CHAPTER 1 INTRODUCTION

1.1 Motivation

In the modernized society, the surveillance system is properly used in many enterprises and organizations. There are many kinds of surveillance systems to choice for managements. In common environments, multiple cameras with fixed lens for human tracking are used frequently.

Traditionally, if an administrator wants to obtain information about a specific person, he has to explore from all video sequences step by step. The process of exploring costs an administrator much time and has no efficiency. We want to construct a system to organize video sequences and to store representative frames about people. When an administrator inputs a test pattern, the system can assist to fast matching and obtaining all video sequences a specific person.

The surveillance system contains useful models, like human-based image sequences, selection of key frames, human searching, and so on. Human-based image sequences record information that is time and position a person enters or leaves and representative frames in the environment. Selection of key frames preserves representative frames of a person in a single camera into archiving. These key frames record principal information, like sizes of people, significant transformations of structure and intensity of a person and appearances with a face. Human searching can obtain candidates for an input pattern quickly when an administrator wants to search a specific person. Hence, these models assist to reduce the workload and make the surveillance system intelligent.

1.2 Problem Definition

Problems that we want to solve in this study are listed as follows.

1.2.1 Obtaining of human-based image sequences in multiple cameras

Human-based image sequences are important for further processing in the surveillance system. How to obtain these in the environment? Can we exploit tracking in a single camera and identifying in multiple cameras to archive.

1.2.2 Obtaining of representative frames to assist matching for a specific person

Representative frames influence human matching significantly. If we can keep key and available frames from human-based image sequences about a person effectively, we can obtain better results in human matching.

1.2.3 Matching of humans in different appearances

Many factors, such as light effect, different views, and so on, could affect the matching result. We want to settle human matching in different appearances resulted from different views for increasing matching rates in human searching.

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.

1.3.3 Fast Search Algorithm for Movement Estimation

Motion estimation plays an important role in motion compensated image sequence coding. Given an image block in the reference frame, the motion estimation problem at hand is to determine a matching block in the target frame such that the error between these two blocks is minimized. The concept is the same for estimation of movements of a person in tracking. We can exploit fast search algorithms in motion estimation to assist tracking. Here, we use movement estimation in our topic replacing motion estimation for preventing confusion that motion is considered as behavior of a person.

In the early 1980s, some conventional fast motion estimation algorithms were proposed. Koga, et al. [19] propose the three-step search. Jain, et al. [20] propose the 2-D log search.

The 2-D log search describes a method of measuring inter-frame motion for digital images. It approximates the inter-frame motion by piecewise translation of one or more areas of a frame relative to a reference frame. It is an extension of the binary or logarithm search in one dimension.

The three-step search algorithm becomes one of the most popular algorithms, owing to its simplicity and effectiveness. It uses a uniformly allocated search pattern, 8 neighbors at boundaries of a search range centered at best matching position of last search step, in each search step.

But, the three-step search uses a uniformly allocated search pattern in its first step, which is not very efficient to catch small motions appearing in stationary blocks. Several adaptive techniques have been suggested to make the search more adaptable to motion scale and uncertainty for the three-step search.

R. Li, et al. [21] propose one new method, new three-step search algorithm. The new three-step search algorithm differs from three-step search algorithm by (1) assuming a center-biased checking point pattern in its first step and (2) incorporating a *halfway-stop* technique for stationary or qusi-stationary blocks.

L.-M. Po & W.-C. Ma [22] propose four-step search algorithm. Like the new three-step search algorithm, use center-biased checking and *halfway-stop* technique. Simulation results show that the method may have better performance than three-step search and similar performance to the new three-step search. In addition, the four-step search also reduces the worst-case and average computational requirements as compared with the new three-step search.

1.4 Assumptions

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

1. Surveillance is an indoor environment.
2. Light condition is stable.
3. Multiple cameras with fixed lens
4. Input images are gray-scale.

1.5 System Description

In this thesis, we develop a system to construct human-based image sequences, to record representative frames from human-based image sequences and to provide candidates in human searching. The system flow diagram is shown in Fig.1.5.1. The main modules of the system are described as follows:

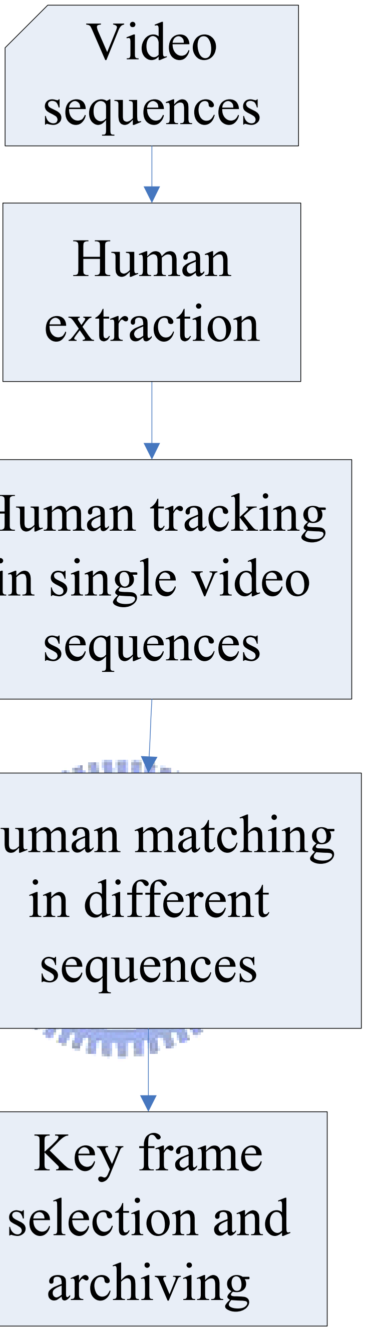
Video sequences
Human extraction
Human tracking in single video sequences
Human matching in different sequences
Key frame selection and archiving

Fig.1.5.1 The system flow diagram

1.5.1 Human Extraction

We construct a statistical background model that could segment the foreground region, humans, from the input image. The foreground region would be collected to form the connected components.

1.5.2 Human Tracking in a Single Video Sequence

For each camera, human tracking is implemented to catch movements of a person. If we can track people effectively, we hope to record walking paths of people to construct a map for a scene.

1.5.3 Human Matching in Different Sequences

After tracking in a single camera, we hope to collect image sequences containing some representative frames about a person in multiple cameras. So, we need human matching for frames in different sequences to connect image sequences of the person.

1.5.4 Key Frame Selection and Archiving

We propose three rules to select key frames according completeness, significant transformations in structure or intensity, and validity of movement directions about a person. From tracking results of a person in each camera, connect representative frames as a lot of image sequencing and record into archiving.

1.6 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 describes human tracking. Chapter 3 describes human matching. Chapter 4 describes archiving of human-based images. Chapter 5 describes experimental results and their analyses. Finally, chapter 6 presents some conclusions and suggestions for future works.

Chapter 2 Human Tracking

If we can predict locations of a person in multiple cameras, we can analyze the movement of the person more efficiently [23]. Human extraction and matching is described in section 2.1. By recording walking paths of people into a walking map, an element in walking map is a guide to instruct which directions are frequently used in a corresponding position of it. Then, we can judge whether a person is suspected by find whether his directions is common. Construction of the walking map is described in section 2.2.

2.1 Human Extraction and Matching

In this section, we describe tracking in a single camera. Firstly, human extraction is implemented in the initial frame of video sequences, discussed in section 2.1.1 human extraction. Secondly, we predict the location of a person in the next frame according to his current location *L* by matching him between the current and next frames. Candidate positions of the tracked person are located in a search region $(2R+1)^2$ centered on the current position *L*. We propose a method to estimate movement of the tracked person, discussed in section 2.1.2 quarter search algorithm. To achieve above, we has to determine the match result between a candidate position and *L*, discussed in section 2.1.3 feature matching.

2.1.1 Human Extraction

We describe the probability background model and the background subtraction approach to segment persons. Background subtraction segments moving regions in image sequences taken from a static camera by comparing each new frame to a model of the scene background. The range of intensity should be fixed for every pixel of the background, because the background is usually invariable under stable light condition. If the intensity of a pixel is rare to appear, the pixel should not belong to the background. Therefore, it is reasonable to use a statistical method to construct a background model.

For computing the probability, we have to get a sample of background. Assume the number of image in the sample is *N*. Let $x_1, x_2, \ldots, x_N$ be a sample of intensity values for a pixel.

The probability density function that this pixel will have intensity value x_t at time *t* can be non-parametrically estimated using the kernel estimator *K* as

$$Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N} K(x_t - x_i)$$

If we choose our kernel estimator function, *K*, to be a Normal function $N(0, \sum)$, where $\sum$ represents the kernel function bandwidth, then the density can be estimated as

$$Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} e^{-(x_t - x_i)\Sigma^{-1}(x_t - x_i)/2}$$

the density estimation is reduced to

$$Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_t - x_i)^2}{2}}$$

Using this probability estimate the pixel is considered a foreground pixel if $Pr(x_t) < th$ where the threshold *th* is a global threshold over all the image. Next we find the connected components of the pixel whose $Pr(x_t) < th$. If the size of a connected component is too small, it would not be considered to the foreground. The smaller connected components could be affected by light or noise. Fig.2.1.1.1 shows some examples.

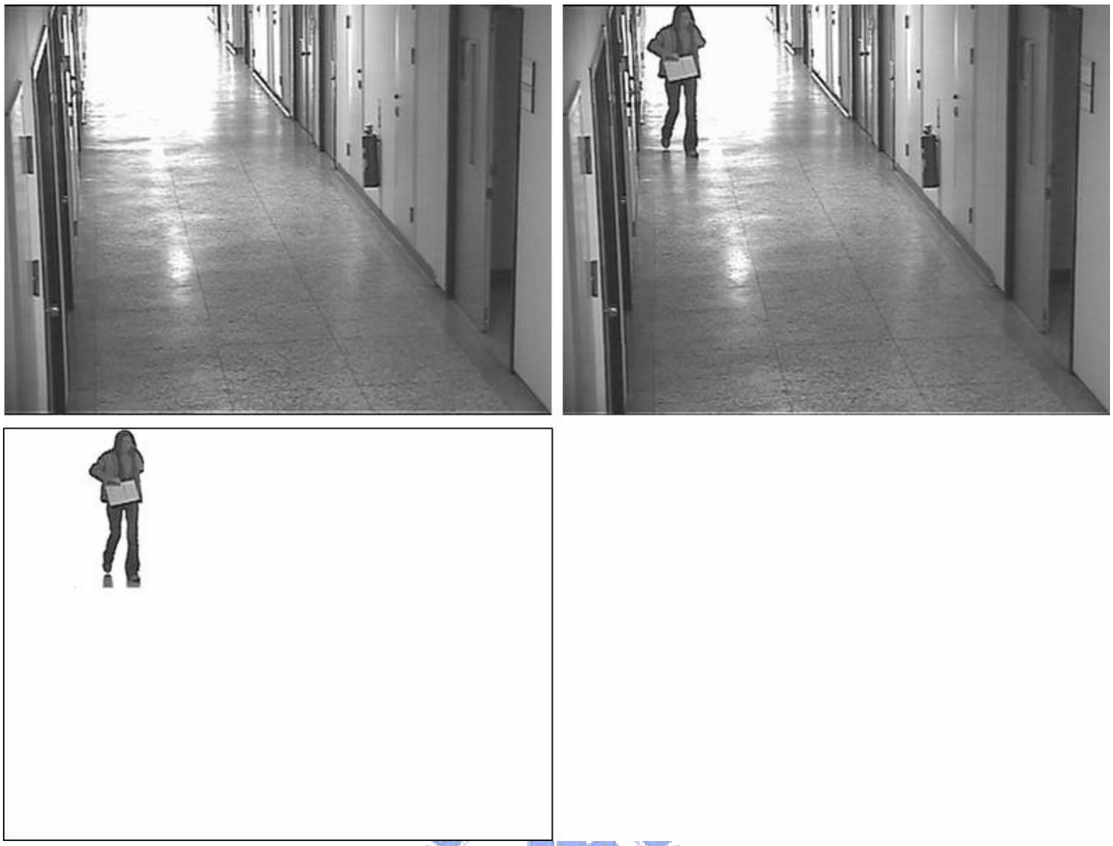

Fig. 2.1.1.1 (a)(b) (a) The background (b) The original image
(c) (c) Human extracted

2.1.2 Quarter Search Algorithm

How to estimate the movement of a person between frames? Common methods for movement estimation used frequently include three-step search algorithm, four-step search algorithm, 2-D log search algorithm, and so on. Suppose the search region is –*R* to *R*. They reduce the number of search candidates to $c \cdot \log R$ from $(2R+1)^2$. A size of an image is 640 x 480 in our experiments. The maximum of *R* is 319; log *R* is 9 at most. The magnitude of constant *c* becomes important.

We propose a new method called quarter search algorithm obtaining a lower *c* than that of the three-step search algorithm.

Suppose the bounding rectangle of a person in a previous frame has size *N* x *N*.

$$E(u,v)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|f_{t-1}(i+u,j+v)-f_t(i,j)\right| \qquad -R\le u,v\le R$$

where (*u*,*v*) is the candidate movement vector, and $f_{t-1}(\cdot,\cdot)$ and $f_t(\cdot,\cdot)$ refer to the bounding rectangles in a previous frame and the current frame that are to be compared. For minimizing *E*(*u*,*v*), how many candidate movement vectors do we need? The key to reducing the computation is reducing the number of candidate movement vectors In hence, we want to decrease candidate movement vectors for speed.

Let we see the concept of the three-step search algorithm. It has an idea that eight search points at the boundaries of the search area may be possible for walking in searching of each layer and coarse-to-fine with multi-layer. An example for three-step search algorithm is in the figure 2.1.2.1. It’s easy to comprehend and implement. However, it may be inefficient to waste time to check unnecessary search points. The idea of our method is that we hope to obtain a quarter that a person moves to most possibly in each layer from the search area. The quarter include four points: the central search point and three search points produced by adding the central search point with horizontal, vertical, and corner vectors, respectively. If we obtain the quarter in searching of each layer, we can comprehend that the person may move with the direction composed of three vectors.

How to obtain a quarter from a search area?

Firstly, we obtain two better matching vectors from four candidate movement vectors located on the vertical and horizontal search area boundaries, respectively. Secondly, the vector between horizontal and vertical better matching vectors is the other we want. Finally, the central point and three points produced by adding the central point with three vectors above are representation of a quarter.

With the concept above, we propose a method. The procedure is as follows:

1. The search starts with a step size equal to half of the maximum search range.
2. In each step, six search points are compared.
   (1) Two better matching points from four search points located on the vertical and horizontal search area boundaries, respectively.
   (2) The point at the corner of these two better matching points
   (3) The central point of the search square
3. The step size is reduced by half after each step, and the search ends with a step size of 1 pixel.
4. At each new step, the search center is moved to the best matching point resulting from the previous step.

An example of the quarter search algorithm is in the figure 2.1.2.2.

Let R represent the size of the search range and R_0 represent the initial search step size; then there are at most $L = \lfloor \log_2 R_0 + 1 \rfloor$ search steps. If $R_0=R/2$, then $L = \lfloor \log_2 R \rfloor$. At each search step, five points are searched, except in the beginning, when six points must be examined. The total number of search points is $5L+1$.

Our algorithm has higher speed and almost the same tracking result as that of the three-step search algorithm. Table 2.1.2.1 shows time complexity of the three-step search algorithm and quarter search algorithm. Table 2.1.2.2 shows time of the three-step search algorithm and quarter search algorithm take, respectively when we input test the same video sequences.

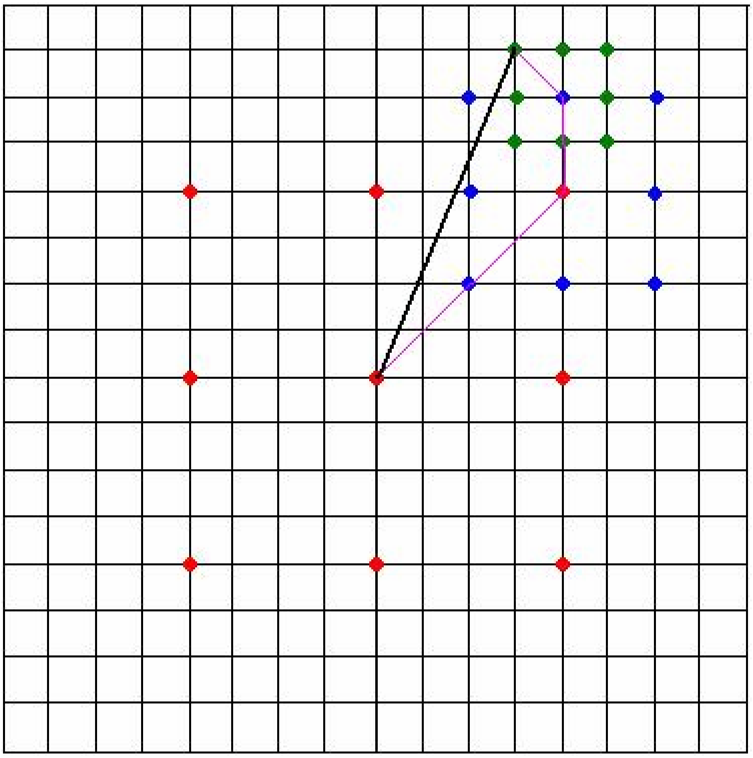

Fig. 2.1.2.1 An example for the three-step search algorithm.

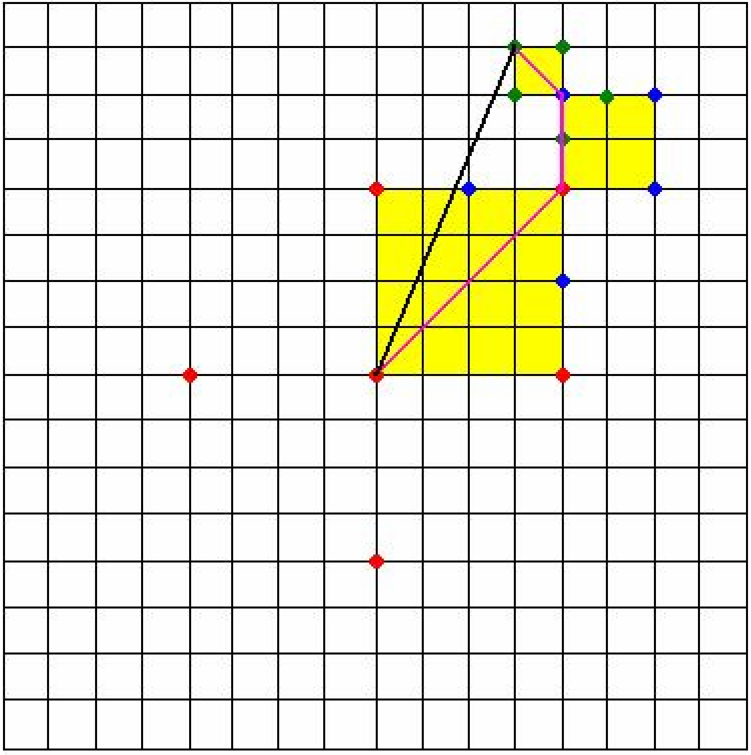

Fig. 2.1.2.2 An example of the quarter search algorithm

	Three-step search algorithm	**Quarter search algorithm**
Time complexity of all search steps	$8L+1$	$5L+1$

Table 2.1.2.1 Time complexity

	Time of quarter search (sec.)	**Time of three-step search (sec.)**
1st test image sequence (17 frames)	**0.375**	**0.594**
2nd test image sequence (84 frames)	**6.358**	**7.529**
3rd test image sequence (31 frames)	**0.873**	**1.861**
4th test image sequence (31 frames)	**1.752**	**2.782**
5th test image sequence (18 frames)	**0.389**	**0.814**
6th test image sequence (55 frames)	**2.5**	**6.371**
7th test image sequence (26 frames)	**1.311**	**1.811**
8th test image sequence (22 frames)	**1.267**	**1.376**
9th test image sequence (26 frames)	**0.673**	**1.188**
10th test image sequence (35 frames)	**0.77**	**1.343**
11th test image sequence (110 frames)	**3.476**	**5.09**
12th test image sequence (73 frames)	**2.612**	**4.316**
13th test image sequence (65 frames)	**2.936**	**4.608**
14th test image sequence (71 frames)	**2.03**	**2.988**
15th test image sequence (89 frames)	**2.476**	**5.142**
16th test image sequence (61 frames)	**2.487**	**3.814**
Average (50 frames)	**2.018**	**3.231**

Table 2.1.2.2 Comparison of computing time

2.1.3 Feature matching

Intensity values may not change rapidly between two successive frames. We use the property to match the bounding rectangle of a person in the current frame with that in candidate positions of the next frame. We exploit intensity values of a person as our main feature for matching. Through feature matching, we hope to obtain the best matching position.

We normalize two bounding rectangles in two successive frames to prevent from size variation. For efficiency, we separate a bounding rectangle into several blocks and use block matching. Then, the feature becomes average intensity value in each block. The function of feature matching is the sum of feature differences.

The figure 2.1.3.1 illustrates above procedures.

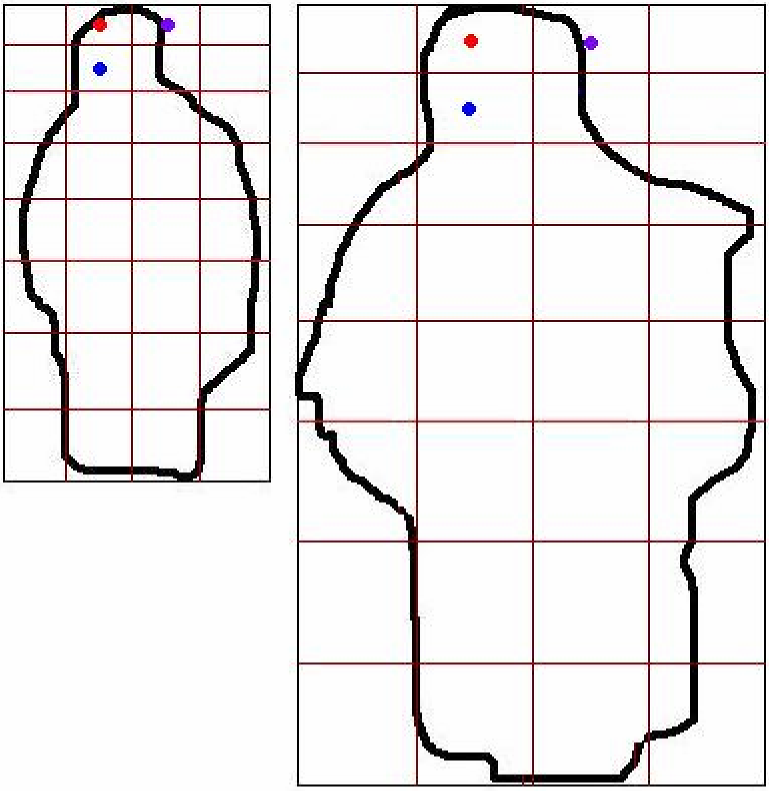

Fig. 2.1.3.1 A diagram for feature matching

Finally, we show an example of tracking results as follows. In the figure 2.1.1, the up row is original images at different time and the low row is tracking results. And, the figure 2.1.2 is similar as the figure 2.1.1.

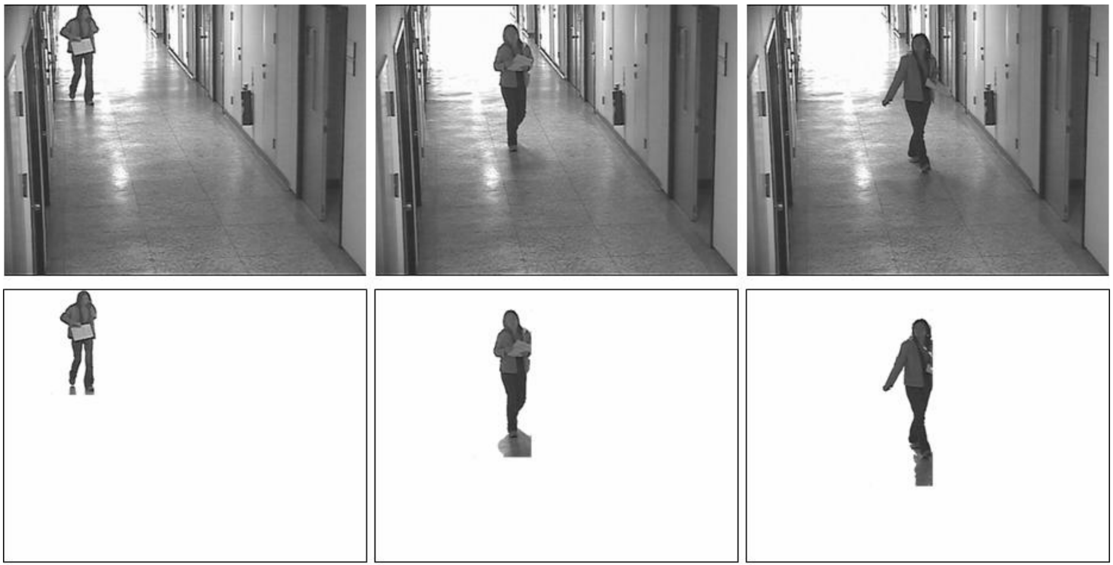

Fig. 2.1.1 (a1)(b1)(c1) Original images at (a) frame 11 (c) frame 17 (e) frame 20
(a2)(b2)(c2) Tracking results at (b) frame 11 (d) frame 17 (f) frame 20

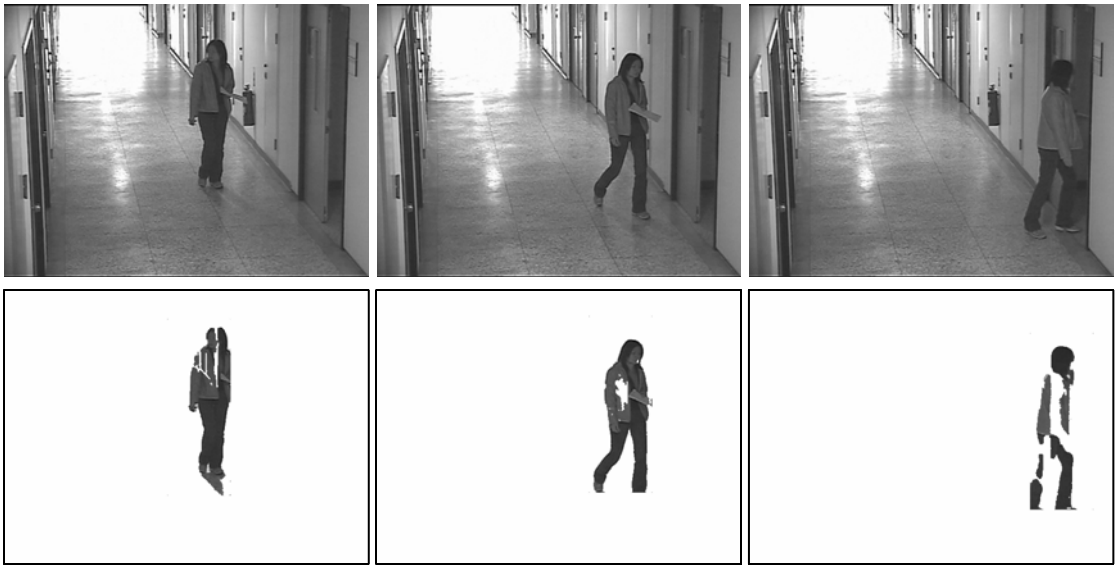

Fig. 2.1.1 (d1)(e1)(f1) Original images at (a) frame 23 (c) frame 26 (e) frame 29
(d2)(e2)(f2) Tracking results at (b) frame 23 (d) frame 26 (f) frame 29

2.2 Construction of the Walking Map

The movement of a person is important information for monitoring the person. When a person walks in an uncommon direction, we hope that the surveillance system may issue a warning signal and records corresponding frames about a person. How to determine an uncommon or strange direction? We hope to exploit walking of people to construct a map recording walking paths. We consider that possible walking paths are limited in a certain range. Because the process of recording walking paths counts directions people walks in at locations, some locations may be passed through frequently in several directions and others may be passed through infrequently. Hence, an element in it represents the probability that people go through the map location in a specific direction.

A walking map is obtained from representation of walking paths and direction recording in training sequences.

I. Representation of walking paths

Because directions a person walks in are too many to assist analyzing, walking directions are divided into 16 types. Represent each walking direction of a person in a camera by one of 16 directions. An example is in fig. 2.2.1.

II. Direction recording

After tracking, we record directions in locations where a person passed through. An example is in fig. 2.2.2. From training samples, we can obtain the frequency of each of 16 directions on each position.

By analyzing directions that a person walks in at the scene according to walking map, we can determine whether a person walks in a strange direction. Then, the walking map assists to obtain strange or uncommon direction in corresponding frames of image sequences.

Now, if there is a direction of a person with low probability in a location according to the walking map of the scene, we may consider that the direction is strange. For example, we would not accept a direction from a window or the outside of a wall to a house. Further, the corresponding frame where a strange or uncommon direction occurs may assist to comprehend actions and movement of a person.

The procedure to determine whether a direction is strange or uncommon is as follows:

1. Find the direction *direc.* a person moves with at a location *L* from time frame *t-1* and *t*.
2. Direction check
   - According to the location where the person is in the walking map, he may go with several directions D_j having higher probabilities in common cases.
   - Compare *direc.* with D_j in common cases.
3. Issue a warning signal if *direc.* is not accepted at *L*.

Then, if a person walks with a strange direction, we can discover that and process further. In hence, we can promote the security of a surveillance system based on the analysis of a direction further.

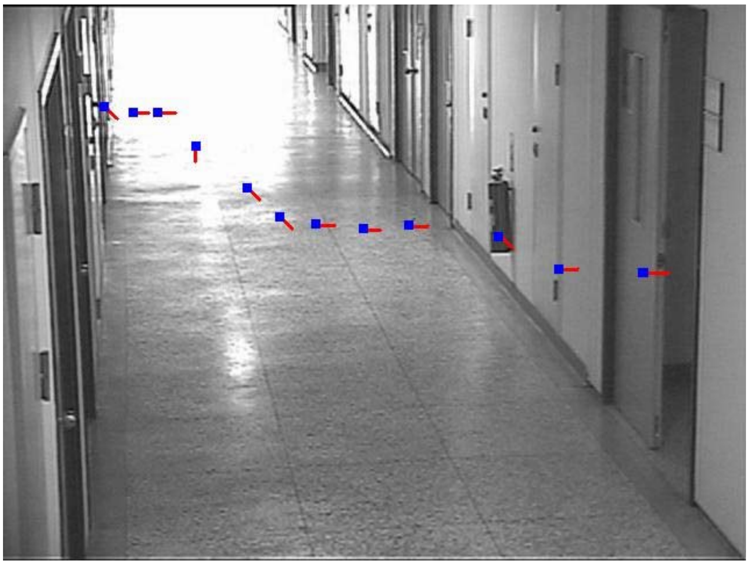

Fig. 2.2.1 Representation of a walking path

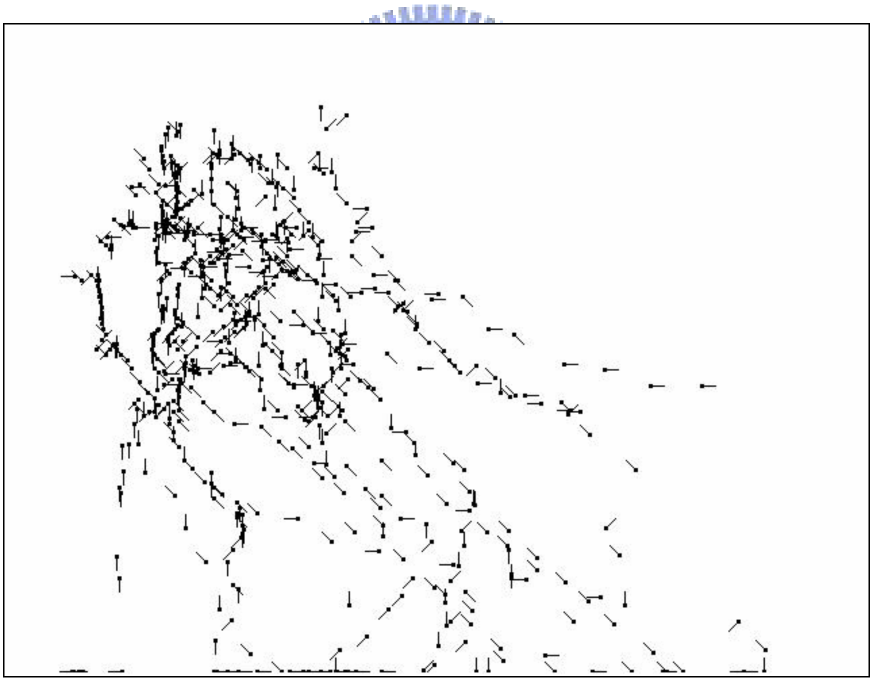

Fig. 2.2.2 A directional map

CHAPTER 3 HUMAN MATCHING

In this chapter, we would introduce appearance-independent human matching in the section 3.1 and two functions in our proposed mechanism for application of appearance-independent human matching in the section 3.2

3.1 Appearance-Independent Human Matching

Why the surveillance system needs human matching?

Firstly, when a person is monitored in the surveillance system, there are two cases needing human matching. The surveillance system needs human matching to identify a person to connect image sequences about him in multiple cameras for human-base when he walks in boundaries of two scenes or he occluded by other objects might reappear.

Secondly, when an administrator wants to search a specific person, the surveillance system needs human matching to provide candidate persons about him.

Two problems in human matching:

(1) Different appearances

Images of a person captured by different cameras usually have different appearances resulting from different capturing views of cameras. It results in difficulty of human matching.

(2) Different human sizes

Sizes of a person captured by different cameras are usually different.

How to solve these effectively in human matching? Krumm et al. [24] use color histograms to match regions. In general, the methods for object matching need camera calibration. However, Javed et al. [25] also develop methods without calibration.

We propose a method using color and relative smoothness features of a person for matching in different appearances resulting from different views.

Why not we use the edge, gradient, or the curvature features? Because of different appearances of a person, using corresponding shape information assists human matching hardly.

In hence, we exploit the color and relative smoothness features of a person. The relative smoothness feature is a measure of gray-level contrast that can be used to establish descriptors of relative smoothness.

Although views captured by different cameras result in different appearances of a person, appearances of a person in different views still have similar features. We can consider that several appearances of a person captured by several cameras seemed to several appearances resulting from horizontal turning of a person captured by a camera. In other words, parts of a human body may turn horizontally among different appearances. Such as fig. 3.1, red parts of (a) and (b) are similar. It seems that the red part of (a) turns horizontally to that of (b).

For matching in different appearances, the bounding rectangle of a human body is divided into a fixed number of *m***n* blocks for size normalization. Next, a human body is represented by three parts: a head (20% of body), an upper body (40% of body), and a lower body (40% of body). We would assign the highest weight to blocks in the head part because a head of a person is distinct from others. And, weight values in blocks in an upper and a lower body are secondly and thirdly high, respectively.

Features we exploit for matching are intensity and relative smoothness. Intensity differentiates people between distinct intensity effectively. However, when two persons have similar appearances, intensity is not good at distinguishing. We use another feature, relative smoothness, to assist to distinguish persons by exploiting that their texture may be different.

Let z be a random variable denoting gray levels in the block b and let $p(z_i)$, i=0,1,…,L-1 be the corresponding distribution, where L is the number of distinct gray levels.

m is the mean value of z and σ^2(z) is the variance. Relative smoothness R, measurement about texture information, is computed from σ^2(z). We use a simple measurement to compute R for simpleness of comprehension.

$$m = \sum_{i=0}^{L-1} z_i p(z_i)$$

$$\sigma^2(z) = \sum_{i=0}^{L-1} (z_i - m)^2 p(z_i)$$

$$R = 1 - \frac{1}{1+\sigma^2(z)}$$

After handling of different human sizes and decision of matching features, the matching procedure can start. We describe as follows.

Firstly, features of each block in the reference frame are compared with those in the same row of the matched frame, such as Fig. 3.2.

Secondly, a weight value *W* according to the position of a block in a human body is used to control feature differences according to the location of a block in one of three regions.

Thirdly, the function of this matching is weighted sum of feature differences.

A function describes above three processes as follows:

$$\sum_{i}\sum_{j} W * \min_{P} \left\| R(i,j) - C(i,P) \right\|$$

where (*i*,*j*) is the coordinate representing a block in a bounding rectangle of a person, *R*(‧ , ‧) and *C*(‧ , ‧) refer to the bounding rectangles of people in a reference frame and a compared frame, and *P* is the corresponding same row for each *j* in the compared frame. We discover the best matching block with the minimal feature difference in the corresponding row *P* for each (*i*,*j*). And, a weight value *W* is added to control feature differences for each block.

Finally, from compared frames, choose a person with the minimum sum of feature differences with the person in the reference frame.

In conclusion, our method prevents human matching from being effected by different appearances. This can assist the surveillance system to cope with human matching effectively robustly.

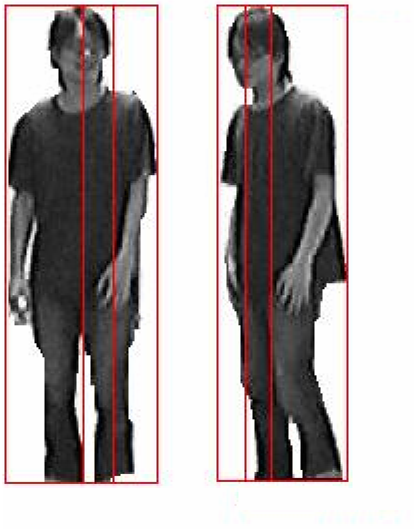

Fig. 3.1 (a) front view (b) side view

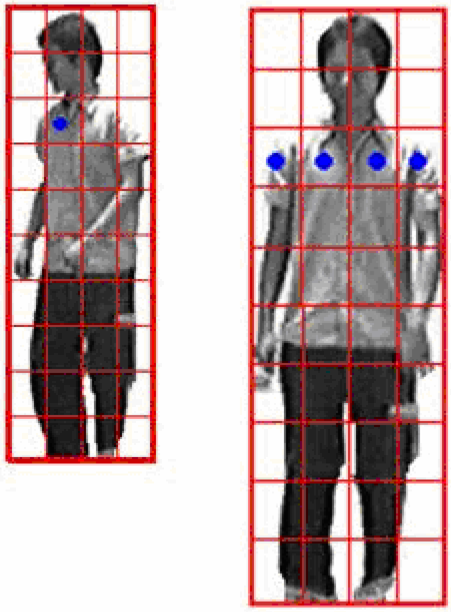

Fig. 3.2 An example of appearance-independent matching

3.2 Application

Appearance-independent human matching can be applied to:

1. Collection of image sequences containing representative frames for a person
2. Human Searching when an administrator wants to obtain corresponding images of a specific person

For a surveillance system, we exploit appearance-independent human matching to construct a proposed mechanism providing two functions:

Function 1: Matching representative frames of a tracked person with those of other tracked people

Function 1 called human matching in multiple cameras in our mechanism.

Function 2: Human searching from several representative frames of people when an administrator wants to obtain a specific person

Function 2 called human searching in archiving in our mechanism.

Functions of our proposed mechanism are discussed in the section 4 later.

CHAPTER 4 ARCHING OF HUMAN-BASED IMAGES

For human-based image sequencing:

1. Obtain key frames of a person in each camera during a time period.

   · We want to obtain key frames from tracking results for a person.

2. Connection of key frames of a person in a single camera and multiple cameras.

   · A person in a single camera or multiple cameras is tracked and key frames are recorded into image sequences for him during tracking. We hope to connect corresponding image sequences about him.

If we can construct human-based image sequences by recording and connecting key frames for each person, a person searched represents all corresponding image sequences obtained in human matching. So, collection of key frames with respect to a person can facilitate the searching of the person in a large data base efficiently.

Problems here:

1. How to select key frames of a person in an image sequence?

   · This effects storage of human-based image sequences and matching rates of human matching much.

2. How to collect key frames in multiple cameras for a person?

   · Combining key frames of a person in each camera needs human matching to determine whether people in these key frames are the same.

   · Then, if we obtain a key frame of a person, other key frames of him can be obtained.

4.1 Selection of Key Frames

I. Definitions of a key frame

Key frames are important information retrieve for a specific person querying and achieve data reducing for archiving. In hence, selection of key frames affects correctness rate of appearance-independent matching significantly.

For a sequence of frames with respect to a specific person, key frames are defined as follows:

1. the first and the last frames in which the size of the bounding rectangle of the person is greater than a threshold,
2. the frame in which the person’s appearance is different sufficiently from the appearance of the previous key frame, and
3. the frame in which the person’s average intensity value is different from that of the previous key frame.

II. Representation of a key frame

1. Sizes S of the bounding rectangle of a person
2. Aspect ratio R of the bounding rectangle
   - We prefer the aspect ratio between 0.25 and 0.55.
3. The average intensity value
4. The body appearance
5. Appearance of a face

III. Details of a key frame

In sizes *S*, confirm to obtain enough and clear resolution for a person. We can comprehend sizes according to locations of him in the scene.

In aspect ratio, confirm the bounding rectangle of a person is complete.

In the average intensity value, feature matching is exploited to match the previous key frame with tracking frames later. The frame in which the person’s average intensity is different sufficiently from that of the previous key frame is the current key frame. So, we can keep significant changes of intensity of a person based on average intensity. The figure 4.1.1 illustrates an example that a person has significant changes in average intensity.

In the body appearance, feature matching is also exploited to match the previous key frame with tracking frames later. The frame in which the person’s partial appearance is different sufficiently from the appearance of the previous key frame is the current key frame. The figure 4.1.2 illustrates an example that frames in which a person has different appearances are recorded when he walks in the scene. In the figure 4.1.2, red parts between (a) and (b) are different sufficiently.

The procedure determining significant changes of average intensity and structure of a person is described as follows.

(1) Let the first key frame f_1 suiting sizes and aspect ratio about a person.

(2) If there is a frame f_i in which appearances or average intensity values of a person are obviously different from those of a person in the previous key frame through feature matching with the previous key frame and f_i, the frame f_i is the current key frame.

(3) After tracking in a single camera, several key frames with significant transformation of structure or average intensity of a person are obtained.

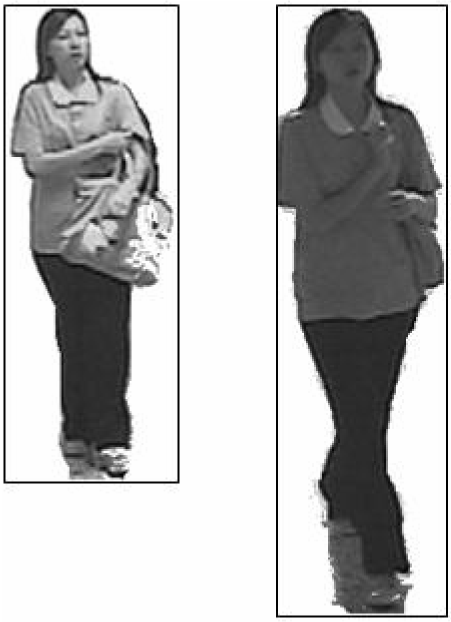

Fig. 4.1.1(a)(b)

An example of key frames with significantly different average intensity values between (a) frame 13 and (b) frame 28

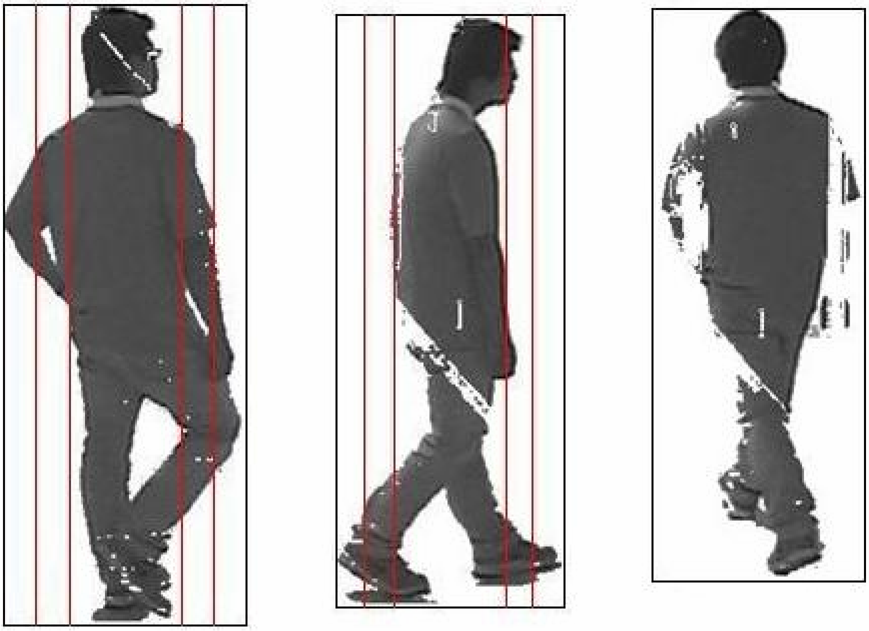

Fig. 4.1.2 (a)(b)(c)

An example of key frames with significantly different appearances between (a) frame 10 and (b) frame 15 and
between (b) frame 15 and (c) frame 28, respectively

In appearance with a face, assist to obtain images in which a person is front view or side view as far as possible.

Information of a head of a person is exploited to distinguish between back view and other view [26-27]. Suppose a head is probably located on 20% of the bounding rectangle of a person. Use two features to distinguish between a face and the back of a head: intensity and luminance regularity.

Exploit differences of intensity and luminance regularity between the face and the back of a head.

- Intensity of the back of a head is darker than that of a face.
- The luminance of the face is non-uniform throughout.
  - Light is reflected off the curvature of the face and facial features at different angles.
  - The back of a head has more even brightness distribution.

The frame in which the person’s facial components are more than that of the previous key frame is the current key frame.

IV. Format of human-based image sequencing

For a person in a single camera C_k , what should be recorded?

- Entering time t_{start} of the first key frame and leaving time t_{end} of the last key frame
- An entering position p_{start} of the first key frame and a leaving position p_{end} of the last key frame
- Key frames f_i

The format is as follows.

$$S_{C_k} :< t_{start}, t_{end}, p_{start}, p_{end}, f_{t_{start}}, \ldots, f_{t_{end}} >$$

In our collected samples, each image sequencing for a person in a single camera contains about four or five key frames averagely.

4.2 Functions

We have briefly discussed two functions in our proposed mechanism is the section 3.2. Here, we detail these two functions.

I. Human Matching in Multiple Cameras

When a person walks in the surveillance system, connection of image sequences of him in multiple cameras is needed. If we can achieve this, human-based image sequences for him can be constructed.

How to connect image sequencing containing key frames of a person in multiple cameras? Key frames of the person in a single camera are compared with those of people else where based on appearance-independent human matching. Key frames with similar appearances are collected. Then, connection of key frames is achieved. According to sorting of entering and leaving time of image sequencing, a lot of image sequencing containing key frames can be organized.

Based on above, a lot of image sequencing containing key frames of a person in each camera can be connected in the surveillance system. We can also comprehend entering and leaving time or positions of a person from image sequencing. All of these usually assist in further processing.

II. Human Searching in Archiving

Every several hours, a lot of image sequencing of tracked people in multiple cameras is recorded into archiving based on appearances. It’s to prevent storages of archiving are too many to implement human matching efficiently.

For human searching, exploit appearances-independent human matching to obtain corresponding key frames in archiving for a specific person. Because we can confirmed that a lot of image sequencing containing key frames of a person can be organized through function 1 of our mechanism, one key frame can obtain other key frames for a specific person based on the same collection. In hence, we can increase matching rates of human searching in archiving significantly.

Performance of function 1 and 2 in the mechanism we proposed will be discussed thoroughly in experiment.

CHAPTER 5 EXPERIMENT RESULT AND DISCUSSION

In this chapter, we present the experiment results and the discussion of our system. The proposed approach has been implemented on the CPU, Pentium IV 2GHz, with 512 MB RAM. The software environment is: (1) Microsoft Windows XP (2) Microsoft Visual C++ 6.0

The input images are gray scale, whose sizes are 640 x 480. We separate our experiment into three parts: tracking, human matching in multiple cameras, and human matching in archiving.

5.1 Experiment Result

5.1.1 Human Tracking

There 50 sets of image frames are tracked. 10 sets of them have occlusion.

For each frame: Correct tracking, if a person can be tracked
Error tracking, otherwise

Table 5.1.1.1 shows tracking result. Despite occlusion, our tracking is good at tracking in general case. If serious occlusion occurs for a person, it may influence tracking much and decrease rates of correct tracking. However, we can exploit human matching to assist to confirm continuity of tracking. Fig. 5.1.1.1 and 5.1.1.2 show examples of human tracking in real life.

	# of frames	# of frames with occlusion	# of frames detected correctly	# of frames detected wrongly	Correctness rates
Sets without occlusion (40 sets)	**1482**	**0**	**1454**	**28**	**98.11%**
Sets with occlusion (10 sets)	**413**	**75**	**394**	**19**	**95.40%**

Table 5.1.1.1 Tracking results

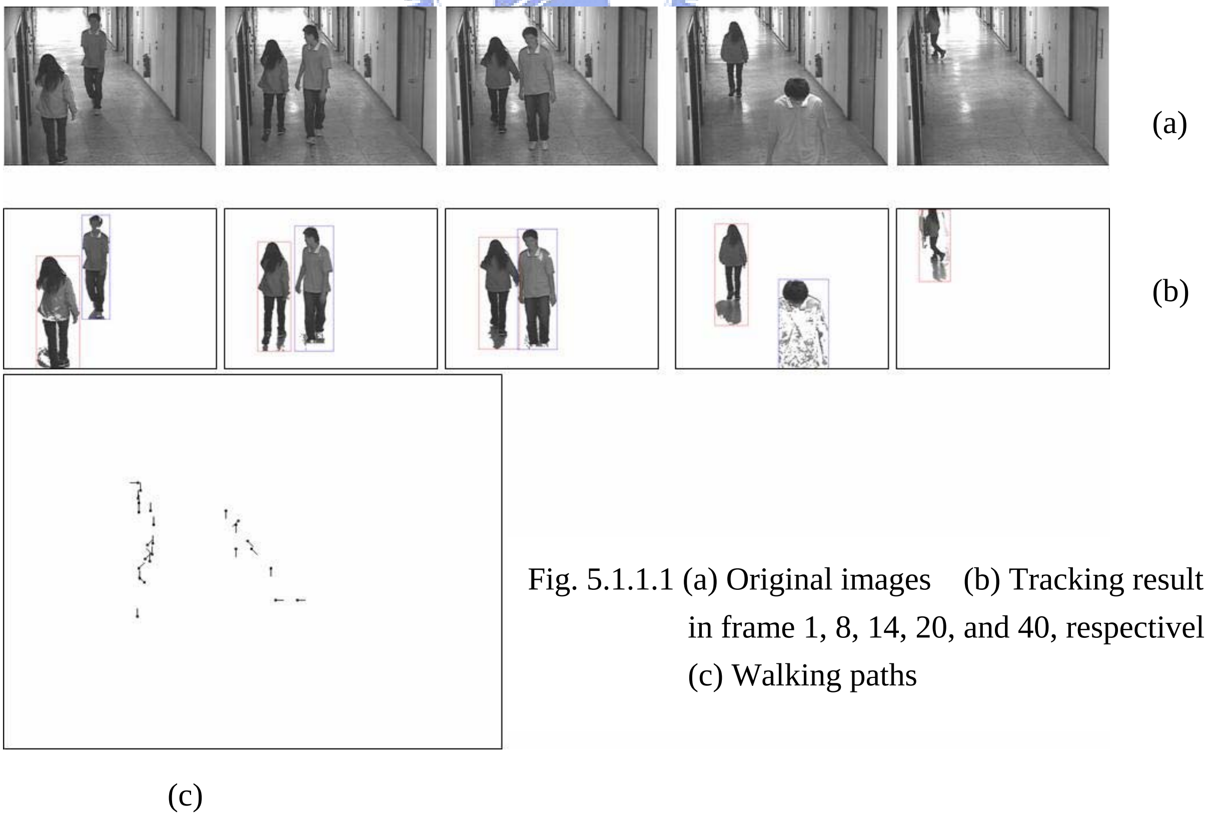

Fig. 5.1.1.1 (a) Original images (b) Tracking results in frame 1, 8, 14, 20, and 40, respectively (c) Walking paths

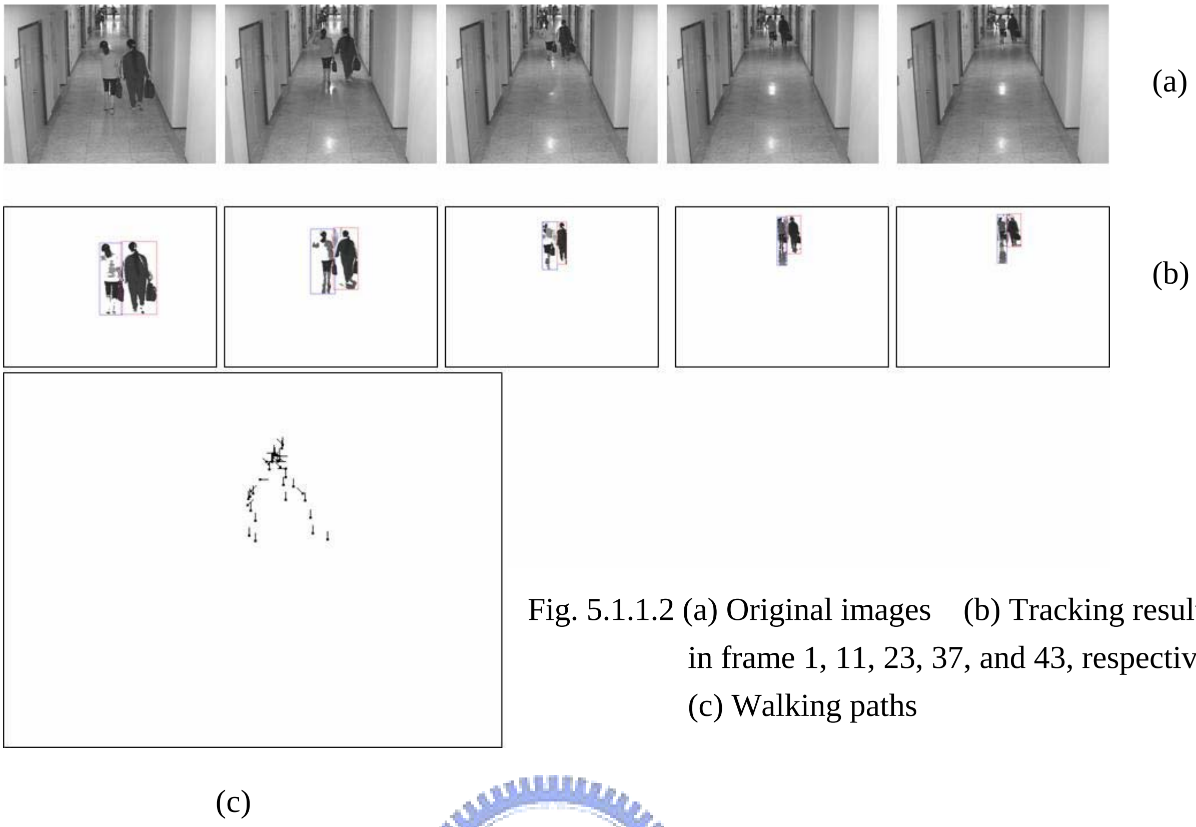

Fig. 5.1.1.2 (a) Original images (b) Tracking results in frame 1, 11, 23, 37, and 43, respectively (c) Walking paths

5.1.2 Human Matching in Multiple Cameras

We test 178 key frames of 42 people in real cases. Three views of these 42 people are shown in figures 5.1.2.1, 5.1.2.2, and 5.1.2.3, respectively. Clearer images of each person are shown from 178 key frames for illustrating the appearance of each person. However, non-clear images exist in 178 key frames for a specific person. Suppose a test person enters a scene a camera C_k captures. After the person passed through the scene, we need to decide who he is. This is known from discussion in section 3.

Our design procedure for this in experiment: Firstly, obtain humans H_c captured by neighbors of C_k. Secondly, compute degree of similarity with H_c for the person. Thirdly, obtain the best candidate according to the highest degree of similarity.

Then, we would decide whether the best candidate in our design procedure is correct matching for the person.

Correcting Matching

1. The case: The test person comes from a neighbor of C_k.

   Result: The candidate is truly himself.

2. The case: The test person does not come from a neighbor of or changes his appearance much in multiple cameras.

   Result: Cannot obtain the candidate for the test person.

Matching Rate		Comparison		
		front view	side view	back view
Reference	front view	97.44% (38/39)	97.37% (37/38)	92.31% (36/39)
	side view	95% (19/20)	95.24% (20/21)	95% (19/20)
	back view	91.67% (22/24)	91.67% (22/24)	96% (24/25)

Table 5.1.2.1 Matching rates in multiple cameras

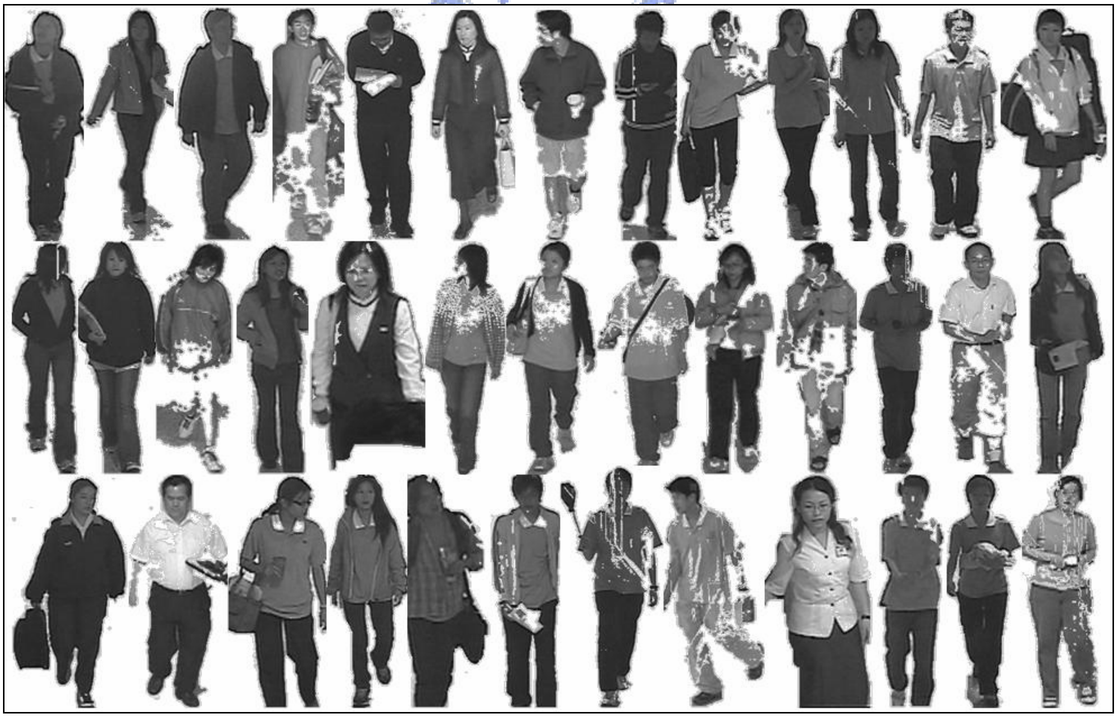

Fig. 5.1.2.1 front view

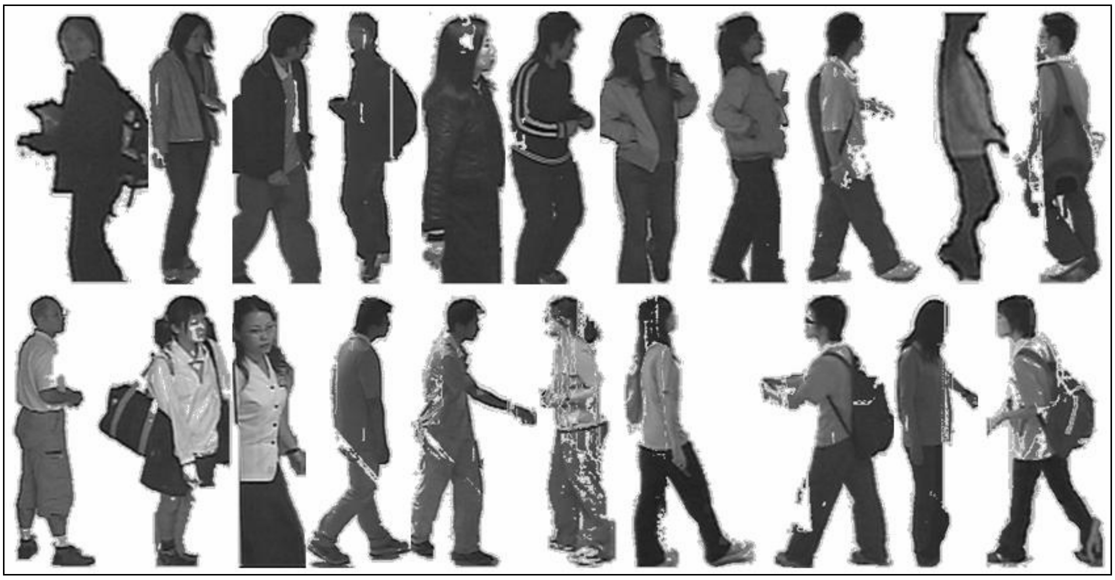

Fig. 5.1.2.2 side view

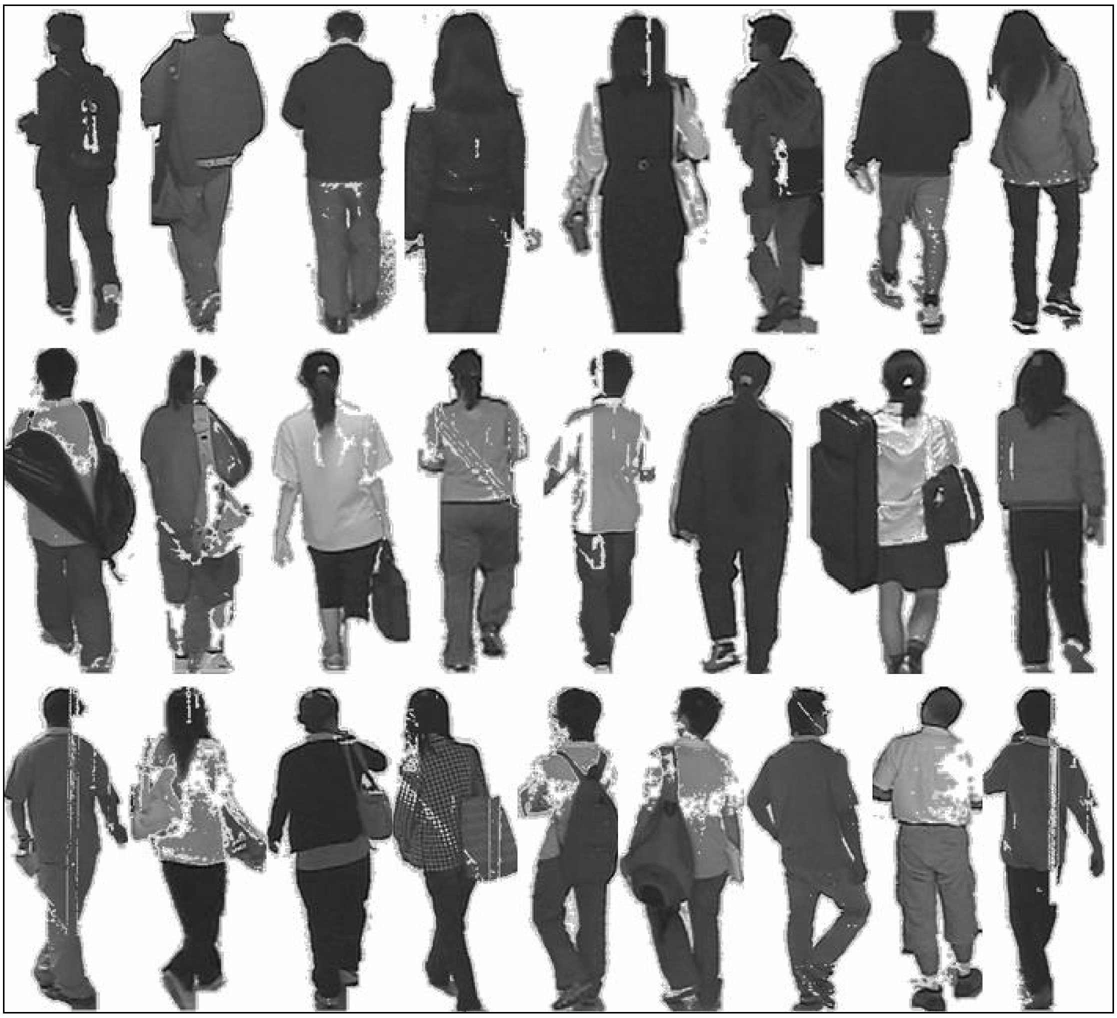

Fig. 5.1.2.3 back view

5.1.3 Human Matching in Archiving

We collect 195 key frames of 42 people from real tracking. Archiving in experiment is composed of 195 key frames. If we want to obtain a specific person from archiving, we will list several candidates for him.

We consider how many candidates are enough to hit the specific person? Too many candidates will result in much time to retrieve of right hits for an administrator. Evaluate performance of the appearance-independent matching by the number of candidates used. If the number of candidates is less and frames hit in archiving are as many as possible, we think performance of the matching method is good.

We suppose there are 25 candidates at most. How many hits for the specific person in 5, 10, 15, 20, 25 candidates, respectively? We use two measurements precision and recall to evaluate.

Precision: # of True / # of candidates

Recall: # of True / # of frames of a person in archiving

Precision represents how many percent of candidates are truly the specific person in matching. Recall represents how many percent of frames in archiving can be obtained for the specific person in matching.

Each frame in archiving is inputted for matching. The result is shown as table 5.1.3.1. We can see that 88.08% of frames in archiving can be obtained for each person when there are 25 candidates. However, lower rate of precision is normal. When the number of frames of a person in archiving is much lower than the number of candidates, precision is lowered. Despite the number of candidates, two right persons are hit for each person in this experiment. We can comprehend the average number of persons hit is two for each person.

From the table 5.1.3.1, matching rates are not much higher. We hope to achieve much higher matching rates. We analyze matching in archiving further. According to the view of a person, all key frames are classified to three views that are front view, side view, and back view.

1. Front view: 39 people (83 frames)
2. Side view: 21 people (45 frames)
3. Back view: 25 people (67 frames)

Matching Rates between Views

We start to compute matching rates between views for each frame in archiving.

Firstly, a frame of a test person is compared with all key frames in each view through appearance-independent human matching. Compute degree of similarity with people in each view for a test human.

Secondly, obtain several better candidates according to higher degree of similarity with the person.

Finally, correct matching

Case 1: Frames of the test person exist in a view compared

Result: The test person is truly in candidates.

Case 2: Frames of the test person do not exist in a view compared.

Result: Cannot obtain candidates for the test person.

Table 5.1.3.2 shows matching result between views for each frame in archiving. We can discover human matching between front view and back view has lower matching rates. However, human matching between front view and back view has good matching rates. The case that all key frames of a person are back view is few.

In hence, we suppose a person would not be captured in back view only in the surveillance system. Exploit front view and side view of people to implement experiment of human matching in archiving again.

Table 5.1.3.3 shows matching result after improvement. Recall is increased much. We can discover recall is almost 90% when the number of candidates is 10. Recall is more than 90% when the number of candidates is more than 15. Recall is almost 100% when the number of candidates is 25. Performance of appearance-independent human matching is excellent for all frames we collected in archiving.

	Precision	Recall
05 candidates	**35.52%**	**47.97%**
10 candidates	**25.76%**	**70.54%**
15 candidates	**19.36%**	**78.23%**
20 candidates	**16.12%**	**85.18%**
25 candidates	**13.54%**	**88.08%**

Table 5.1.3.1 Experiment result
of human matching in archiving

Matching Rate		Comparison			average
		front view	side view	back view	
Reference	front view	**97.44%** **(38/39)**	**97.37%** **(37/38)**	**79.49%** **(31/39)**	**91.43%**
	side view	**85%** **(17/20)**	**95.24%** **(20/21)**	**95.24%** **(20/21)**	**91.83%**
	back view	**75%** **(18/24)**	**92%** **(23/25)**	**96%** **(24/25)**	**88.67%**

Table 5.1.3.2 Matching rates between views

	Precision	Recall
05 candidates	**33.48%**	**71.62%**
10 candidates	**21.46%**	**88.43%**
15 candidates	**15.51%**	**93.81%**
20 candidates	**11.85%**	**95.31%**
25 candidates	**9.75%**	**98.69%**

Table 5.1.3.3 Experiment result of human matching
in archiving after improvement

5.2 Analysis of Error Results

5.2.1 Small sizes of people

A person is too small to track correctly. We cannot track effectively in our experiment if the resolution of a person is smaller than 20 x 50

5.2.2 Occlusion

I. Occlusion among different moving people

Accuracy of tracking result may be influenced much when the other moving people approach. Suppose a person to be our tracking target. If he was not completely occluded, the tracking target could not become other people. An example is in the figure 5.2.1. However, the tracking target will become other people if information of our tracking is less and less relatively. An example is in the figure 5.2.2.

II. Occlusion between people and the background.

Observe the scene to obtain locations occluding people easily. Keep on tracking by employing human matching in the neighborhood of these locations later. We can confirm little miss of tracking.

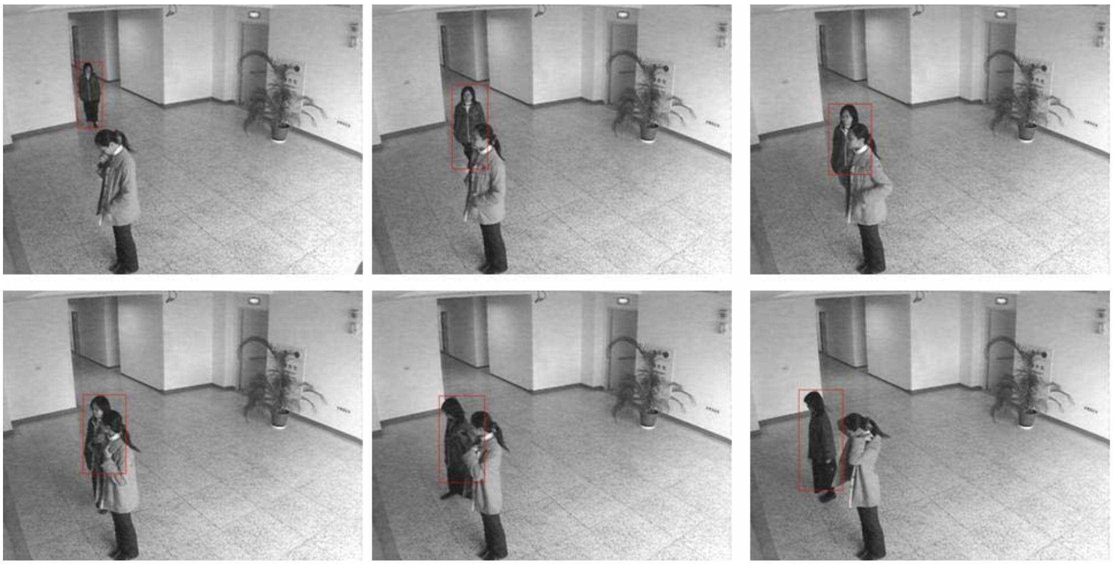

Fig. 5.2.1 Our tracking target is not completely occluded

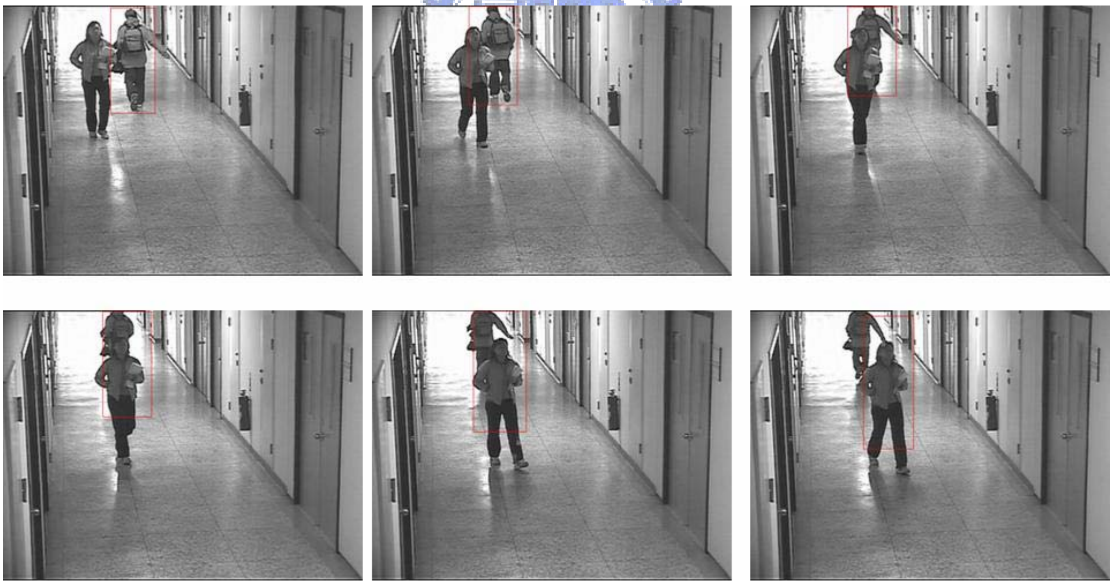

Fig. 5.2.2 Information of our tracking target is less and less.

5.2.3 Variant Intensity

Intensity of the same human may be variant because of different illuminance in different scenes such as the figure 5.2.3. In appearance-independent human matching, significant change of intensity will result in miss matching for a person. However, if we can obtain several frames in different illuminance for a person in selection of key frames, we may prevent miss matching. Through changing input frames in retrieve, we can keep frames of right matching based on similar intensity.

5.2.4 Crumbled Entity of People

A entity of a person is crumbled much because parts of him may be similar with the background in intensity. We consider two cases.

One case is the human in the reference image is crumbled. The figure 5.2.4 is an example. The person is too crumbled to obtain high degree of similarity with others. In other words, we cannot obtain better matching for him.

The other case is the person that should be matched is missed matching because of a crumbled entity such as the figure 5.2.5. Because an entity of the person is crumbled, we may obtain another similar person with a complete entity. In the figure 5.2.5 (a), the left is a person with a complete entity in the reference image, the median is a person matched, and the right is the person himself missed matching resulting from a crumbled entity. And, the figure 5.2.5 (b) is similar.

5.2.5 Different Appearances in Front View and Back View

Appearances of front view and back view of a person may be distinctly different such as the figure 5.2.6. For example in the figure in 5.2.6, appearances of front view and back view are different when a person wears a jacket or carries an object.

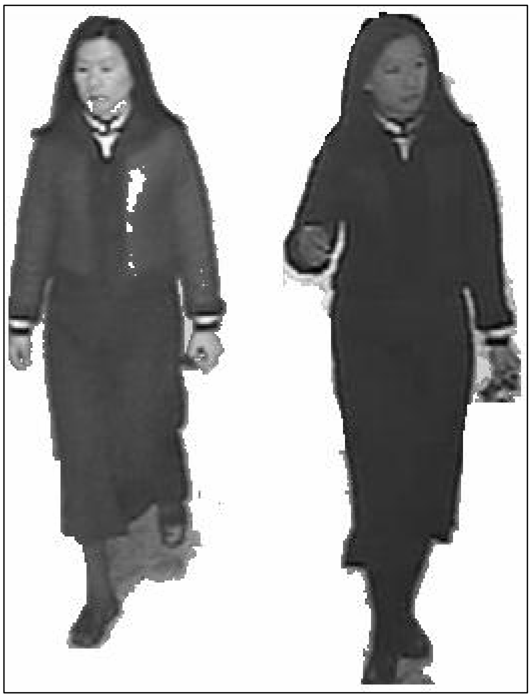

Fig. 5.2.3 A person has different intensity.

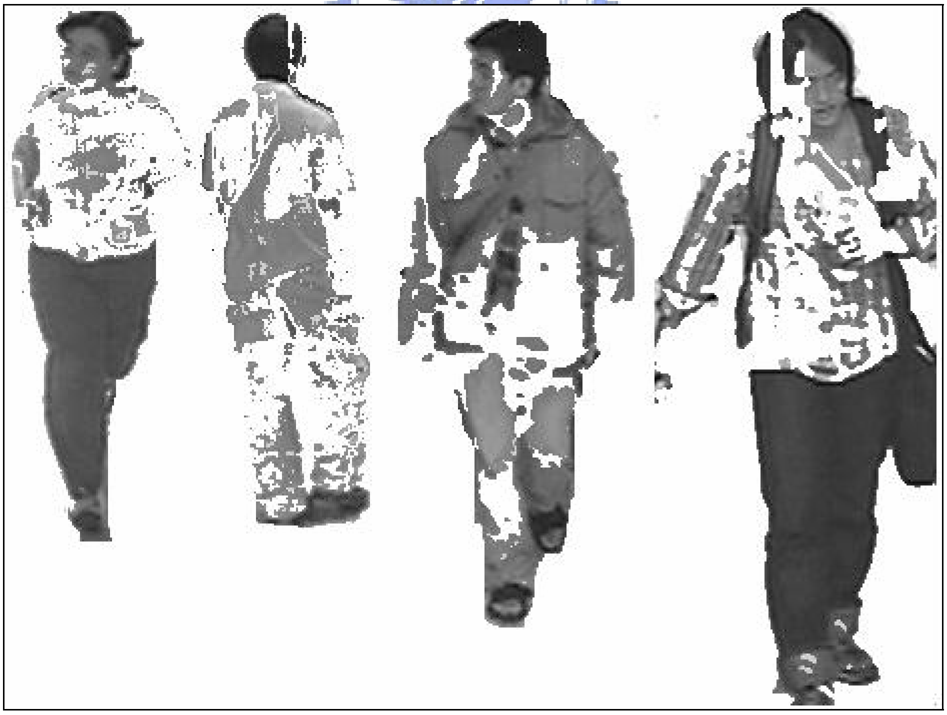

Fig. 5.2.4 Entities of people are crumbled.

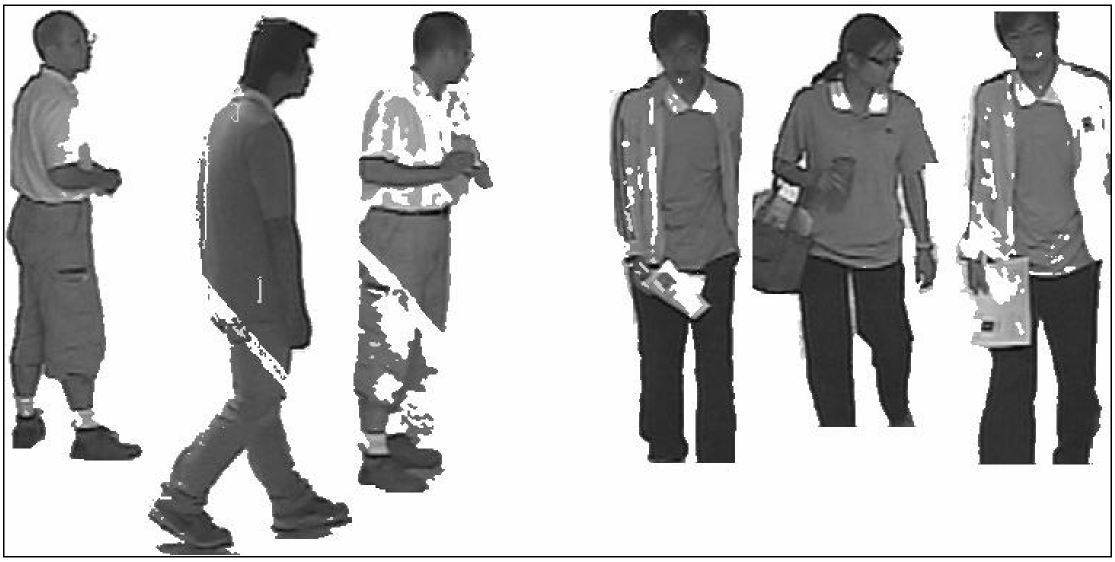

Fig. 5.2.5 The person that should be matched is crumbled.

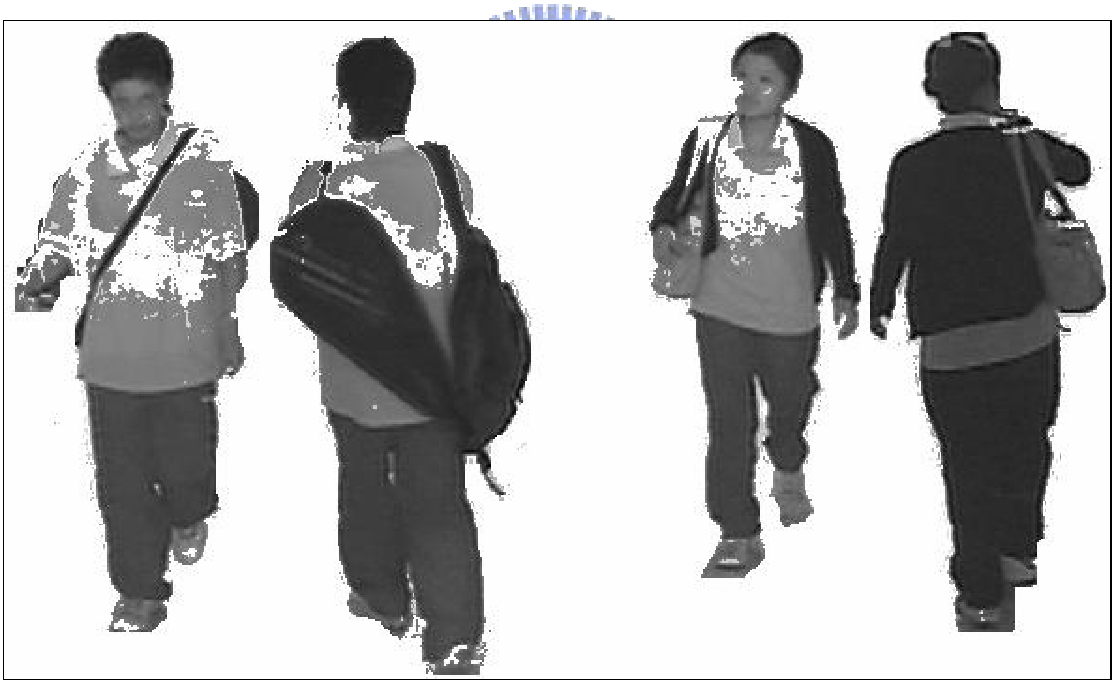

Fig. 5.2.6 Appearances of front view and back view may be much different.

5.3 Discussion

The table 5.1.1.1 shows efficient performance in tracking.

The table 5.1.2.1 shows that human matching in multiple cameras has high matching rates because people needed to compare is less than those in arching.

The table 5.1.3.3 shows that exploiting appearance-independent human matching can obtain the best matching from image archiving effectively.

The table 5.1.3.2 shows:

1. Exploiting people in front view to obtain those in side view is the most efficient.
2. Exploiting people in side view to obtain those in front view and back view is also excellent.

Because the appearance of side view is between that of front view and that of back view, exploiting the appearance of side view is better than exploiting that of front view to obtain that of back view.

CHAPTER 6 CONCLUSION AND FUTURE WORKS

In this thesis, we design a system facilitating human matching in archiving. The system consists of three stages: human tracking, human matching, and key frame selection and archiving.

We apply the statistical background model to extract the foreground from the image. The method could segment the foregrounds efficiently. The quarter search algorithm assists to obtain best matching position of a person in the next frame from the search region centered on the current position *L*. And, feature matching determines the matching result between a candidate position from the search region and *L* in the processing of the quarter search algorithm. These three methods cope with human tracking effectively.

A method is proposed to human matching under different appearances resulting from different capturing views of cameras. This method is capturing-view-resistant for matching different appearances of a person. We can confirm that human matching is effective in spite of different appearances.

A method is proposed to obtain key frames of a person in the surveillance system and connect corresponding key frames into image sequences for him and record these image sequences to archiving based on appearances of him.

Then, if we want to obtain several images from archiving for a specific person, we can exploit appearance-independent human matching to find corresponding image frames of him. The experimental results show that human matching could supply the reliable matching results from archiving.

For the future work, we suggest several directions to improve the performance of our system.

The foreground segmentation could be more efficiently by adding the color information. The problem of people in group could be solved by using tracking information.

Human matching could be more accurate by adding color information.

Design a light-resistant method to handle variable intensity of a person resulting from changing light in human matching.

Improve matching rates efficiently when different persons having similar appearances.

Reference

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

[2] C. StauOer and W. Grimson, “Adaptive Background Mixture Models for Real-Time Tracking,” *Proceedings of the IEEE CS Conference on Computer Vision and Patter Recognition*, vol. 2, pp. 246-252, 1999.

[3] A. Elgammal, D. Harwood, and L. Davis, “Non-Parametric Model for Background Subtraction,” *Proc. IEEE Frame Rate Workshop*,1999.

[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.

[19] Jain, J. R., and A. K. Jain, “Displacement Measurement and its Application in Inter-Frame Image Coding,” *IEEE Transaction on Communication*, vol. Com-29, no. 12, Dec., 1981.

[20] Koga, and T., “Motion-Compensated Inter-Frame Coding for Video Conferencing,” *Nat. Telecommun. Conf.*, G5.3.1-5, Nov., 1981.

[21] R. Li, B. Zeng, and M. L. Liou, “A New Three-Step Search Algorithm for Block Motion Estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 4, Aug., 1994.

[22] L.-M. Po and W.-C. Ma, “A Novel Four-Step Search Algorithm for Fast Block Motion Estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 313-317, Jun., 1996.

[23] Q. Cai and J. K. Aggarwal, “Tracking Human Motion in Structured Environments Using a Distributed-Camera System,” *IEEE Trans. on Pattern Analysis and Machine Intell.*, vol. 21, no. 12, Nov., 1999.

[24] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, “Multi-Camera Multi-Person Tracking for EasyLiving,” *Proc. IEEE Int. Workshop Visual Surveillance,* pp. 3-10, July, 2000.

[25] Q. Javed, S. Khan, Z. Rasheed, and M. Shah, “Camera Handoff: Tracking in Multiple Uncalibrated Stationary Cameras,” *Proc. IEEE Workshop Human Motion*, pp. 113-118, 2000.

[26] H. Wu, Q. Chen, and M. Yachida, “Face Detection from Color Images Using a Fuzzy Pattern Matching Method,” *IEEE Trans. on Pattern Analysis and Machine Intell.*, vol. 21, no. 6, June, 1999.

[27] D. Chai and K. N. Ngan, “Face Segmentation Using Skin-Color Map in Videophone Applications,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, no. 4, June, 1999.