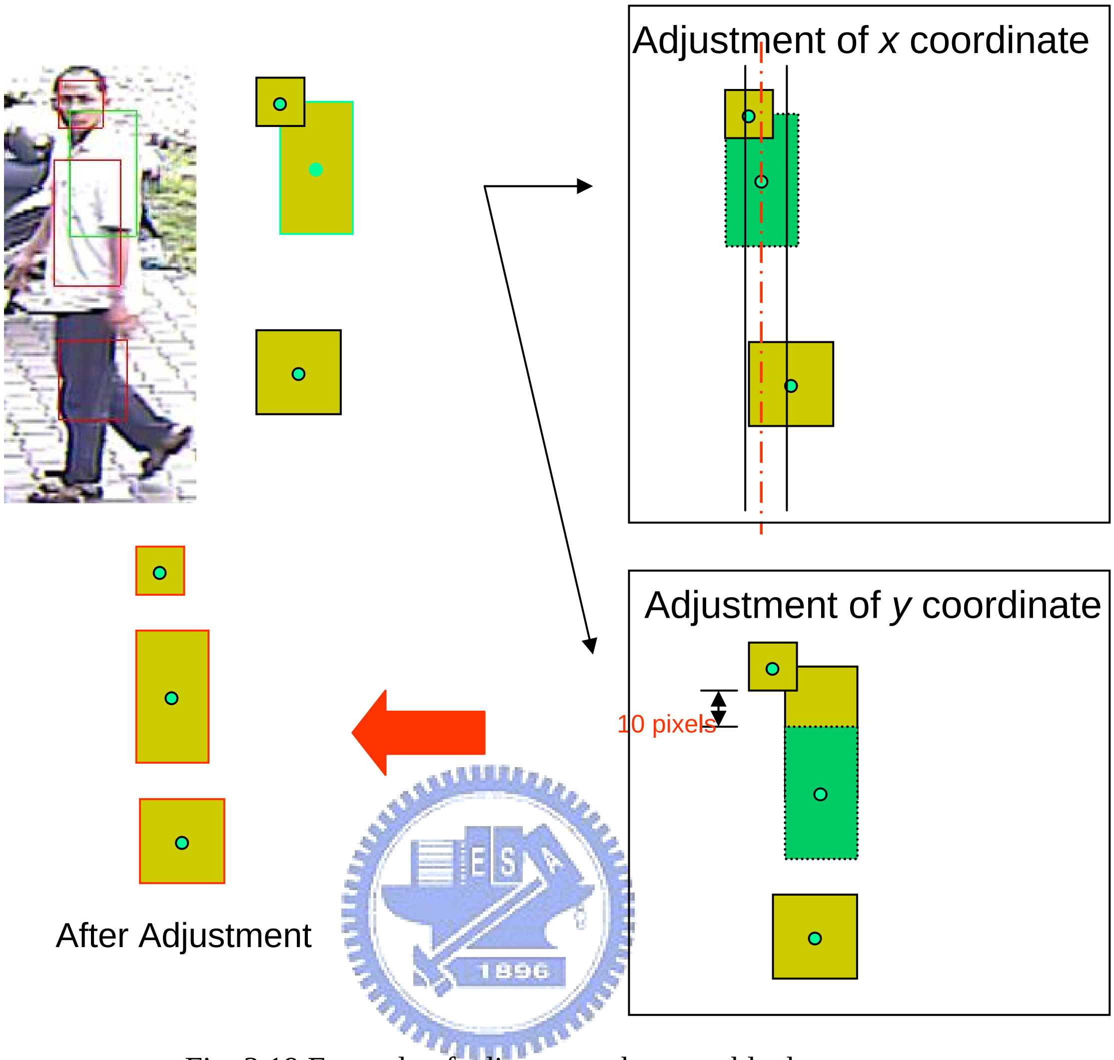


Fig. 3.19 Example of adjustment the torso block

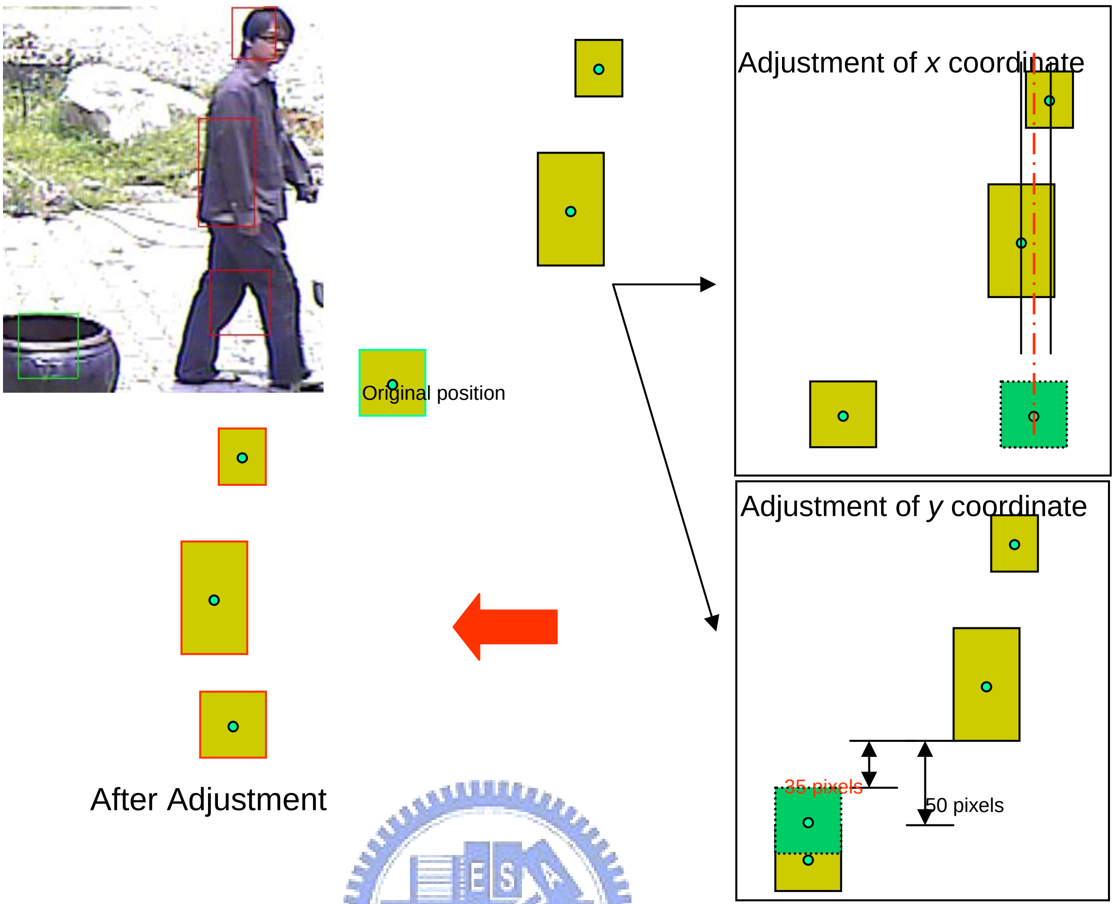


Fig. 3.20 Example of adjustment the leg block

CHAPTER 4 EXPERIMENTAL RESULTS AND DISCUSSION

In this chapter, we present the experiment results and the discussion of our system. The proposed approach has been implemented in a personal computer with Pentium IV 3GHz CPU and 1G RAM. The software environment is Microsoft Windows XP and Microsoft Visual C++ .NET 2003

The input images are color, whose resolutions are 720 x 480. We separate our experiments into five parts: 3-parts block human tracking, abnormal detection, and state Adjustment

4.1 Experiment Result

4.1.1 Adjustment of variance:

When we calculate the weight of sample states, we adjust the standard deviation σ such that the more similar these two color histograms are, the larger the weight is. In table 4.1, the number of particle is 16 and σ = 0.05 and 0.025. In our following tracking experiments, we take σ=0.025.

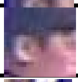
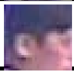
original state 		
Particle	Weight of σ =0.05	Weight of σ =0.025
2 	0.024810	0.005018
7 	0.229348	0.432472
9 	0.025291	0.005259
11 	0.161053	0.213259
14 	0.027217	0.006091
15 	0.153358	0.193365

Table 4.1 Comparison of standard deviation σ =0.05 with σ =0.025

Tracking

- 40 video clips
- Average 175 frames

Frame 100 Frame 125 Frame 140

Frame 150 Frame 160 Frame 170

Fig 4.1 Example of tracking without abnormal.

Frame 832 Frame 846 Frame 862

Fig. 4.2 Test of scale change

Frame 280 Frame 284 Frame 289

Fig. 4.3 Head moving case in frame 280,284 and 289, respectively : The 1st row uses YC_bC_r color histogram, the 2nd row is improved by histogram smoothing.

Frame 287 Frame 302 Frame 333

Fig. 4.4 The body block of 1st row is affected by background, the 2nd row uses overlapping detection and adjustment.

- Example of adjustment the head block.

Frame 270 Frame 287 Frame 305

Frame 289

Fig. 4.5 In frame 289 system detect the abnormal of head block and adjust it

Frame 826

Frame 832

Frame 852

Frame 840

Fig. 4.6 In frame 840 system detect the abnormal of torso block and adjust

Frame 328

Frame 342

Frame 353

Frame 346

Fig. 4.7 In frame 346 system detect the abnormal of leg block and adjust it.

Fig. 4.8　Our tracking target is not completely occluded.

Fig. 4.9 The tracking target is not completely occluded.

	30-th frame		60-th frame		90-th frame		120-th frame		150-th frame	
	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2
Head	85%	100%	85%	100 %	75%	95 %	77.7 %	88.8 %	76.4 %	88.2 %
Torso	85%	95%	80%	100 %	75 %	100 %	72 %	88.8 %	76.4 %	82.3 %
leg	95%	100%	90%	100 %	85 %	100 %	83 %	88.8 %	88.2 %	82.3 %

	180-th frame		210-th frame		240-th frame		270-th frame		300-th frame	
	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2
Head	81.2%	87.5%	78.5%	92.9 %	76.9%	92.3 %	76.9 %	92.3 %	76.9 %	92.3 %
Torso	75%	87.5%	85.7%	85 %	76.9 %	84.6 %	76.9 %	84.6 %	76.9 %	84.6 %
leg	75%	87.5%	71.4%	85 %	69.2 %	84.6 %	69.2 %	84.6 %	69.2 %	84.6 %

Table 4.2 The data to show the efficient of our experiment.

M1 = Tracking processes without detection and correction.

M2 = Tracking processes with detection and correction.

4.2 Analysis of Erroneous Results

- Occlusion between people and the background.

CHAPTER 5 CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this thesis, we have proposed the suspect searching system which consists of four main phases: human detection, human body decomposition and feature measurement, feature selection, and human searching. The proposed mechanism can be used for searching the specific person and decrease the time wasted on matching dissimilar persons.

In human detection, we have used the modified frame differencing to detect the moving persons in videos. In some cases, the detected human may be fragmental. We use the edge information to fill the missing foreground. As the experimental results shown, the accuracy rate can reach 90%.

In human body decomposition and feature measurement, we segment the human body into three parts and measure the different features in the different parts. Our correct rate of human body decomposition can achieve above 85%. In the part of feature measurement, we describe the different features we want to test in different parts. These features are mainly color-based and texture-based. In different body parts, the features used are different.

We decompose a human body into three-parts, and use color-base particle filters to track each block separately.

After we detect and correct abnormal particle states, we can track the target normally.

5.2 Future Work

Use different features to track different parts. Change the number of particles dynamically according to the preceding state. Test the effects in different color spaces and state variables.

References

[1] H. Yang and M.D. Levine, "The Background Primal Sketch: an Approach for Tracking Moving Objects, " *Machine Vision and Applications*, vol. 5, pp. 17-34, 1992.

[2] C. Stauffer and W.E.L. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking," *Proceedings of the IEEE CS Conference on Computer Vision and Patter Recognition*, vol. 2, Fort Collins, Colorado, pp. 246-252, 1999.

[3] A. Elgammal, D. Harwood, and L. Davis, "Non-Parametric Model for Background Subtraction," *Proceedings of International Conference on Computer Vision*, Kerkyra, Greece, September 1999, pp 751-767.

[4] W. Hu, T. Tan, L. Wang, and S. Maybank, "A Survey on Visual Surveillance of Object Motion and Behaviors," *IEEE Transaction on System, Man, and Cybernetics—Part C: Applications and Reviews*, vol. 34, no. 3, Aug., 2004.

[5] K. Karmann and A. Brandt, "Moving Object Recognition Using an Adaptive Background Memory," *Time-Varying Image Processing and Moving Object Recognition* , vol. 2, 1990.

[6] M. Kilger, "A Shadow Handler in a Video-Based Real-Time Traffic Monitoring System," *Proc. IEEE Workshop Applications of Computer Vision*, pp. 11-18, 1992.

[7] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking Groups of People," *Comput. Vis. Image Understanding*, vol. 80, no. 1, pp. 42-56, 2000.

[8] A. Baumberg and D. C. Hogg, "Learning Deformable Models for Tracking the Human Body," *Motion-Based Recognition*, pp.39-60, 1996.

[9] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-Based Object Detection

in Images by Components," *IEEE Trans. Pattern Recognit. Machine Intell.*, vol. 23, pp. 349-361, Apr., 2001.

[10] A. Galata, N. Johnson, and D. Hogg, "Learning Variable-Length Markov Models of Behavior," *Comput. Vis. Image Understanding*, vol. 81, no. 3, pp. 398-413, 2001.

[11] Y. Wu and T. S. Huang, "A Co-Inference Approach to Robust Visual Tracking," *Proc. Int. Conf. Computer Vision*, vol. 2, pp. 26-33, 2001.

[12] N. Peterfreund, "Robust Tracking of Position and Velocity with Kalman Snakes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 564-569, June, 2000.

[13] C. A. Pau and A. Barber, "Traffic Sensor Using a Color Vision Method," *Proc. SPIE—Transportation Sensors and Controls: Collision Avoidance, Traffic Management, and ITS*, vol. 2902, pp. 156-165, 1996.

[14] B. Schiele, "Vodel-Free Tracking of Cars and Ppeople Based on Color Regions," *Proc. IEEE Int. Workshop Performance Evaluation of Tracking and Surveillance,* pp. 61-71, 2000.

[15] B. Coifman, D. Beymer, P. McLauchlan, and J. Malik, "A Real-Time Computer Vision System for Vehicle Tracking and Traffic Surveillance," *Transportation Res.*, vol. 6, no. 4, pp. 271-288, 1998.

[16] J. Malik and S. Russell, "Traffic Surveillance and Detection Technology Development," Univ. of California, 1996.

[17] T. J. Fan, G. Medioni, and G. Nevatia, "Recognizing 3-D Objects Using Surface Descriptions," *IEEE Trans. Pattern Recognit. Machine Intell.*, vol. 11, pp. 1140-1157, Nov., 1989.

[18] D. S. Jang and H. I. Choi, "Active Models for Tracking Moving Objects,"

Pattern Recognition, vol. 33, no. 7, pp. 1135-1146, 2000.