UNIVERSITY OF SOUTHAMPTON

On Using Gait to Enhance Face Extraction for Visual Surveillance

by

Sung-Uk Jung

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Physical and Applied Sciences
School of Electronics and Computer Science

May 2012

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL AND APPLIED SCIENCES
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Sung-Uk Jung

Visual surveillance finds increasing deployment for monitoring urban environments. Operators need to be able to determine identity from surveillance images and often use face recognition for this purpose. Unfortunately, the quality of the recorded imagery can be insufficient for this task.

This study describes a programme of research aimed to ameliorate this limitation. Many face biometrics systems use controlled environments where subjects are viewed directly facing the camera. This is less likely to occur in surveillance environments, so it is necessary to handle pose variations of the human head, low frame rate, and low resolution input images.

We describe the first use of gait to enable face acquisition and recognition, by analysis of 3D head motion and gait trajectory, with super-resolution analysis. The face extraction procedure consists of three stages: *i*) head pose estimation by a 3D ellipsoidal model; *ii*) face region extraction by using a 2D or a 3D gait trajectory; and *iii*) frontal face extraction and reconstruction by estimating head pose and using super-resolution techniques.

The head pose is estimated by using a 3D ellipsoidal model and non-linear optimisation. Region- and distance-based feature refinement methods are used and a direct mapping from the 2D image coordinate to the object coordinate is developed. In face region extraction the potential face region is extracted based on

the 2D gait trajectory model when a person walks towards a camera. We model a looming field and show how this field affects the image sequences of the human walking. By fitting a 2D gait trajectory model the face region can then be tracked. For the general case of the human walking a 3D gait trajectory model and heel strike positions are used to extract the face region in 3D space. Wavelet decomposition is used to detect the gait cycle and a new heel strike detection method is developed. In face extraction a high resolution frontal face image is reconstructed with low resolution face images by analysing super-resolution. Based on the head pose and 3D ellipsoidal model the invalid low resolution face images are filtered and the frontal view face is reconstructed. By adapting the existing super-resolution the high resolution frontal face image can be synthesised, which is demonstrated to be suitable for face recognition.

The contributions of this research include the construction of a 3D model for pose estimation from planar imagery and the first use of gait information to enhance the face extraction and recognition process allowing for deployment in surveillance scenarios.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, **Sung-Uk Jung**, declare that the thesis entitled,

"**On Using Gait to Enhance Face Extraction for Visual Surveillance**"

and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- none of this work has been published before submission, **or** [delete as appropriate] parts of this work have been published as: [please list references]

Signed: ……..

Date:…….

Acknowledgements

Firstly, I would like to express my gratitude to my supervisors, Professor Mark S. Nixon for his invaluable guidance and encouragement during my Ph.D. Especially, without his cheerful encouragement and insightful comments this research would not be emerged to the bright place.

Thanks also go to my friend, Dr. John D. Bustard for providing his practical comments and helping me to understand the vague field of the 3D geometry. Additionally, I would like to thank my friends and colleagues from ISIS group for their assistance and friendship over the last few years.

Last, but not least, I am indebted to my wife who shared an enormous part of the burden in order to complete my degree. Also, my gratitude goes to my parents, for their love and encouragement throughout my Ph.D. and everything.

Thank you all.

Chapter 1

Introduction

1.1 Motivation and Scope

As Closed-Circuit TeleVision (CCTV) becomes more widespread, it can be used for authentication, visual surveillance, and monitoring; however, developments have yet to meet application requirements. For example, the police use manual search tactics to find a criminal in CCTV images and airports depend on documents, although biometrics is becoming more prevalent. Automatic face recognition potentially has a major role to play in surveillance, information security, and access control applications. To date, automatic face recognition has mainly used two dimensional frontal face pattern and texture information. Even when these are confined to indoor surroundings, the resulting recognition capability is not perfect, although performance continues to improve. However, one of the main reasons for this improvement has been the use of very high resolution frontal face images in constrained environments [1]. In contrast, in (visual) surveillance environments, a camera is in an elevated position to achieve the widest field of view. Many of these cameras have low resolution and low frame rate. Therefore, there are many constraints to adapting existing face tracking /recognition algorithms.

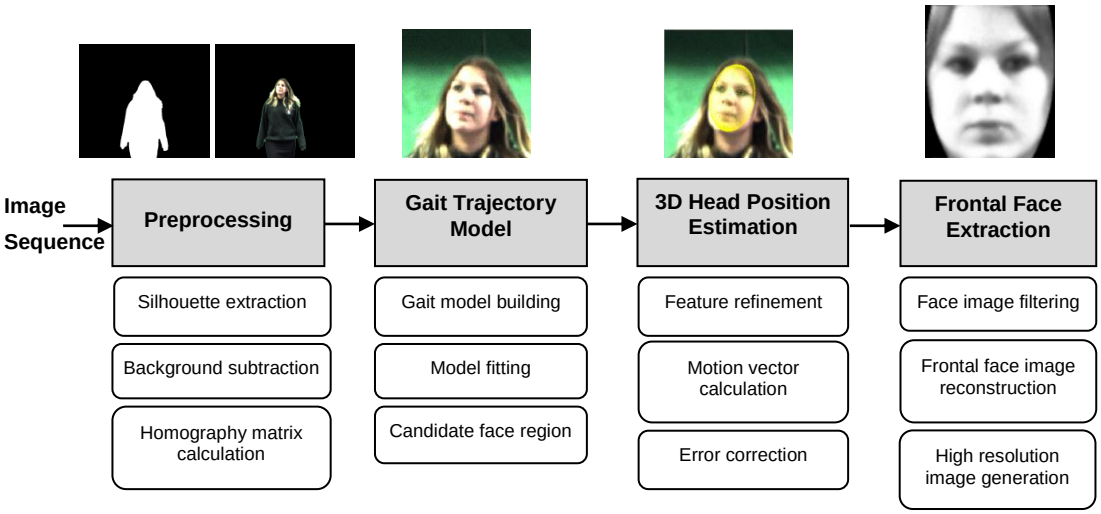


Figure 1.1: Front-face acquisition process

In this research, we focus on the specific but also one of the most general environments in visual surveillance where the subject is viewed walking towards a camera. We suppose that the image sequences are recorded at low frame rate and low resolution. To overcome these constraints, a 3D ellipsoidal head pose model is built, and we estimate 3D head pose using non-linear optimisation. However, the head pose model can fail to track the head when there is large head pose variation and changes in illumination. Thus, we use additional information – the gait trajectory. The gait trajectory can be obtained from the movement of the corresponding points between frames. Analysing the gait trajectory allows the system to detect the approximate face regions. In these face regions the detection rate can be higher. Accordingly, this is the first work which uses gait to enhance face acquisition and extraction.

Figure 1.1 describes the whole process of frontal face extraction; preprocessing, gait trajectory model generation, head pose estimation, and frontal face extraction.

In preprocessing, we first remove the redundant parts from the input images and calculate the relationship between the object movements according to the frame number. The silhouette image is generated for every frame. Then, the background is subtracted to remove the invalid region. By using Scale Invariant Feature Transform (SIFT) [40] feature extraction, the Homography matrices between all of the images are calculated to track the movement of the specific pixel's position from the first frame.

We extract the approximate region which is most likely to contain the face by using 2D and 3D gait trajectory models. By using the trajectory calculated by the Homography matrix this approach can be robust to low image resolution and frame rate compared with previous face detection algorithms which used the face pattern. The movement of the upper body between the frames is calculated using the Homography matrix. The gait trajectory models are applied to extract the accurate gait trajectory enabling the approximate face region to be extracted.

For head position estimation, we estimate the head pose from the approximate face regions. The tracking result can be a closer fit than applying the pose estimation algorithm directly to the raw image. First, the SIFT corresponding points are filtered by region- and distance-based refinements. Then, the head pose is calculated by non-linear optimisation which minimises the objective function consisting of unknown rotation and translation vectors. Finally, optical flow and texture map from a 3D ellipsoidal model are used to correct errors in the estimated head pose.

For frontal face extraction we extract or generate the best quality frontal face image. Our target is to derive images of the face suited to front view automatic face recognition algorithms or for use by human observers. Considering the rotation of the head and the resolution of the face image, first we remove the low resolution face images with large head rotation. Then, we reconstruct the frontal

view face from the images. Finally, a High Resolution (HR) image is generated from the selected Low Resolution (LR) images by super resolution analysis.

1.2 Related Work

1.2.1 High Resolution Face Image Reconstruction

There are many approaches applied to obtain the HR face images. Medioni et al. [2] detected a person and located their face using a fixed ultrahigh-resolution camera at a distance. Using the face region, they framed and reconstructed a 3D face model and recognised a face within it. Wheeler et al. [3] initialised LR face images using an Active Appearance Model (AAM) [22] and Bilateral Total Variation (BTV) [36] to generate HR images. This process was tested with an outdoor video using a Pan-Tilt-Zoom (PTZ) camera and a commercial face recognition system (FaceIt - Identix). Mortazavian et al. [4] described an example-based Bayesian method for 3D-assisted pose-independent face texture super-resolution. They used a 3D Morphable Model (3DMM) [14] to map facial texture from a 2D face image. Jia and Gong [5] proposed learning-based face Super Resolution (SR) techniques to generate a HR image of a single facial modality such as a fixed expression, pose and illumination from given LR images.

Other studies have learned the relation between HR and LR image, and utilised the relationship to recognise the LR face images from visual surveillance data. Hennings-Yeomans [6] proposed LR face image recognition when there is a HR training set available. Unlike conventional methods, the face features such as Eigen-faces or Fisher-faces are included in a super-resolution method as prior information. Zou [7] proposed a nonlinear face super resolution algorithm from surveillance video which learns the nonlinear relationship between LR and HR face image in nonlinear kernel feature space. Li [8] showed the coupled mappings

method which projects the face images with different resolution into a unified feature space which favours the task of classification without using a super resolution algorithm.

For the above approaches it is necessary to align the LR images for applying the SR algorithm. In other words, if there is variation between the LR images the quality of the HR images could be degraded. Also, the HR generation process is greatly affected by the SR algorithm performance.

1.2.2 3D Face Model based Face Reconstruction

Three dimensional model-based approaches have been developed to obtain an accurate face model [9, 10]. Park and Jain [11] initialised face images using an AAM and synthesised a 3D frontal face from 2D low resolution images. They demonstrated performance using commercial recognition software (FaceVACS – Cognitec) and the Face In Action (FIA) database. This system was a single camera-based system, requiring initialisation. Duhn et al. [12] generated a 3D face model using three 2D images and tracked a face using an AAM. In the tracking stage, a generic model was adapted to the different views of the faces. However, this method still needed automatic initialisation for tracking. Given 2.5D face laser scan images, Lu et al. [13] constructed a 3D head model by fitting the laser scan data with the pre-trained face texture and shape. They used Iterated Closest Points (ICP)-based surface matching to construct the 3D model; however, this approach suffered from the computational costs of 3D data acquisition and processing. The most representative method of 3D face modeling is a 3DMM [14] wherein a 3D face model was constructed by using a form of 3D Principal Components Analysis (PCA). The coefficients of texture and shape are used for face recognition. However, both training and test images need to be manually labeled and the cost of 3D data acquisition and processing is high. Liao et al. [15]

built a 3DMM based on a single image to recognise a person. They generated synthetic 2D images in the training stage and recognised a face using a Support Vector Machine (SVM) with imposter rejection by local linear embedding. In the CHIL project [16] the goal of the project was to acquire information about the (smart) room by video surveillance of the people in it and their interaction. The tasks of this project included person tracking, person identification, and head pose estimation, and the recognition rate of individuals was from 66.2% to 75.9% using low resolution face images.

1.2.3 Face and Gait Fusion

There are other approaches in visual surveillance using multi-modal biometrics – face and gait. Since the most distinguishable feature in these environments is the human gait, earlier approaches tried to combine these two modalities to improve recognition. Kale et al. [17] discussed the fusion of face and gait cues for a single camera, based on sequential importance sampling. Gait recognition was used to reduce the set of candidates for face recognition. The results of fusion experiments were demonstrated on the National Institute of Standards and Technology (NIST) database which had outdoor gait and face data of 30 subjects [18] and suggested that a multi-modal biometric could achieve a higher recognition rate than a single-modal biometric. Shakhnarovich et al. [19] developed a view-normalisation approach to multi-view face and gait recognition. In this approach, the Image-Based Visual Hull (IBVH) was computed from a set of monocular views and used to render virtual views. Zhou et al. [20] combined cues of face profile and gait silhouette from a single camera video sequence. They reconstructed a high resolution face profile image and then used a Gait Energy Image (GEI) to characterise human walking properties.

The above approaches use gait biometrics as a parallel module to complement the face recognition algorithm and enhance the recognition rate. Therefore, score fusion between the face and the gait recognition can be another issue.

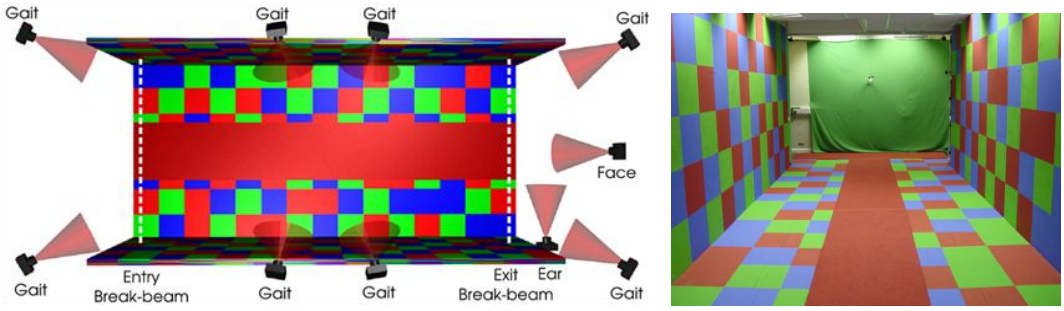
1.3 Databases

Several databases have been used for demonstrating the performance of the proposed methods. As representatives of controlled environments, the Biometric tunnel database [21] and the Boston face dataset [24] are used. For examples of the representing uncontrolled environments, we use the CAVIAR database [45] and the PETS 2006 data [46].

1.3.1 Biometric Tunnel Database

The Multi-Biometric tunnel [21] has been specifically designed as a non-contact biometric access portal, providing a constrained environment for people to walk through, whilst facilitating recognition. It contains 12 synchronised IEEE 1394 cameras for gait analysis and two cameras for face and ear images. The gait cameras have a resolution of 640 × 480 pixels and capture at a rate of 30 FPS (Frames per Second). The resolution of a face image is 1600 × 1200 pixels and 10 FPS. The dataset consisted of a total of 2705 samples from 227 subjects. Of the subjects, 67% were male, the majority were aged between 18 to 28 years old, and 70% were of European origin [44].

In Chapters 3 and 6, the frontal camera images are used to evaluate the gait trajectory model and synthesise the high resolution face image. The images from the gait camera are used for detecting heel strike position and extracting the head region in Chapters 4 and 5. Figure 1.2 shows a data capture environment – the Multi-Biometric tunnel and sample data from the different views.



(a) Multi-Biometric tunnel

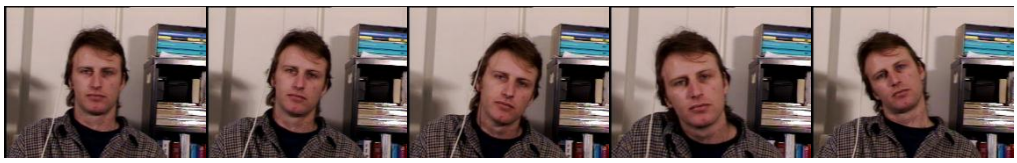
(b) Sample images from a face camera

(c) Sample images from a gait camera

Figure 1.2: Biometric tunnel and the sample data

1.3.2 Boston Dataset

The Boston face dataset [24] aims to test 3D head pose estimation algorithms in the presence of large head rotation (-40°~+40° in pitch, yaw, roll direction). It consists of 45 image sequences for uniform-light environments and 27 image sequences for varying-light environments. All image sequences were taken indoors at a fixed distance from a camera. In addition, this dataset provides image sequences with the ground truth data. The image size is 320 × 240 pixels and the frame rate is 30 FPS. In Chapter 2, we deploy this dataset to verify the head pose estimation algorithm. Figure 1.3 shows some sample data.



(a) Rotation in roll direction

(b) Rotation in yaw direction

(c) Rotation in pitch direction

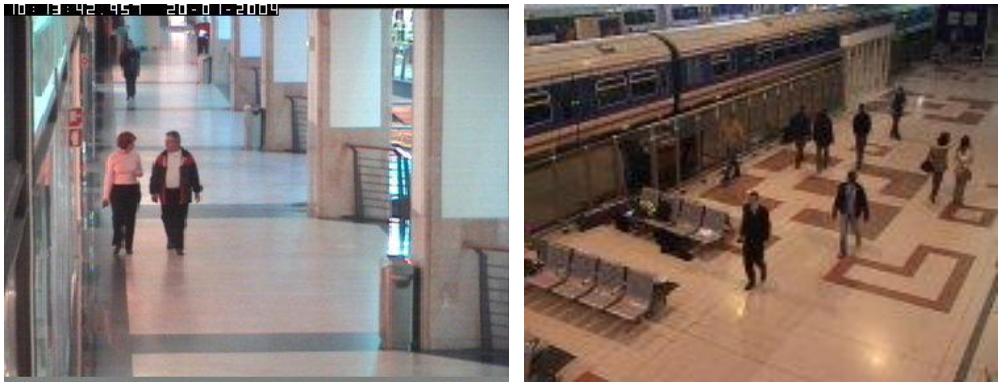
Figure 1.3: Sample data of the Boston face dataset

1.3.3 CAVIAR Database

For the CAVIAR project a number of video clips were recorded acting out the different scenarios of interest. These include people walking alone, meeting with others, window shopping, entering and leaving shops, fighting and passing out

and last, but not least, leaving a package in a public place [45]. In this research we used the second set of data for evaluating the heel strike detection algorithm in Chapter 4.

The second set of data used a wide angle lens along and across the hallway in a shopping centre in Lisbon. For each sequence, there are two time-synchronised videos, one with the view across the hall and the other with the view along the hallway. The resolution is 384 × 288 pixels and the frame rate is 25 FPS.



(a) The 2nd set of data in CAVIAR (b) The 4th camera image in PETS06

Figure 1.4: Sample data of the CAVIAR and the PETS06 dataset

1.3.4 PETS 2006 Database

The aim of this dataset is to use existing systems for the detection of left (i.e. abandoned) luggage in a real-world environment. The scenarios are filmed using multiple cameras and involve multiple actors [46]. This database consists of Datasets S1 to S7 and each dataset was recorded in four different environments. The image resolution is 768 × 576 pixels and the frame rate is 25 FPS.

In Chapter 4 we use an image sequence from camera 4 of Dataset S1 in this database to evaluate the performance of heel strike detection in a realistic surveillance. Figure 1.4(b) shows the capturing environment.

1.4 Contributions

Several new methods for extracting the frontal face image by using human gait characteristics are developed in this study. The major contributions of this study are as follows:

- We propose a new method for estimating head pose using a 3D ellipsoidal model and a non-linear optimisation method [A][D]. Based on the 3D ellipsoidal model we develop new feature refinement methods and a direct mapping from 2D image coordinate to object coordinate. The Levenberg-Marquardt is used to calculate the motion vector which minimises the error function.

- We propose a new method for extracting the face region based on a 2D gait trajectory model when a person walks towards the camera [A][F]. We describe a looming field and show how this field affects the image sequences of the human walking. Based on the looming field we define a 2D gait trajectory model which can estimate the face region regardless of image constraints such as low frame rate, low image resolution, and changes in illumination.

- We propose a new method for extracting the face region by a 3D gait trajectory model which is not constrained by the walking speed and direction [C][G]. This method is based on the analysis of the movement of the human upper body and the heel strike position in 3D space [B][E]. We analyse the upper body movement using wavelet decomposition and develop a new method of heel strike detection.

- We propose a new method for reconstructing a high resolution frontal face image by analysing super-resolution [F]. Based on the head pose and the

3D ellipsoidal model we remove large head motion face images and reconstruct frontal view face images. Then, by using the super-resolution techniques, we synthesise a high resolution frontal face image from the pose-corrected low resolution images.

1.5 List of Publications

The publications resulting from this research, so far include

[A] S. U. Jung and M. S. Nixon, "On using gait biometrics to enhance face pose estimation," in *Proceedings of the IEEE Conference on Biometrics: Theory, Applications and Systems (BTAS10)*, 2010, 6 pp.

[B] S. U. Jung and M. S. Nixon, "Detection human motion with heel strikes for surveillance analysis," in *Proceedings of the International Conference on Computer Analysis of Images and Patterns (CAIP11)*, *LNCS 6854*, 2011, pp. 9-16.

[C] S. U. Jung and M. S. Nixon, "Estimation of 3D head region using gait motion for surveillance video," in *Proceedings of the International Conference on Imaging for Crime Detection and Prevention (ICDP11)*, 2011, 6 pp.

[D] S. U. Jung and M. S. Nixon, "Model-based feature refinement for ellipsoidal face tracking," submitted to *International Conference on Pattern Recognition (ICPR12),* 2012.

[E] S. U. Jung and M. S. Nixon, "Heel strike detection based on human walking movement for surveillance analysis," submitted to *Pattern Recognition Letters,* Dec. 2011. (invited from *CAIP11*)

[F] S. U. Jung and M. S. Nixon, "On using gait to enhance frontal face extraction," Revision submitted to *IEEE Transaction on Information and Forensics Security*, March. 2012.

[G] S. U. Jung and M. S. Nixon, "Head region detection with gait motion," submitted to *IEEE Transaction on Systems, Man, and Cybernetics, Part A,* April. 2012.

Chapter 2

3D Head Pose Estimation

2.1 3D Head Pose Model

Calculating the 3D head pose is a fundamental process for unconstrained automatic face recognition systems. The 3D head pose describes not only direction but also basic information such as the size and position of the head. Accurate information of the head pose is also essential for face recognition. In the case of 2D face recognition the variation of the head pose results in a different recognition rate. Also, in the case of 3D face recognition, it indicates the position in a 3D space. Therefore, if the information is not accurate, it can result in a low recognition rate [76].

Basically, estimating the head pose requires calculation of the translation and rotation information in 3D space. There are two main categories in existing methods: using an actual 3D head model or using an approximate head model. In the case of using actual 3D head models such as 3D AAM [22] and 3DMM [14], the advantage of these models is to be able to obtain accurate 3D texture and face shape. There are however some disadvantages; for example, exact initialisation is required and imperfect initialisation can cause tracking errors. Unlike the actual

models, the approximate head models such as 2D plane, 3D cylinder, and 3D ellipsoid cannot generate the actual shape and texture of a face. However, the model can be simple to implement and the computational load of a fitting process is much lower than the actual model methods. Also, the initialisation can be automatic.

To estimate 3D head pose using an approximate head model, Liu et al. [41] used SIFT features [40] to match the corresponding feature points between two adjacent views. Using Epipolar geometry [29], the fundamental matrix was calculated to convert the fundamental matrix into the essential matrix to obtain the pose information; however, this method needs a frontal face as a reference image to obtain accurate rotation angles. Hager et al. [23] generated a 2D plane model using a single camera and Lucas-Kanade tracking. However, this method was not robust enough for a large out-of-plane rotation and a movement in depth. Cascia et al. [24] generated a 3D cylinder model, where 3D head motion was treated as a linear combination of motion templates and orthogonal illumination templates. The system was initialised automatically using a simple 2D face detector. Basu et al. [25] interpreted the optical flow in terms of the possible rigid motion and applied it to heads with a variety of shapes and hair styles, using a 3D ellipsoidal model. They tested the sequence of images including low and high frame rate and noisy camera images. Xiao et al. [26] used a 3D cylinder model to track the head. An Iteratively Re-weighted Least Squares technique (IRLS) was adapted to fit the face to the model. Also, the templates were updated to diminish the effects of self-occlusion and gradual lighting changes while tracking. Jang and Kanade [27] initialised the face using the Bayesian Tangent Shape Model (BTSM) face alignment method. SIFT and normalised correlation methods were used to extract and match the feature points. The method removed outliers using Weighted Least Squares (WLS) and estimated the motion using a Kalman filter.

There are three main approximate head models. The 2D plane model is simple but not effective for the human head since it does not represent curved surfaces and is not robust to out-of-plane rotations. The 3D cylinder model can represent a vertically curved surface well, although it is less accurate than an ellipsoidal model because the ellipsoidal model approximates better the top and the bottom of the head. Also, a more accurate unfolded face image (as used to correct the tracking error) can be generated from the fit of the ellipsoidal model than can be generated by the cylinder model and the flat model. So, we shall use the 3D ellipsoidal model to represent the human head.

The 3D ellipsoidal model is defined by the following:
A point on the 3D object $\mathbf{P}_o$ can be represented by $[X_o\ Y_o\ Z_o]$. There is a simple relationship between $\mathbf{P}_o$ and the angles (α,β) shown in equation (2.1).

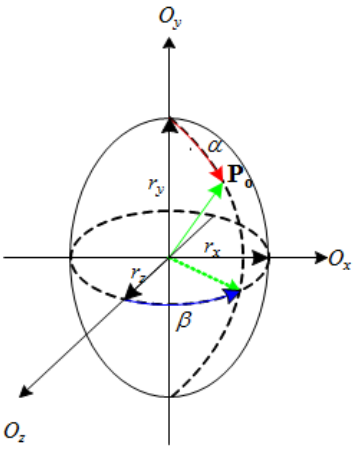
$$X_o = r_x \sin\alpha \sin\beta \quad Y_o = r_y \cos\alpha \qquad Z_o = r_z \sin\alpha \cos\beta \tag{2.1}$$

where r_x, r_y, r_z are the radius of the object along each axis.

Each angular resolution is one degree, so that total number of model components is 360 × 360. The Scale Invariant Feature Transform (SIFT) [32] is deployed to find corresponding points between adjacent images. We calculate a 3D rotation and translation matrix from the SIFT points matched between adjacent images later.

In figure 2.1(a) the yellow dots represent the visible 3D points and the red dots are the extracted corresponding points between the adjacent images. The green cross line depicts the centre line of the model (α = +90° and β = 0° in the visible parts). Figure 2.1(c) shows the unfolded image from the fitted model (0° ≤ α ≤ +360°, -180° ≤ β ≤ +180°). Unlike the cylinder model the ellipsoidal model considers the distinguishable part of the face excluding the background.

(a) The sample of model fitting



(b) 3D ellipsoid model

(c) Unfolded image

Figure 2.1: The 3D ellipsoidal model

2.2 Feature Extraction

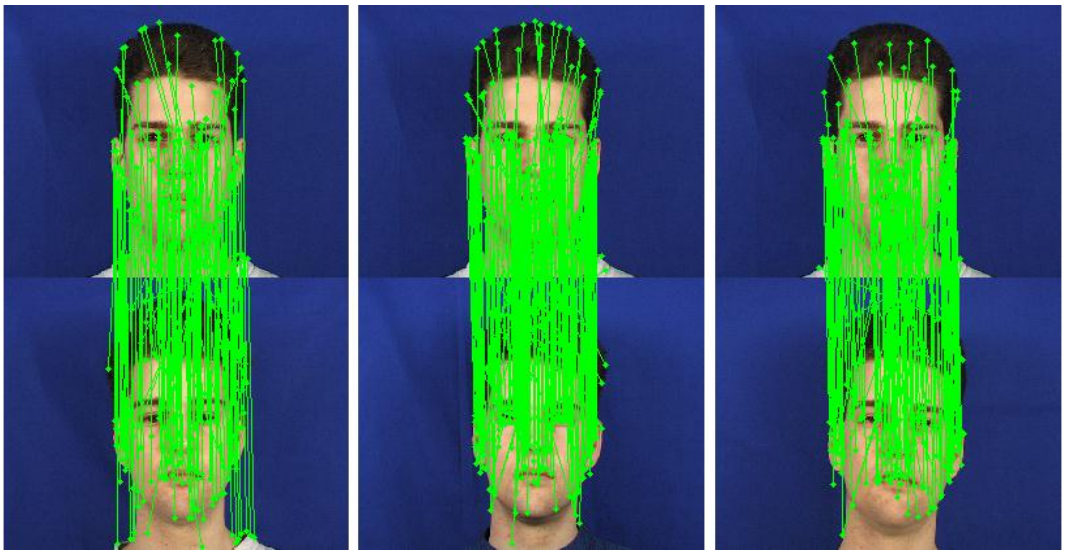
2.2.1 Feature Extraction in Pose and Size Variation

Feature extraction is the fundamental step for analysing an object's movement in video. In visual surveillance images there might be large pose and size variations, so features should be invariant to image scale, rotation, and affine transformations. Further, the features detected in successive images should be robust against changes in illumination and 3D viewpoint, and to noise. In this thesis, the first step of 3D head pose estimation and gait trajectory calculation is to extract the corresponding points between the target images. There are many approaches for local feature extraction [30, 40, 69-72]. Among them, two state of the art feature extraction methods are deployed: Scale-Invariant Feature Transform (SIFT) [40] and Speeded-Up Robust Feature (SURF) [30].

Currently, the target object of this thesis is a human face and the target environment is where a person walks towards a camera with the ultimate aim of an unconstrained surveillance environment. Thus, in the environment, there are head pose and image size variations. For example, the size of the head becomes larger when a person walks towards the camera and the captured face image can change with the direction of head.

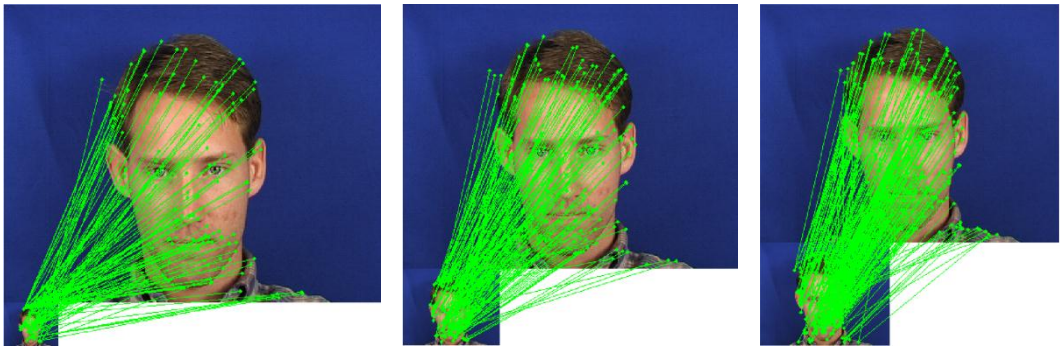
To compare the performance between the above two feature extraction methods, first, the features are extracted from frontal face images. The experimental samples are the frontal faces from the XM2VTS face database [73] and each size is 692 × 548 pixels. The feature extraction method is SIFT. In figure 2.2 the green points show extracted interest points and the lines display the matched points. There are many matching points in the high resolution images. In total, there are 157, 193 and 173 matching points including mismatches. This result reveals that if

the image is of sufficiently high resolution, enough corresponding points can be extracted.



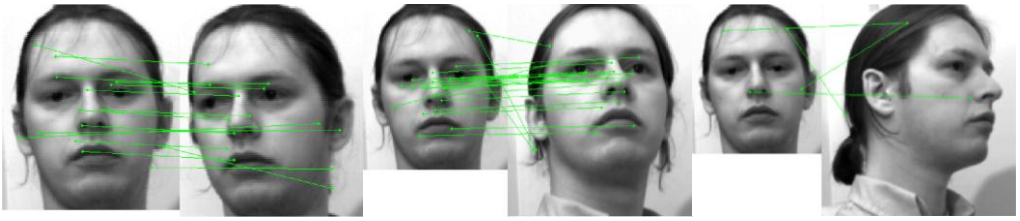
(a) 157 key point match (b) 193 key point match (c) 173 key point match

Figure 2.2: Feature extraction between same person and size

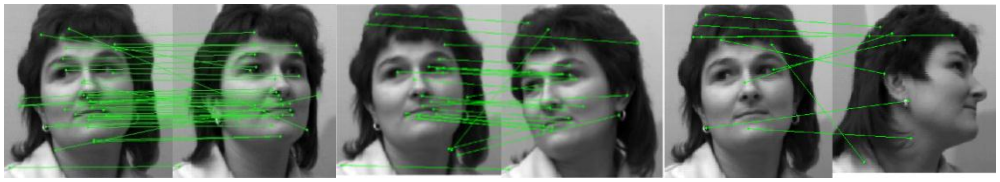


(a) 692 × 548 to 100 × 80 (b) 692 × 548 to 200 × 160 (c) 692 × 548 to 300 × 240

Figure 2.3: Feature matching between same person and different size



(a) 18 key point match (b) 23 key point match (c) 4 key point match



(d) 39 key point match (e) 28 key point match (f) 3key point match

Figure 2.4: Feature matching between same person and different pose

In the second case, three different resolution images are tested again using the XM2VTS face database. First, the images are reduced by Bicubic image resizing [74]. Then, the corresponding points are extracted by SIFT as shown in figure 2.3. As a result, the number of interest points is 1906 in the High Resolution (HR) image, 30 in the Low Resolution (LR) image, and 133 matching points in figure 2.3(a). In figure 2.3(b), there are 1906 points in the HR image and 126 points in the LR image, and the number of matching points is 202. In figure 2.3(c), there are 1906 points in the HR image, 274 points in the LR image, and 272 matching points. The result shows that over-fitting can take place during matching. In the first and second cases the number of matching points exceeds the number of interest points extracted from the LR image. Although this result depends on a threshold value within SIFT, it demonstrates that there are difficulties in extracting corresponding points if images are of significant different sizes.

The last experiment concerns pose variation. The Sheffield face database [75] is used, and two cases are tested: small pose and large pose variation. As shown in the results, the matching result can be reasonable in the case of a small pose variation (Figure 2.4(a),(b),(d),(e)). On the contrary, the matching almost fails

between the frontal and the profile faces (Figure 2.4(c),(f)). This is because SURF and SIFT are based on using a 2D image homography to determine the corresponding points. For a 3D shape like a face, occlusion or pattern distortion results from pose variation when the shape is projected into a 2D image.

From the above three experiments, we can conclude that it is reasonable to use SIFT and SURF methods if a high image frame rate is guaranteed or the object does not have a large movement. In other words, the point feature can be used if there are small head pose variations between frames, and the apparent variation in size of a head is not large.

2.2.2 Performance Comparison of Local Features

There are a number of methods that can be applied to estimate local feature performance [42, 43]. The Repeatability Score [42] estimates how well the detector determines corresponding scene regions. This is measured by comparing the ground truth transformation and detected region overlap. The ground truth is a homography that projects points onto the reference frame. Figure 2.5 shows the process of the estimation method [42].

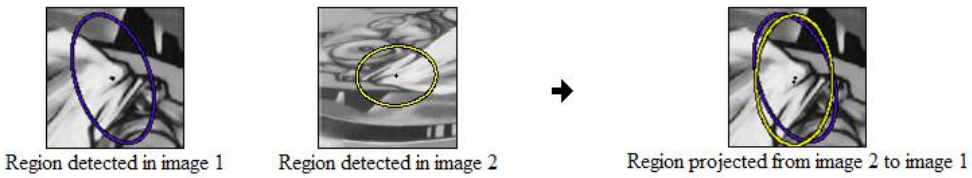


Figure 2.5: An example of the repeatability score

Figure 2.6: Samples of the reduced images

Two corresponding regions have an overlap defined by the ratio of their intersection/union. If the region detected in the first image is **A** and in the second one is **B**, then

$$\mathbf{E}=\mathbf{Region(A\cap B)\ /\ Region(A\cup B)} \tag{2.2}$$

If **E** is bigger than 40% the corresponding features are considered as matching points. The Matching Score [42] is another way to estimate local features. It compares the number of labeled regions corresponding with those in the ground truth. Matches are the nearest neighbours in the descriptor space. Recall vs. **1**-precision [43] is another estimation method. Recall is the number of correctly matched regions with respect to the number of corresponding regions between two images of the same scene. The number of false matches relative to the total number of matches is represented by **1**- precision. However, all of the above methods are the cases of 2D transformations. Thus, these estimation methods are insufficient to apply to a face, since a face is a 3D shape and does not include many textures. Therefore, in this thesis, we use the number of extracted interest points and matched points as a measure to compare two feature extraction methods by using different size face images.

To investigate the relative performance of SIFT vs. SURF we assume that there is no affine transform between the face images. As such, the inliers and outliers can be easily distinguished by analysing distance between matching points. Figure 2.6 shows samples of the resized images. The frontal faces from the XM2VTS database are used and reduced in the eight step sizes. The test database consists of 287 subjects and each subject contains eight resized images. However, to avoid over-fitting the reduced image is resized again to the same size as the reference image by two different interpolation algorithms: Bicubic and Nearest neighbour.

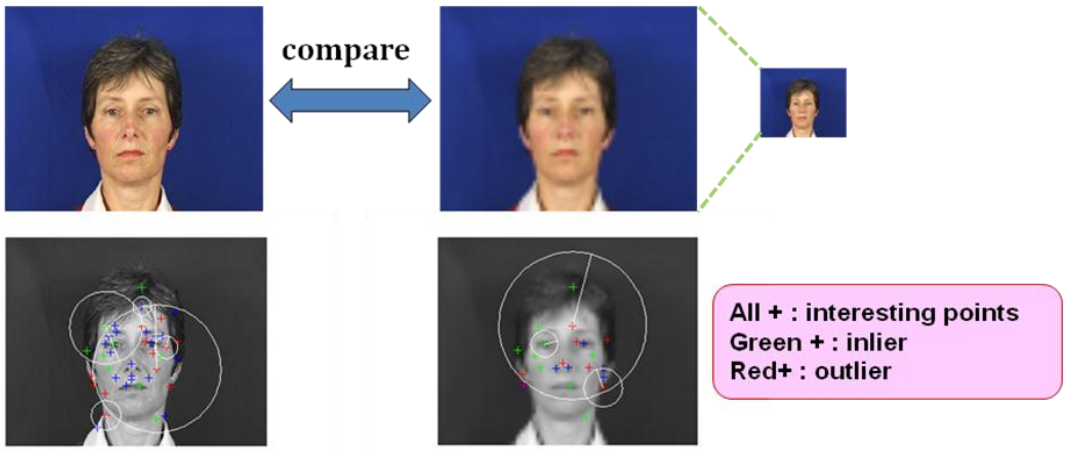
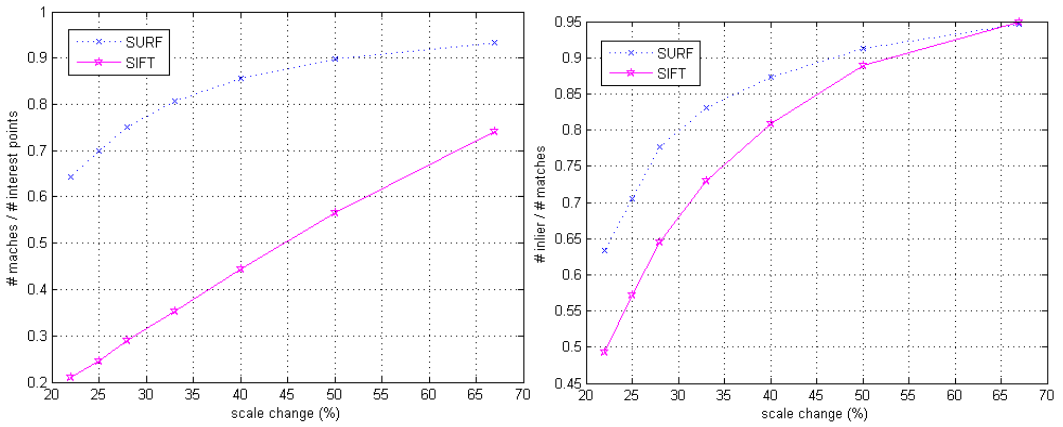


Figure 2.7: Experimental process

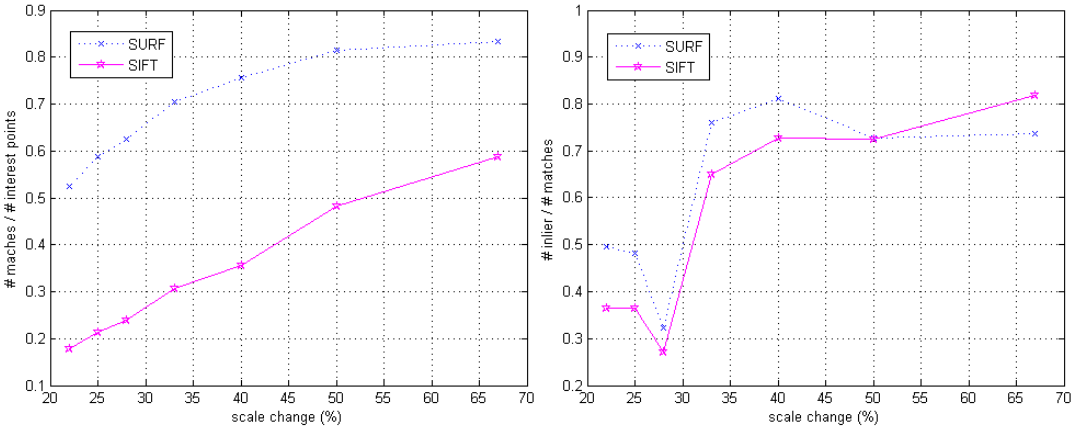
Figure 2.7 describes the experimental process. The original size image and resized image are compared. The green points are inliers, and the red points represent outliers. The matching process is conducted between the original sized image and the reduced images. The distance between the matching points decides whether the point is an inlier or an outlier. The matching result is displayed in the following tables and figure 2.8. Tables 2.1-2.4 display the number of interest points in the original image, matching points, inliers and outliers. As shown, the number of extracted interest points becomes smaller according to reduction in image size. Comparing SURF with SIFT, SIFT can extract more than five times as many interest points so that the number of matching points is much higher than that for SURF.

Figure 2.8 displays the ratio between matching points and interest points according to image size and ratio between inliers and matching points. Basically, the ratios become smaller when the reduction increases. Further, the ratios are almost 30% higher than those of SIFT. In other words, although SIFT can extract more interest points, many of them are mismatches. This means that SURF can extract the feature more efficiently than SIFT. In this thesis, however, we apply the SIFT method to calculate the motion vector due to its capability of extracting

many features. In visual surveillance environments the quality of the image could be low, such as low resolution, blurring, and changes in illumination. Thus, the approach which selects the most points appears most appropriate.



(a) SIFT and SURF performance comparison using Bicubic interpolation



(b) SIFT and SURF performance comparison using Nearest neighbour interpolation

Figure 2.8: Performance comparison

TABLE 2.1: Test results of SURF with size variation using Bicubic sampling

SURF point	67%	50%	40%	33%	28%	25%	22%
Num. total points	16510	16510	16508	16510	16510	16510	16510
Num. total matches	13769	13448	12483	11615	10323	9710	8655
Num. inlier	10150	9775	10126	8840	3337	4668	4301
Num. outlier	3619	3673	2356	2775	6986	5042	4354

TABLE 2.2: Test results of SIFT with size variation using Bicubic sampling

SIFT point	67%	50%	40%	33%	28%	25%	22%
Num. total points	78538	78538	78538	78538	78538	78538	78538
Num. total matches	46144	37839	27944	24149	18772	16675	13996
Num. inlier	37828	27417	20337	15731	5108	6084	5118
Num. outlier	8316	10422	7607	8418	13664	10591	8878

TABLE 2.3: Test results of SURF with size variation using Nearest neighbor sampling

SURF point	67%	50%	40%	33%	28%	25%	22%
Num. total points	16510	16505	16498	16488	16499	16499	16505
Num. total matches	13576	13721	12673	12358	11491	10640	10135
Num. inlier	9589	9247	7455	7099	4704	2987	2911
Num. outlier	3987	4474	5218	5259	6787	7653	7224

TABLE 2.4: Test results of SIFT with size variation using Nearest neighbor sampling

SIFT point	67%	50%	40%	33%	28%	25%	22%
Num. total points	78538	78538	78538	78538	78538	78538	78538
Num. total matches	38921	32501	24960	22744	17856	16208	13527
Num. inlier	27579	20649	13352	11623	6347	3916	3258
Num. outlier	11342	11852	11608	11121	11509	12292	10269

2.3 Head Pose Estimation using Non-linear Optimisation

2.3.1 Feature Refinements

The goal of this chapter is to calculate a 3D rotation and translation matrix from the matched SIFT points [40]. If the 3D corresponding points exist and there is a small movement between images, the transformation matrix could be calculated by using optical flow or the twist representation [28]. For the twist representation let points $\mathbf{x}_t = [x_t \ y_t \ z_t]^{\mathrm{T}}$, $\mathbf{x}_{\text{t-1}} = [x_{t-1} \ y_{t-1} \ z_{t-1}]^{\mathrm{T}}$ at time t and t-1 respectively. Thus, the estimated transform matrix can be calculated in the following relationship,

$$\begin{bmatrix} x_t \\ y_t \\ z_t \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & -\Delta\theta_z & \Delta\theta_y & \Delta t_x \\ \Delta\theta_z & 1 & -\Delta\theta_x & \Delta t_y \\ -\Delta\theta_y & \Delta\theta_x & 1 & \Delta t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ z_{t-1} \\ 1 \end{bmatrix} \tag{2.3}$$

where $\Delta\theta_x \ \Delta\theta_y \ \Delta\theta_z$ represent Euler rotation angles and $\Delta t_x \ \Delta t_y \ \Delta t_z$ represent translations for each x, y, z axis.

Another way of calculating the motion vector is to use Epipolar geometry [29]. This method needs camera calibration to obtain the essential matrix from which the rotation and translation information can be extracted. Let $\mathbf{x}_1, \mathbf{x}_2$ be the corresponding points for each image. The pose information can be obtained by following relationship,

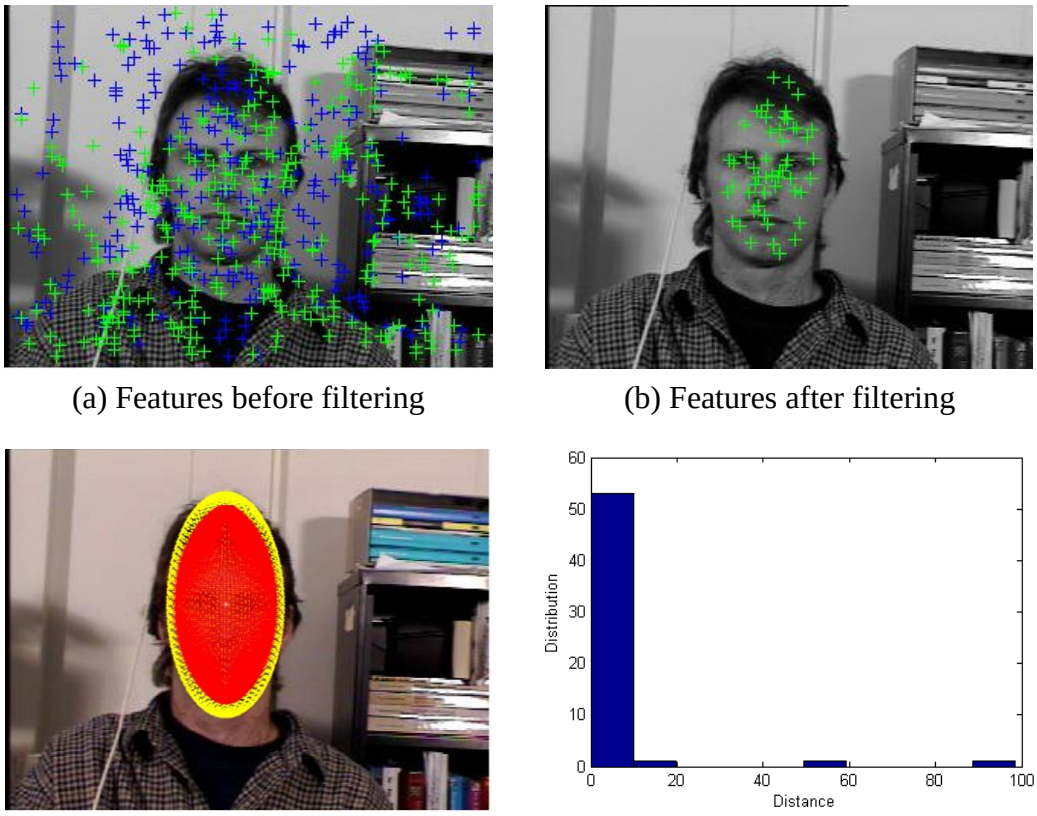
$$\begin{aligned} \mathbf{x}_2^{\mathrm{T}} \mathbf{F} \mathbf{x}_1 &= \mathbf{0} \\ \mathbf{E} &= \mathbf{K}'^{\mathrm{T}} \mathbf{F} \mathbf{K} \\ \mathbf{E} &= [\mathbf{t}]_{\times} \mathbf{R} \end{aligned} \tag{2.4}$$

where **F**,**K**,**R**,**E** represent the fundamental, camera, rotation and essential matrices, respectively. $[\mathbf{t}]_{\times}$ is the matrix representation of the cross product with **t**.

The optical flow can only be calculated precisely when the texture of the images is clear. Also, the twist representation method calculates the motion vector based on angular approximation in 3D space. Using Epipolar geometry could be unstable if the corresponding points used to calculate the fundamental matrix lie on the same plane. Accordingly, none of the above methods is suitable for our application, motivating us to apply a new way to estimate the head pose. First of all, the model-based feature extraction method is described. Then, a motion vector is estimated based on an objective function which describes the relationship between reconstructed 3D points and 2D corresponding points by using non-linear optimisation. Finally, the error is corrected by the analysis of the 2D representation of the 3D model.

Once the initialisation process is complete, which requires manual specification of the rotation and translation of the model in the first frame, the invalid features should be removed. Figure 2.9(a) shows the extracted SIFT features. The green crosses depict the SIFT points which are matched to the following frame whilst the blue crosses represent the unmatched points. The corresponding points selected by the SIFT descriptor are refined by a region-based and a distance-based measure because misalignment could occur since matching with only the SIFT descriptor considers the pattern around the extracted SIFT point. Therefore, first, the SIFT matching points only within ± 50° from the centre point (α = 90°, β =0° in the ellipsoidal model shown in figure 2.1) are considered. We assume that the region is a confidential region since the reconstructed 3D position of SIFT point from the outside of the region can be inaccurate (The 3D model is ellipsoidal). Figure 2.9(c) displays the region. The yellow region is the fitted model, the red region shows the region within ± 50° from the centre point, and the white cross in the model is the centre point. As shown in the figure, the red region can cover the facial components (eyes, nose and mouth) which hold rich information concerning head pose.

The second is to filter the invalid matching points based on the distances between each pair of potential corresponding points. The distances can show the distribution of the head translation movement in 2D image plane. In a histogram of the distances, the misaligned points can be easily detected and removed. In an alternative representation, the distribution of the histogram can be modeled as Gaussian ($X \sim N(\mu, \sigma^2)$). The SIFT points are taken between μ-2σ and $\mu+2\sigma$, covering 68.2% of the Gaussian distribution. Figure 2.9(d) shows the distance distribution from each pair of matched SIFT points. The final valid features are shown in figure 2.9(b).



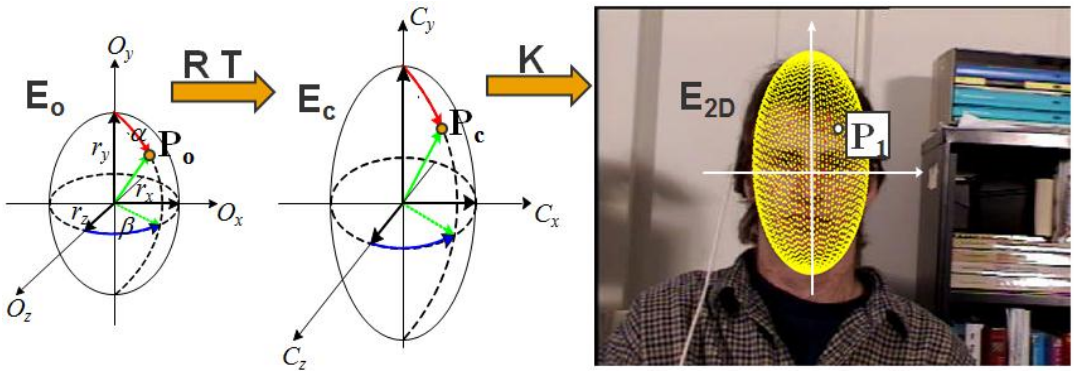
(a) Features before filtering (b) Features after filtering

(c) Confidence region (d) Distance histogram

Figure 2.9: Feature filtering

2.3.2 3D Position Reconstruction using Direct Mapping

After feature filtering, the 3D positions of SIFT points in the object coordinates are reconstructed by direct mapping. As shown in figure 2.10, the 3D point on the ellipsoidal model is mapped to the 2D point in the image via rotation, translation, and projection.



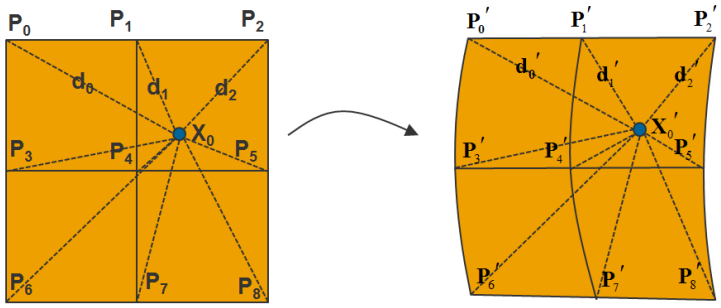
(a) object coordinate (b) camera coordinate (c) image coordinate

Figure 2.10: Direct mapping from 2D image plane to 3D space

In figure 2.10, a point $\mathbf{P}_o$ is mapped to the point $\mathbf{P}_c$ by changing from the object coordinates to the camera coordinates. Then, it is mapped to the point $\mathbf{P}_1$ in the image plane by projection using a camera matrix. Conversely, if the point $\mathbf{P}_1$ is known the point $\mathbf{P}_o$ (in the object coordinates) can be found directly. Therefore, assuming that the point $\mathbf{P}_1$ is an extracted SIFT point, the 3D position in the object coordinate can be calculated.

Practically, the ellipsoidal model $\mathbf{E}_o$ and $\mathbf{E}_c$ have 360 × 360 (spherical) points. For the projected ellipsoidal model $\mathbf{E}_{2D}$ only 180 × 360 points are valid due to invisible parts. An extracted SIFT point could be one from the model $\mathbf{E}_{2D}$ or between points. Therefore, to reconstruct an exact 3D position in the model $\mathbf{E}_o$ we approximately linearise the position using the distance ratio between the extracted SIFT point and around nine points on the model. First, the closest point of the model to the SIFT point is found, and the nine points around the SIFT point are

chosen. Then, the distance ratios between the SIFT point and around the points are calculated. The same procedure is applied to find 3D position in the object coordinates as the ratio should be consistent. The corresponding points in the object coordinates for all the 2D points can be found. Then, using the distance ratio, the approximate 3D position of the 2D SIFT points can be calculated.



(a) (b)

Figure 2.11: Extracted SIFT point and around points in (a) 2D and (b) 3D

Figure 2.11 shows this relationship: $\mathbf{X}_o$ and $\mathbf{P}_n$ are an extracted SIFT point and the nine points in the model $\mathbf{E}_{2D}$. $\mathbf{X}_o'$ and $\mathbf{P}_n'$ are the corresponding points in the model $\mathbf{E}_o$. Let the distance between $\mathbf{X}_o$ and $\mathbf{P}_n$ be d_n; thus

$$r_k = (1/d_k) \Big/ \sum_{k=1}^{n} 1/d_k \quad \text{then} \quad \sum_{k=1}^{n} r_k = 1 \tag{2.5}$$

The 3D position $\mathbf{X}_o'$ can be calculated by

$$\mathbf{X}_o' = \sum_{k=1}^{n} r_k \cdot \mathbf{P}_k' \tag{2.6}$$

From the above relationship the 3D position is reconstructed and this 3D position will be used to calculate the motion vector in the next Section.

2.3.3 Motion Vector Calculation using Non-linear Optimisation

The Levenberg-Marquardt algorithm is one of the non-linear optimisation algorithms which can provide the numerical solution to minimise an objective function. It can be thought of a combination of Gauss-Newton and Gradient Descent optimisation. When the current solution is far from the correct one, the algorithm behaves like a Gradient Descent method; slow, but guaranteed to converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method [77].

To explain the Levenberg-Marquardt algorithm the Gauss-Newton method is introduced. Let the objective function (or error function) be defined by the following equation,

$$S(\mathbf{v}) = \sum_{i=1}^{m} \left[y_i - f(x_i, \mathbf{v}) \right]^2 \tag{2.7}$$

where (x_i, y_i) represents given independent and dependent variables and **v** is the initial parameter vector which need to be optimised.

The Levenberg–algorithm Marquardt uses an iterative procedure to find the minimum. In each iteration step, the parameter vector, **v**, is replaced by a new estimate, **v**+**δ**. To determine **δ**, the functions $f(x_i, \mathbf{v})$ are approximated by a first order Taylor series

$$f(x_i, \mathbf{v} + \boldsymbol{\delta}) \approx f(x_i, \mathbf{v}) + J_i \boldsymbol{\delta} \tag{2.8}$$

where Jacobian **J** is the gradient of *f* with respect to **v**.

To determine a minimum of $S(\mathbf{v})$ the gradient of the object function with respect to **δ** should be zero.

$$S(\mathbf{v} + \boldsymbol{\delta}) \approx \sum_{i=1}^{m} \left[y_i - f(x_i, \mathbf{v}) - J_i \boldsymbol{\delta} \right]^2 \tag{2.9}$$

It results in the following relationship of the vector notation for the Jacobian matrix **J**:

$$(\mathbf{J}^{\mathbf{T}}\mathbf{J})\boldsymbol{\delta} = \mathbf{J}^{\mathbf{T}}[\boldsymbol{y} - \boldsymbol{f}(\mathbf{v})] \tag{2.10}$$

This equation is the Gauss-Newton method. In the Levenberg-Marquardt algorithm, the normal equations $(\mathbf{J}^T\mathbf{J})\boldsymbol{\delta} = \mathbf{J}^T[\boldsymbol{y} - \boldsymbol{f}(\mathbf{v})]$ are replaced by the augmented normal equations $(\mathbf{J}^T\mathbf{J} + \lambda\mathbf{I})\boldsymbol{\delta} = \mathbf{J}^T[\boldsymbol{y} - \boldsymbol{f}(\mathbf{v})]$ for some value of λ that varies between iterations where $\mathbf{I}$ is the identity matrix. When λ is very small, the method is essentially the same as the Gauss-Newton iteration. On the other hand, when λ is large, then the normal equation matrix is approximated by $\lambda\mathbf{I}$, and the normal equation become $\lambda\boldsymbol{\delta} = \mathbf{J}^T[\boldsymbol{y} - \boldsymbol{f}(\mathbf{v})]$. This is similar to the Gradient Descent method.

Thus, to use the Levenberg-Marquardt algorithm, we define the objective function to extract the motion information in the following equation:

$$\arg\min_{\mathbf{R}\,\mathbf{T}} \sum_{k=1}^{n} \left\| \mathbf{p}'_{2d,k} - \mathbf{K}(\mathbf{R}\mathbf{p}_{3d,k} + \mathbf{T}) \right\| \tag{2.11}$$

where $\mathbf{p}'_{2d,k}$ are the SIFT points in the next frame, and $\mathbf{p}_{3d,k}$ is the reconstructed 3D points from the 2D SIFT points in the current frame. The rotation matrix ($\mathbf{R}$) contains $\Delta\theta_x$, $\Delta\theta_y$, $\Delta\theta_z$ and the translation matrix ($\mathbf{T}$) contains Δx, Δy, Δz. n is the total number of the pairs. So, the initial vector $\mathbf{v}$ in equation 2.7 contains these six parameters ($\Delta\theta_x$, $\Delta\theta_y$, $\Delta\theta_z$, Δx, Δy, Δz) and in the first frame these parameters are chosen manually to fit the model to face. In each iteration step, the initial vector $\mathbf{v}$ is changed into $\mathbf{v}+\boldsymbol{\delta}$. The initial translation and rotation matrices in the first frame are given manually.

In equation (2.11), $\mathbf{p}_{3d,k}$ is a position of point in the object-oriented coordinate and $\mathbf{p}'_{2d,k}$ is a position of SIFT point in the image coordinate. Also, we assume the camera matrix ($\mathbf{K}$) is given. Basically, equation (2.11) determines the rotation and translation matrices which minimise the distance between matched points in 2D space. The term $\mathbf{K}(\mathbf{R}\mathbf{P}_{3d,k} + \mathbf{T})$ in equation (2.11) converts the point from the 3D space coordinate into 2D image space. The motion information is extracted by the Levenberg-Marquardt algorithm. In this way, the rotation and translation matrix can be updated using the previous motion information for each image frame.

2.4 Error Correction

2.4.1 Correction by Optical Flow

An error correction module is necessary since the model used is approximate (the assumption that the head shape is ellipsoidal) and an accumulated tracking error and initialisation error can occur.

First, assuming that the difference between the potential motion vectors calculated previously and the real motion vector is quite small, optical flow can be used to correct the motion vector.

Under small motion variation and no illumination change the velocity relationship between 2D and 3D motion can be described as the following equation [47],

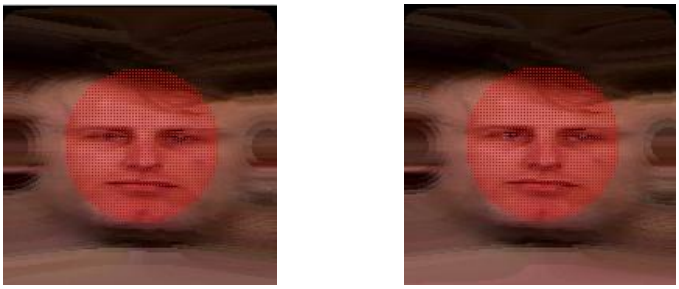
$$\frac{1}{Z}\left[\, fI_x \;\; fI_y \;\; -(xI_x + yI_y)\right]\mathbf{R}\left[\mathbf{I} \;\; -[\mathbf{P}_o]_\times\right]\begin{bmatrix}\Delta\mathbf{t}\\ \Delta\mathbf{r}\end{bmatrix} = -I_t \qquad (2.12)$$

where I_x, I_y and I_t are image-intensity gradients with respect to x, y, and t respectively. $\mathbf{P}_o$ is a point in the object coordinates. $\mathbf{R}$, $\Delta\mathbf{t}$, and $\Delta\mathbf{r}$ are the 3D rotation matrix and instantaneous translation and rotation, respectively. $[\mathbf{I} \;\; -[\mathbf{P}_o]_\times]$ is a matrix formed by concatenating $\mathbf{I}$ and $-[\mathbf{P}_o]_\times$. $[]_\times$ denotes a skew-symmetric matrix. $\mathbf{I}$ is the identity matrix and f is the focal length.

To deploy equation (2.12) the rotation matrix $\mathbf{R}$ is changed into $\mathbf{R_p}$ which is the calculated rotation matrix described in Section 2.3.3. Thus, the instantaneous rotation vector can be small. Therefore, even though there is huge variation between motion vectors, equation (2.12) can be used to correct the potential motion vector. A linear equation can be made by adapting equation (2.12) to all visible pixels. Then, a least squares solution is used to calculate the translation and rotation vector.

2.4.2 Correction by Texture Map

Another correction method is based on the texture map. Our model is the 3D ellipsoidal model whose resolution is one degree. Once the model is fitted to the image plane a 2D texture map can be obtained by unfolding the 3D ellipsoidal model per frame.



(a) frame 1 (b) frame 2

Figure 2.12: Unfolded images

Figure 2.12 shows the unfolded image in the range of $0° \leq \alpha \leq 180°$, $-90° \leq \beta \leq 90°$. This is only dependent on angles so that the image size is 180×180 pixels regardless of the actual head size. Ideally, if the frame rate is high enough, the adjacent unfolded images could have the same texture.

Using these texture maps, a 2D similarity transformation matrix between the adjacent images can be calculated. The procedure is similar to that used to calculate the 3D motion vector by using the corresponding SIFT points and non-linear optimisation. First, the corresponding points are extracted using SIFT on the confidence region (± 50° from the centre point). Then, the parameters of the similarity transform are calculated by using non-linear optimisation. When $\mathbf{x}$ and $\mathbf{x}'$ are corresponding points, the similarity transform ($\mathbf{H}_s$) and the objective function are described as follows [29],

$$\mathbf{x}' = \mathbf{H}_s \mathbf{x} = \begin{bmatrix} s_{2D}\mathbf{R}_{2D} & \mathbf{T}_{2D} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x} \tag{2.13}$$

$$\arg\min_{\mathbf{R}_{2D},\mathbf{T}_{2D},s_{2D}} \sum_{k=1}^{n} \left\| \mathbf{x}'_k - \mathbf{H}_s \mathbf{x}_k \right\| \tag{2.14}$$

where $\mathbf{H}_s$ is a 3×3 homogeous matrix, s_{2D} is a scaling factor , $\mathbf{R}_{2D}$ is a 2×2 rotation matrix, $\mathbf{T}_{2D}$ is a translation 2- vector, and $\mathbf{0}$ is a null 2-vector.

There is a relationship between the 2D motion vector and the 3D motion vector. Let the 2D and the 3D motion vectors be $\mathbf{M}_{2D}$ and $\mathbf{M}_{3D}$:

$$\mathbf{M}_{2D} = [s_{2D}, t_{2Dx}, t_{2Dy}, \theta_{2D}] \tag{2.15}$$

$$\mathbf{M}_{3D} = [t_{3Dx}, t_{3Dy}, t_{3Dz}, \theta_{3Dx}, \theta_{3Dy}, \theta_{3Dz}] \tag{2.16}$$

The corrected motion vector is

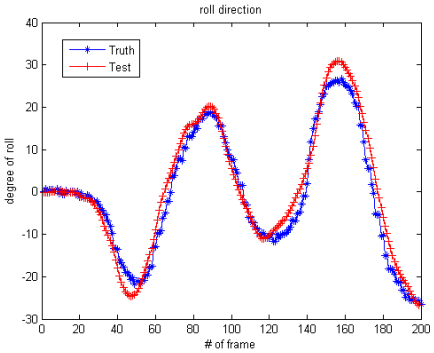
$$\begin{aligned} \mathbf{M}'_{3D} = [&t_{3Dx}, t_{3Dy}, t_{3Dz} \times s_{2D}, \\ &\theta_{3Dx} + t_{2Dy}, \theta_{3Dy} + t_{2Dx}, \theta_{3Dz} + \theta_{2D}] \end{aligned} \tag{2.17}$$

Here, the *y* axis translation in 2D (t_{2Dy}) is a variation in the direction of the *x* axis rotation in 3D space, the *x* axis translation in 2D (t_{2Dx}) is a variation in the direction of the *y* axis rotation in 3D space. The scaling factor (s_{2D}) is directly matched to the *z* axis translation.

2.5 Experimental Results

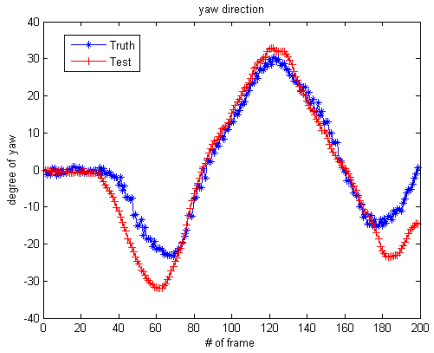
To verify the performance of face tracking we used the Boston face database [24] which has 45 image sequences plus the ground truth of 3D head pose. The experimental procedure is as follows. First, corresponding SIFT points are extracted and 3D points in the first frame are calculated by given a motion vector. Then, the potential motion vector is calculated using non-linear optimisation between the 2D SIFT points and the corresponding 3D SIFT points. From every motion vector, the unfolded image is generated from the fitted ellipsoidal model. Then, the final motion vector is corrected by optical flow and the similarity matrix.

Subject Jam1

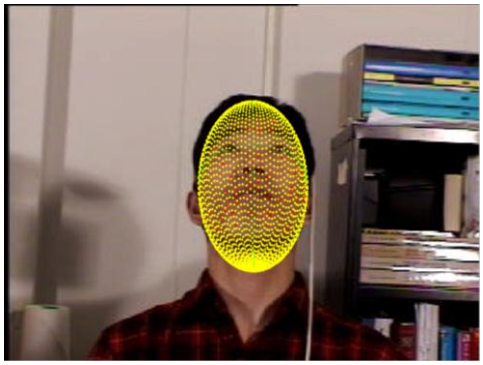


Roll direction tracking

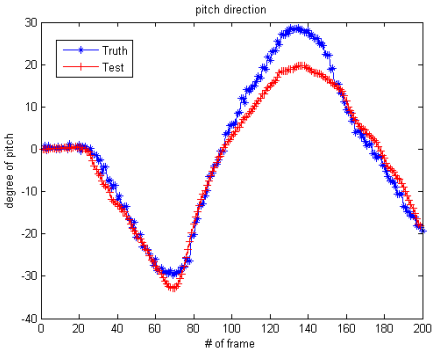
Subject Jim1



Yaw direction tracking



Subject Llm6



Pitch direction tracking

Figure 2.13: Examples of tracking results of roll, yaw, and pitch directions; the first column shows the fitting results and the second column displays the tracking results

TABLE 2.5: Error of the each direction

Subject	Roll (°)	Yaw (°)	Pitch (°)	Average (°)
Jam1	2.57	3.96	2.28	2.93
Jam2	1.08	2.73	1.72	1.84
Jam3	1.50	2.47	2.64	2.20
Jam4	1.34	2.39	1.76	1.83
Jam5	1.43	2.74	2.97	2.38
Jam6	2.47	2.55	8.83	4.61
Jam7	1.74	2.26	1.27	1.75
Jam8	3.51	4.37	3.59	3.82
Jam9	3.34	7.15	1.56	4.01
Jim1	1.39	7.21	2.93	3.84
Jim2	0.68	2.41	4.06	2.38
Jim3	0.91	0.90	2.99	1.60
Jim4	1.61	1.44	3.56	2.20
Jim5	1.65	1.95	3.19	2.26
Jim6	2.73	5.19	2.46	3.46
Jim7	2.93	3.07	3.39	3.13
Jim8	2.01	3.54	4.69	3.41
Jim9	1.93	7.61	3.19	4.24
Lim1	1.91	3.17	2.71	2.59
Llm2	1.75	2.95	5.35	3.35
Llm3	2.13	3.59	4.37	3.36
Llm4	2.06	7.89	4.80	4.91
Llm5	5.17	2.58	3.85	3.86
Llm6	1.97	2.48	2.92	2.45
Llm7	1.28	1.29	1.62	1.39
Llm8	2.25	2.14	2.19	2.19
Llm9	2.35	4.18	4.77	3.76
Ssm1	1.35	3.68	2.64	2.55

Ssm2	0.45	1.93	4.55	2.31
Ssm3	1.09	2.48	3.04	2.20
Ssm4	3.23	5.57	5.22	4.67
Ssm5	1.94	3.64	5.09	3.55
Ssm6	2.31	1.39	4.91	2.87
Ssm7	2.15	5.90	3.20	3.75
Ssm8	3.54	1.48	4.00	3.00
Ssm9	2.98	1.92	3.24	2.71
Vam1	1.05	1.41	1.85	1.43
Vam2	2.11	3.80	2.11	2.67
Vam3	1.89	4.42	4.72	3.67
Vam4	2.44	4.92	6.58	4.64
Vam5	4.81	5.86	5.28	5.31
Vam6	1.59	5.55	3.07	3.40
Vam7	2.69	5.92	6.03	4.88
Vam8	1.83	4.17	7.39	4.46
Vam9	1.85	3.12	4.45	3.14

TABLE 2.6: Comparison with previous methods

	Proposed method	An's method [47]	Xiao's method [26]
Database	45 image seq. [24]	45 image seq. [24]	45 image seq. [24] + 5 image seq. [26]
Illumination	Uniform light	Uniform light	uniform (45 seq.) and varying (5 seq.) light
Model	3D ellipsoidal model	3D partial ellipsoidal model	3D cylinder model
Image size (pixels)	320×240	320×240	320×240
Average error (Roll,Yaw,Pitch)	2.1°, 3.6°, 3.7°	2.8°, 3.9°, 4.0°	1.4°, 3.2°, 3.8°

In figure 2.13, the first column shows the fitting result and the second column displays the tracking result for an image sequence. In the first column, the yellow dots represent the model and the red dots are the selected SIFT features. In the second column, the blue line represents the ground truth and the red line shows the test results. Table 2.5 shows the error in degrees for each direction for each subject. The translation error is not displayed here because the Boston face database does not provide the camera information. In addition, the object translation can be covered by the later gait trajectory model in Chapters 3 and 5. The average error across the whole database in each direction is shown in table 2.6. The difference within the comparator results can be viewed as illusory in the sense that tracking results can depend on initialisation. All methods show good performance (less than 4° error in each direction). However, the accuracy of the comparator results may be compromised by the desire to achieve video-rate processing and these methods assume there are small variations of head pose between frames.

2.6 Conclusions

We can estimate a 3D head pose by SIFT-based local feature matching and a 3D ellipsoidal model. We introduced region- and distance-based feature refinements. In addition, we generated a 3D feature position by the direct mapping method. The approaches have been demonstrated with the Boston face database showing that the tracking error was less than 4°. The method differs from the previous methods in two aspects. First, we calculated 3D information from 2D information only using a single camera. A further difference is that we could remove the assumption in the 3D position calculation that there should be a small variation between images. However, this result is confined in the constrained environments. To generalise this method to moving people we need to use their gait information.

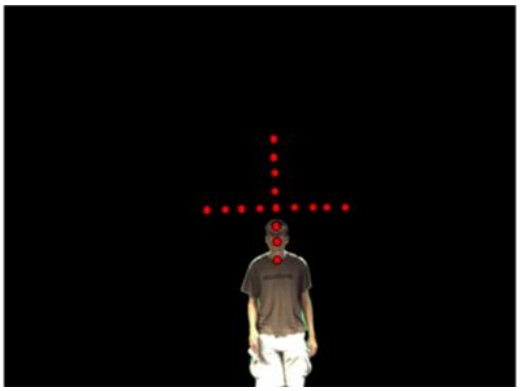
Chapter 3

Face Region Estimation by 2D Gait Trajectory

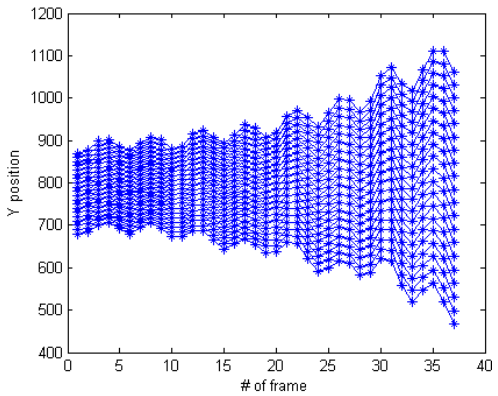
3.1 Looming Field

From the previous Chapter, the head pose can be calculated using a 3D face model in a controlled environment particularly when the distance between the object and the camera is fixed. In reality, however, in visual surveillance environments, there are many changes in illumination and large head movements. Also, a low frame rate could affect a detection rate. For example, the head movement might not be continuous and the illumination could change frame by frame, especially in a low frame-rate video. Unfortunately, most public CCTV installations have the above limitations. To overcome these difficulties we use not only the head pose but also use alternative biometric information: gait.

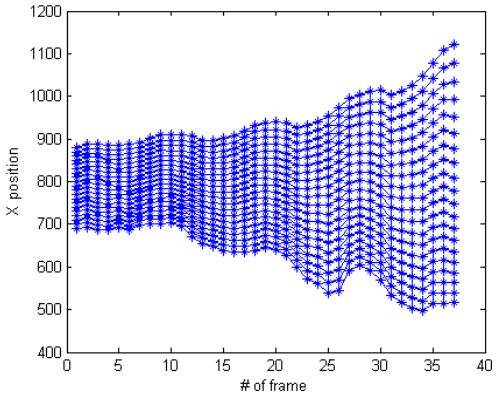
Before describing the face region estimation method by a gait trajectory we need to explain a looming effect. The looming effect occurs when a subject walks towards a camera resulting in a non-linear increase in apparent subject size [31].



(a) Chosen points



(b) The trace of *y* position at the fixed *x*



(c) The trace of *x* position at the fixed *y*

Figure 3.1: Tracking results at the specific points

Under the pinhole camera model, the relationship in perspective projection is following [30]; $x = fX/Z$, $y = fY/Z$ where (x,y) is a point in image plane, (X,Y,Z) is a point in 3D space, and f is focal length. Therefore, when the walking position Z is changed the projected position (x,y) is changed in non-linear way.

To show this phenomenon we extract a trajectory around the head in the following way. First, the background subtraction image is extracted [32], then the corresponding SIFT points between adjacent images are extracted. The same feature refinement methods are applied as in Section 2.3 (the region- and distance-based methods). To determine the potential trajectory of a specific point we

calculate the 2D Homography relationship for each adjacent image. Then, as shown in figure 3.1(a), the *x* position is fixed in the first frame and then, the successive corresponding *y* axis trajectories are extracted by using the Homography matrices. Also, the *y* position is fixed and the trajectories of the successive corresponding *x* positions are found. Figure 3.1(b) and (c) show the trajectories of the above points. These are affected by the looming effect otherwise it is supposed to be shown in linear increase. During walking, from some position, here we call 'looming centre', the trajectory can vary in non-linear manner.

The trajectory of the looming centre shows the constant variation (the gradient is around zero) and the variation according to the frame number shows dependency proportional to the distance between the looming centre and the chosen points. For example, the position of a point with a lower position than the centre decreases. Conversely, the higher position of the centre increases. In the case of figure 3.1, we found that the looming centre is located in around (780, 800) in the *x* and *y* axes respectively. Note that the looming centre is not the centre of the image (the image size is 1280 × 1024 pixels).

3.2 Gait Feature Extraction

To understand the looming effect, we need to describe the gait trajectory. When a person is walking the movement of the head must be large and sinusoidal [33]. When a person walks in the first half of the gait cycle, the hip is in continuous extension as the trunk moves forward over the supporting limb. In the second half, once the weight has been passed onto the other limb, the hip flexes in preparation for the swing phase. As such, specific points such as human joints also show sinusoidal variation in position [34].

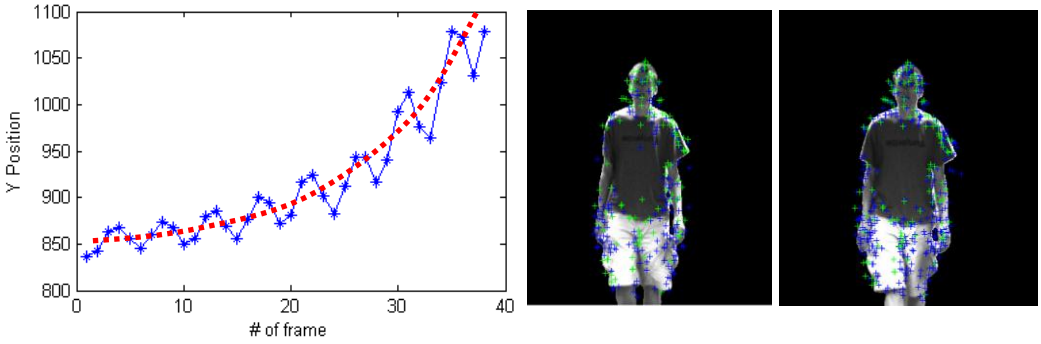
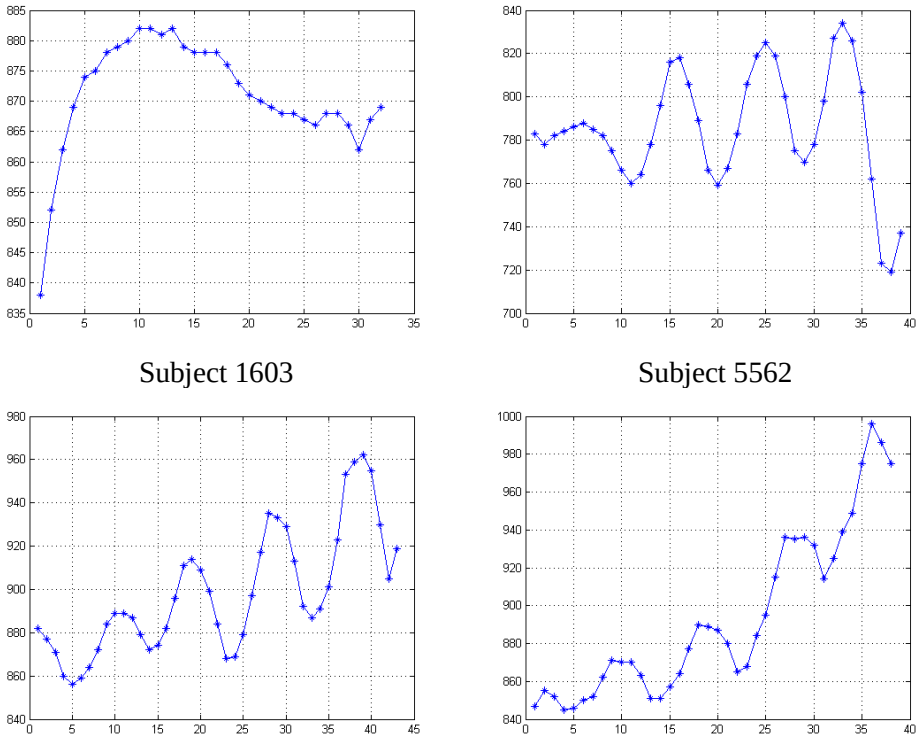


Figure 3.2: A sample gait trajectory and SIFT points of the human body



Subject 1603 Subject 5562

Subject 2584 Subject 5181

Figure 3.3: Horizontal variation of the neck point

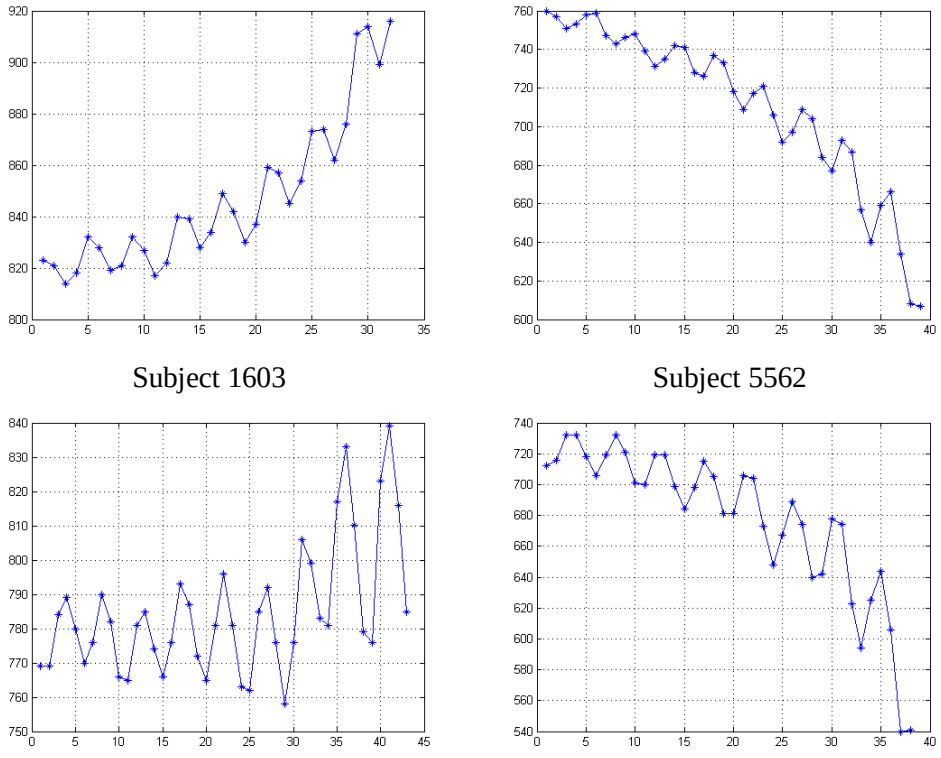


Figure 3.4: Vertical variation of the neck point

Figure 3.2 represents the trajectory of the head position according to each frame. As the person walks towards the camera, the variation of the head movement in the vertical (*y*) direction increases. It also reveals that there is periodic movement in the *y* direction. To clarify these facts we extract the exact human trajectory using some manually chosen points such as the points below the neck or the points in the chest. We use 36 samples (26 males and 10 females, with around 40 images in each sequence) chosen randomly from the Biometric tunnel database and extract the corresponding points for all frames. Figure 3.2 also shows the results for the corresponding points between two images where the green points represent the corresponding points. Here, we could choose the centre point immediately below the line joining the two labeled neck points.

Figures 3.3 and 3.4 show the results for the horizontal and vertical trajectories, respectively. As shown in these figures, the vertical gait trajectory has a consistent trend. Its nature depends on the distance between the looming centre and the position of the tracked pixel. Unlike the vertical trajectory the horizontal gait trajectory changes with a subject's gait. Also, the variation between each point in the vertical trajectory is much larger than the variation in the horizontal trajectory shown in figure 3.4. Therefore, we shall ignore the variation of the horizontal gait trajectory. In the next section we shall generate a model of the vertical gait trajectory and show the performance of model fitness using a non-linear optimisation method.

3.3 2D Gait Trajectory Model Definition

To define the gait trajectory model we use the following assumptions:

1. The variation of the *z* direction (walking direction) should be much larger than that in the *x* and *y* directions.
2. The walking speed is constant.
3. The sampling rate is constant.
4. The testing area does not change.

In Section 3.2 the gait trajectory can be divided into two parts: a periodic factor and a scaling factor. Under the above constraints, the periodic factor increases or decreases in the same ratio; however, the scaling factor is unknown. Therefore, the gait trajectory model could be defined in the following way, where the vertical position $y(t)$ is a function of gait frequency ω and $p(t)$, $r(t)$ are the periodic factor and the scaling factor. First of all, the general case is

$$y(t) = p(t) \times \sin(\omega t + \theta) + r(t) \tag{3.1}$$

As mentioned before, under the pinhole camera model, the relationship in perspective projection [30] is

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} fX / Z \\ fY / Z \end{pmatrix} \tag{3.2}$$

where (x, y) is a point in the 2D image plane, (X, Y, Z) are the coordinates in 3D space, and f is focal length. This is a conversion between 3D camera coordinates and 2D image coordinates.

Assuming that a person is walking towards a camera at a consistent speed we can approximate the relationship between the walking direction and the time linearly:

$$Z(t) = \mu - \delta t \quad \mu, \delta > 0 \quad Z(t),\ t \geq 0 \tag{3.3}$$

where $Z(t)$ represents the distance from the camera, μ is a track length, δ is a walking speed, and t is time or frame number.

In 3D space, the height of the human is constant and the trajectory of the upper body shows a periodic function. Therefore, the vertical trajectory can be modeled,

$$Y(t) = C_1 \sin(\omega t + \theta) + C_2 \tag{3.4}$$

Substituting equations (3.3) and (3.4) into equation (3.2), we can obtain a simple relation as shown in equation (3.5),

$$y(t) = \frac{k_1}{\alpha(1 - t/\lambda)} \sin(\omega t + \theta) + \frac{k_2}{\alpha(1 - t/\lambda)} \tag{3.5}$$

where θ is the initial phase, λ is a total time, $\lambda = \mu / \delta$ and C_n ,k_n are constants. Therefore, the scaling and the periodic factor can be defined from equation (3.5).

Equation (3.5) describes the general case of gait trajectory. It describes a sine wave with non-linear magnitude; however, this equation does not handle the case when a tracked point is located around the looming centre (in figure 3.1, the gradient around the looming field is about zero). Around the looming centre, the

scaling factor can be modeled as

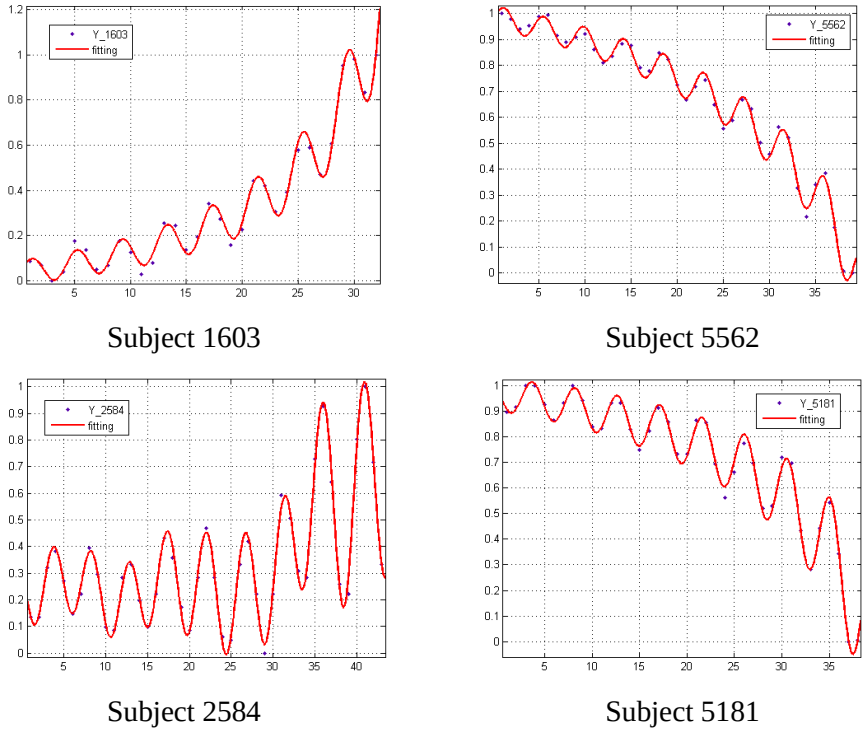
$$scaling \quad factor : g_c(t) = C_o t + I_o \tag{3.6}$$

where C_o is almost zero and I_o is the initial vertical position of the trajectory. Generally, average adult walking velocity on level surfaces is approximately 80 *metres*/*minute*. For men, it is about 82 *m*/*min*, and for women, about 79 *m*/*min* [37]. While investigating the gait trajectories we can find that one period of gait trajectory has four to five gait trajectory points in the case of using 10 FPS camera. For example, in figure 3.2, there are four or five sampling points in a gait cycle. So, we set the constraint of gait frequency between 1/5 and 1/4 because the size of the biometric tunnel is around 6 *m*.

To evaluate the performance of the model we normalise all of the extracted gait trajectories in order to express the error as a percentage. After fitting using the Levenberg-Marquardt algorithm, R-squared and Sum of Squares Error (SSE) are evaluated (equations (3.7) and (3.8)). Figure 3.5 shows the model fitting results for a typical gait trajectory. The blue points are the gait trajectory and the red line is the result of model fitting. We also calculate the autocorrelation of fitting error in figure 3.6 which shows no periodic components for the error. Since the gait trajectory has a periodic factor, we need to verify whether the error contains a period factor.

The experimental result shows that there is no specific pattern of the noise. Also, another advantage of this model is that it can express the trajectory between the trajectory points. Table 3.1 shows the errors of model fitting for 36 subjects from the Biometric tunnel database. The average value of the R-squared fitting metric for all samples is greater than 0.98, and the average for the SSE error metric is 0.037. So, we consider that the model correctly describes the gait trajectory.

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{3.7}$$



Subject 1603 Subject 5562

Subject 2584 Subject 5181

Figure 3.5: Fitting results between the actual data and the model

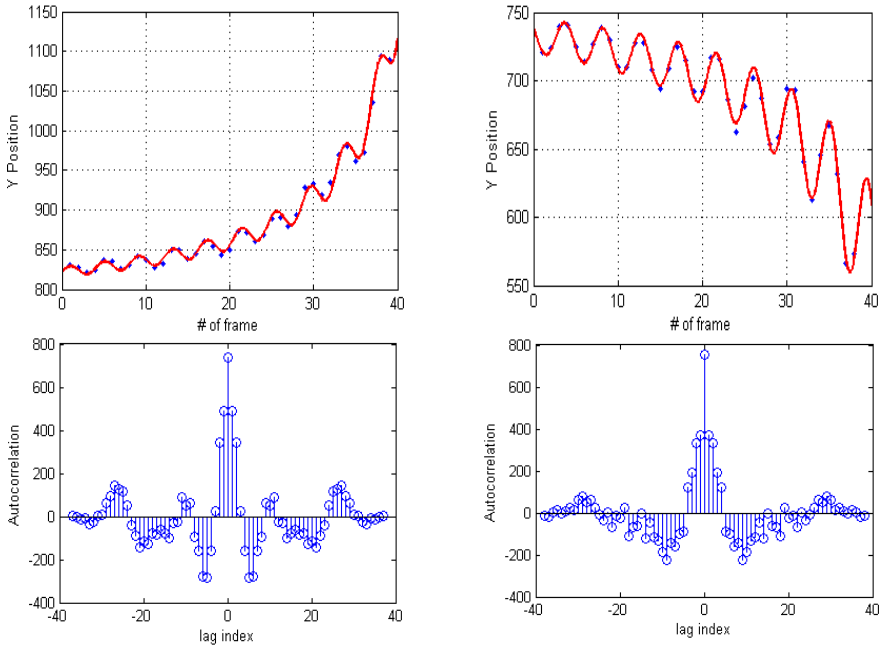


Figure 3.6: Samples of the fitting result and autocorrelation of the error

$$R-squared = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.8}$$

where y_i and $\hat{y}_i$ are the normalised values of the ground truth and the test result at i^{th} position of gait trajectory, respectively. $\bar{y}$ is the mean of the observed data.

TABLE 3.1: Database performance analysis

Subject	SSE	R-squared	Subject	SSE	R-squared
1603	0.0183	0.9930	6932	0.0724	0.9647
2345	0.0857	0.9526	2359	0.0866	0.9503
2393	0.0031	0.9985	2415	0.0843	0.9578
2431	0.0238	0.9868	5129	0.0370	0.9835
2452	0.0757	0.9720	395	0.0255	0.9883
2512	0.0292	0.9832	2441	0.0130	0.9892
2575	0.0952	0.9474	2465	0.0425	0.9866
2589	0.0203	0.9892	2481	0.0035	0.9974
2635	0.0247	0.9865	2498	0.0123	0.9931
2653	0.0236	0.9886	2522	0.0260	0.9858
4992	0.0459	0.9748	2584	0.0194	0.9915
5019	0.0067	0.9971	2671	0.0563	0.9618
5080	0.0619	0.9734	5271	0.0366	0.9868
5181	0.0111	0.9953	5303	0.0129	0.9946
5194	0.0640	0.9662	6052	0.1167	0.9073
5252	0.0162	0.9911	6005	0.0107	0.9950
5461	0.0052	0.9975	6182	0.0310	0.9836
5562	0.0131	0.9954	6929	0.0074	0.9955

3.4 Experimental Results

In initialization, the Homography matrix is calculated for each frame using the matched SIFT points. Then, a specific point such as the centre of the face, or the neck, is chosen manually so that a potential trajectory can be obtained. After that, the gait trajectory model is applied to the potential trajectory.

Unlike the potential trajectory, the constructed gait trajectory is continuous and can be interpolated. Since the Homography relationship and the trajectory between frames are known, the approximate face region of each frame can be extracted to initialise the height and width of the face region. In tracking, first the 3D ellipsoidal model was applied in the first frame using a given motion vector. In the 2D projection area from a 3D ellipsoidal model the SIFT points can be extracted so that the outliers which result in mismatching can be removed. For the next frame, the face is tracked using a calculated motion vector in the previous frame. Also, the motion vector is calculated using the valid area mentioned above.

By this procedure, we tested the 36 sequences from the Biometric tunnel database. Figure 3.7 shows the experimental results. The first row per subject in figure 3.7 shows the result of extracting the approximate face regions by using the gait trajectory. The second row shows the tracking results of the 3D model fitting in image sequences. As shown in figure 3.7, the background changes; however, the centre of the face and the size of the face are not changed as the subject walks. The face region can be extracted regardless of pose variation, illumination change, and low resolution. The face region is clearly extracted well, especially the comparison with the data from which it was extracted (figure 1.2(b)). Note in particular the fourth and fifth images where the pose of the subject changes considerably instead of looking forward. The subjects look right or downwards. Despite this, there is still an excellent fit of 3D model to the data.

Figure 3.7: Examples of the approximate face region by gait trajectory and fitting results according to the frames

To confirm the effectiveness of using the gait trajectory for face acquisition we test the face detection ratio with and without the gait trajectory model. Basically, the 3D ellipsoidal model cannot be applied directly to the raw image due to the image resolution. To use the ellipsoidal model, a minimum image size should be guaranteed. So, in the above experiment we resize the approximate face region. An alternative way is to deploy the Viola-Jones front face detection system [57] implemented using the OpenCV library [79]. We test three cases; i) face detection ratio with raw images using [57], ii) one with the approximate face regions using [57], and iii) one using only the gait trajectory. Table II shows the results of face detection.

TABLE 3.2: Test results with/ without the gait trajectory

	1st test	2nd test	3rd test
Image size (pixels)	1600×1200 (Raw image)	300×300 (Approximate face region)	1600×1200 (Raw image)
Aalgorithms	Viola & Jones face detector	Viola & Jones face detector	New method
Detection rate (%)	80.8% (957/1185)	94.0% (1114/1185)	98.0% (1161/1185)

As shown in Table 3.2, using the gait trajectory shows a better face detection performance. Comparing all the tests (with and without the gait trajectory), it shows a 17.2% improvement beyond the use of Viola-Jones face detection. The second test shows detection aided by the gait trajectory. Ideally, the results of first and second test should show the same result. However, even though the resizing ratio is the same (the default is 1/1.2), the number of resized pixels differs. Therefore, this result also indicates the usefulness of the gait trajectory model. The differences between the second and the third test are caused by variation in head pose and illumination, confirming that the new method is indeed more robust than the conventional approaches.

3.5 Conclusions

We have proposed a new method for estimating the face region by a 2D gait trajectory model when a person walks towards a camera. Moreover, we included the looming effect in the gait trajectory model based on a characteristic of human gait: there is conspicuous sinusoidal movement in the vertical direction. We demonstrated the performance using the Biometric tunnel database. It shows that this method can extract the face region well regardless of head pose, image resolution and changes in illumination. Moreover, we can apply a method of 3D head pose estimation in Chapter 2 to the approximate face regions. Otherwise, there are many difficulties related to applying the head pose estimation method directly to raw data. This approach has the limitation in that the speed of the object should be constant and the walking direction is supposed be the same, although this method can be used in the constrained environments such as corridor and airport portal. In the next two chapters, we will describe a generalised gait trajectory model which can be applied in less constrained environments.

Chapter 4

Heel Strike Detection

4.1 Introduction

Heel strike detection is a basic yet important process in non-invasive analysis of people at a distance, especially for human motion analysis and recognition by gait in a visual surveillance environment. Since the gait periodicity, stride and step length can be calculated directly from the position of the heel strikes; this information can be used to represent the individual characteristics of a human, the walking direction, and the basic 3D position information (given a calibrated camera). This heel strike detection method will be used to extract the face region within an unconstrained environment in Chapter 5.

There are two key observations concerning heel strike detection. During the strike phase, the foot of the striking leg stays at the same position for nearly half a gait cycle (when the foot is in contact with the floor), whilst the rest of the human body moves forward [48]. Also, when the left and right feet cross, the head is at its highest position. Conversely, when the stride is at its largest, the head is at its lowest position in a gait cycle. We develop our new heel strike detection method based on these observations.

Generally, heel strike detection is a preliminary stage in gait recognition or model-based human body analysis and visualisation. Previously, two major strategies were used for the detection of heel strike. The first is the model-free strategy which uses low level data such as silhouettes and edges to detect gait motion. Bobick and Johnson [49] recovered static body and stride parameters of subjects using the action of walking to extract relative body parameters. BenAbdelkader et al. [50] identified people from low resolution video by estimating the height and stride parameters of their gait. Jean et al. [51] proposed an automatic method of detecting body parts using a human silhouette image. A five-point human model was detected and tracked. They solved for self-occlusion of the feet by using optical flow and motion correspondence. Bouchrika and Nixon [48] built an accumulator map of all corner points using the Harris corner detector [69] during an image sequence. Then, the heel strike position was estimated using the density of proximity of the corner points.

The alternative strategy is model-based; this uses prior information such as 3D shape, position and trajectory of body motion. The heel strike position is a part of this analysis. Vignola et al. [52] fitted a skeleton model to a silhouette image of a person. Each limb was fitted independently to speed up the fitting process. Zhou et al. [53] extracted full-body motion of walking people from video sequences. They proposed a Bayesian framework to introduce prior knowledge into a system for extracting human gait. Sigal and Black [54] estimated human pose using an occlusion-sensitive local image likelihoods method. Zhang et al. [55] presented a three-level hierarchical model for localising human bodies in still images from arbitrary viewpoints. They handled self-occlusion and large viewpoint changes using Sequential Monte Carlo optimisation. Sundaresan et al. [56] proposed a graphical model of the human body to segment 3D voxel data into different articulated chains.

Many approaches, however, consider only the fronto-parallel view of a walking subject wherein the subject walks in a direction normal to the camera's plane of view. Also, in the model-based approaches, there is much computational load in initialisation and tracking. Moreover, in the visual surveillance environment, the image quality could be low and the image sequences derived from a single camera only are available. Therefore, an alternative heel strike detection method is needed which is robust to low resolution, foot self-occlusion, and camera view point, and suitable for a single camera.

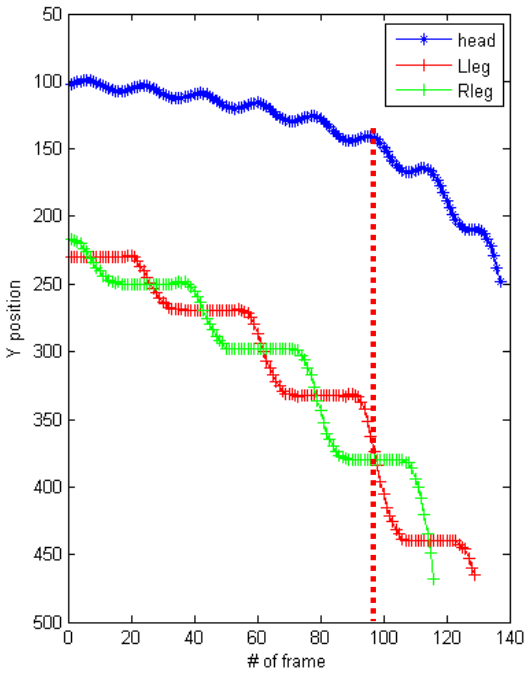
In this chapter, to overcome the above constraints, we propose a novel method of heel strike detection using the gait trajectory model. The frame in which the heel strike takes place can be extracted even when the foot is hidden by another leg and the image quality is low. The model is applied to detect the key frame in which the heel strikes occur. The silhouette image at the key frame is used to remove background data and determine the Region of Interest (ROI). Then, we calculate the heel strike position from the process of the candidate detection and verification.

4.2 Key Frame Calculation

The basic characteristics of gait are used to find the moment of a heel strike. When people walk there is conspicuous sinusoidal head movement in the vertical plane [33]. As mentioned above, the highest point in a gait cycle is when both feet cross (stance) and the lowest point is when the gait stride is the largest (heel strike). Therefore, the vertical position is a cue for gait cycle detection. We define the key frames when the highest and the lowest positions of the vertical gait trajectory in a gait cycle occur. In Chapter 3, we constructed the gait trajectory

model and fitted it with the actual data. After this procedure, we analyse the gait trajectory to determine the key frames.

The key frame can be calculated after the construction of the model. The highest points of trajectory are calculated from gait frequency f ($t = ((2n+1/2)\pi - \theta)/2\pi f$, where n is any integer). Figure 4.1 shows a sample of the key frame extraction process. In figure 4.1(a) the highest position of y is when the left foot and the right foot cross (here, the key frame number is 97). Figures 4.1(b) and (c) are the original and the silhouette image at the key frame. In figure 4.1(b) the head is in the highest position when the feet cross. In the next stage, the silhouette image is used for filtering the accumulator map to extract the ROI since the accumulator map is constructed by using all of the images in a sequence and the filtered accumulator map must contain at least one heel strike.



(a) Trajectories of y position for left, right feet and the head

(b) Original image at the key frame

(c) Silhouette image at the key frame

Figure 4.1: Key frame extraction

4.3 Heel Strike Detection

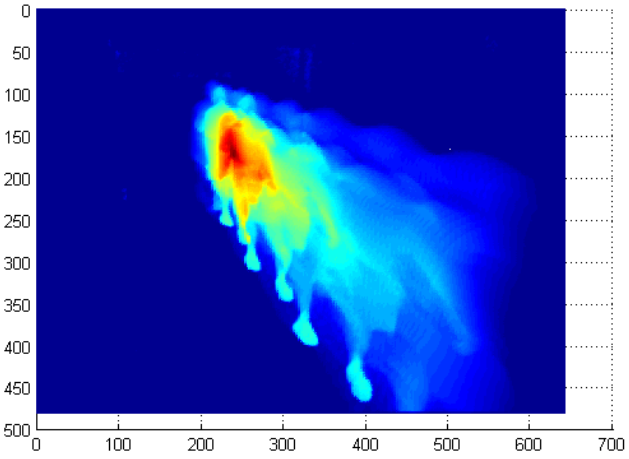
4.3.1 Heel Strike Candidate Extraction

As a pre-processing step, we calculate the silhouette image [32] from the intensity and the colour difference (between the background image and foreground image) at each pixel. Then, the accumulator map of a silhouette is the total number of silhouette pixels in the $(i, j)^{th}$ position. Low pass filtering is deployed to smooth the accumulator surface.

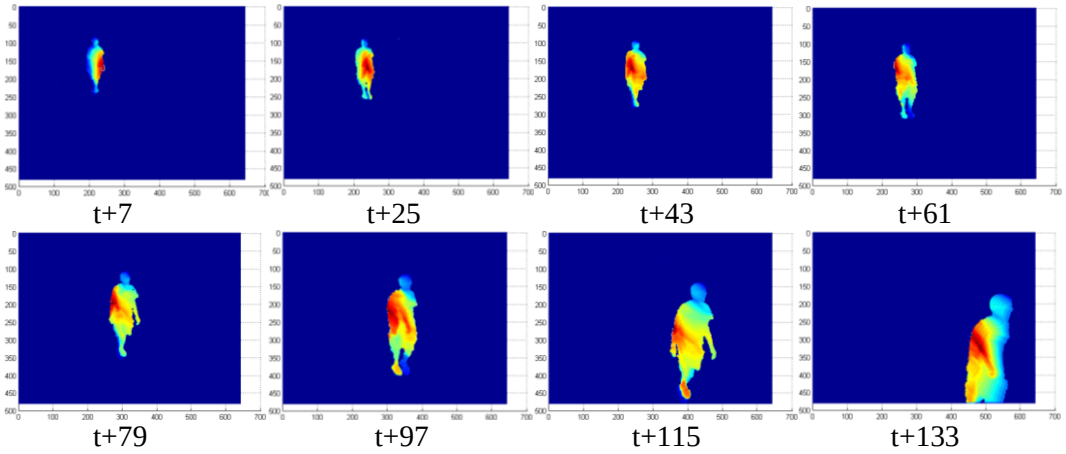
$$Accumulator(i, j) = \sum^{\#of\ images} Silhouette(i, j) \tag{4.1}$$

Figure 4.2 shows an accumulator map and filtering results by a key frame silhouette image. The colour in the figures indicates the number of silhouette pixels from blue (few) to red (many). As shown in figure 4.2(a), the heel strike region can clearly be distinguished from other body part regions.

The filtered accumulator map shows the distribution of the number of silhouette pixels. It reveals that the position of heel strike has a relatively higher distribution than other regions. Using the characteristic, we extract a Region of Interest (ROI) and smooth it before applying the Gradient Descent algorithm. Figure 4.3 shows the process for finding the heel strike positions from the accumulator map. The accumulator map shown in figure 4.3(a) is filtered by Gaussian function (filter size 12×12, $\sigma = 2.0$). Then, the approximate heel region, which is one eighth of a person's silhouette height from the bottom of silhouette, is extracted (figure 4.3(a)). Accordingly, the heel strike position can be extracted by Gradient Descent. Figure 4.3(b) shows the three dimensional view of the extracted ROI. Figure 4.3(c) shows the result of analysing the approximate heel strike region as shown in figure 4.3(b) using the Gradient Descent. The small arrow in the figure is the point where the orientation changes. Figure 4.3(c) shows convergence to the local maximum.

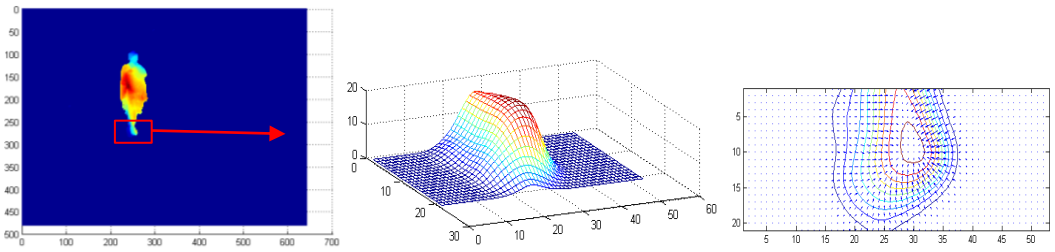


(a) Accumulator map for a walking image sequence



(b) Filtering result using silhouette image at the key frames

Figure 4.2: Accumulator map and filtering results



(a) ROI extraction (b) 3D view of ROI (c) Result of Gradient Descent

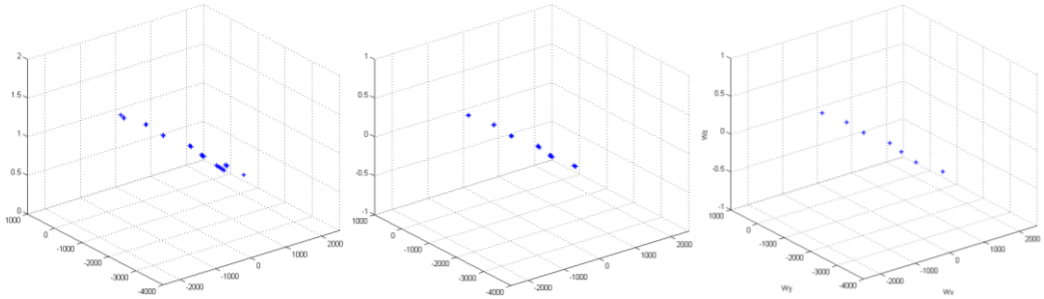
Figure 4.3: Procedure of heel strike candidate detection

4.3.2 Heel Strike Position Verification

This section describes a heel strike verification process. In our method the silhouette image is used when the feet cross, so it is possible to extract the candidates from the foot of the heel strike and also for another foot. For example, in the second heel strike of figure 4.4(a), the candidates are detected from both feet. To remove the invalid candidates, other key frames are selected when the vertical position of the gait trajectory is lowest within a cycle (figure 4.1(a)) and the same procedure is executed in Section 4.3.1 to find other heel strike candidates. Since the moment at the lowest vertical position is when the gait stride is largest, the feet are in contact with the floor, and the feet are separated, so the candidates from other key frames are considered as the potential heel strikes. These candidates are deployed to remove the invalid candidates. Simply, the distance between these two groups of candidates (at the highest and lowest y coordinates) is calculated. Then, the candidates within a fixed distance (here, 5 pixels) are selected from the group of candidates of lowest values for y. As shown in figure 4.4(b), after this filtering process, the invalid candidates from another foot are removed.

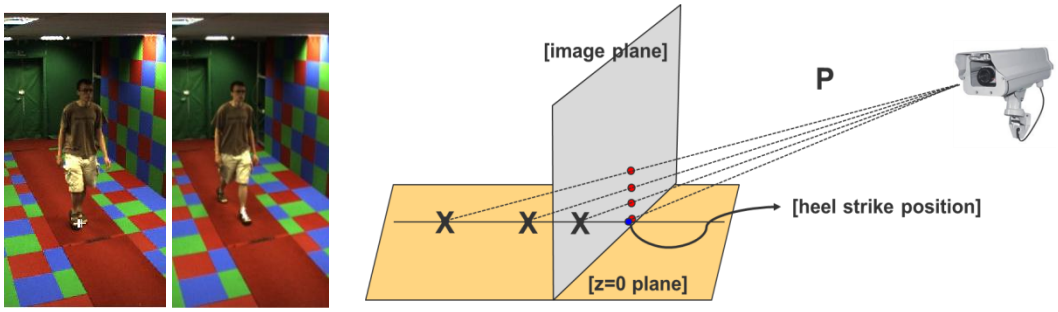
The accumulator map depends on the camera view. Once the camera is calibrated the invalid candidates could be removed using the back projection from a 2D image plane into a 3D world space. Using 3D projection the candidate which is the closest to the camera is selected. Since a single camera is used in our approach, we assume a ground floor is known, i.e. that $z = 0$ (the z axis is in the vertical position). This enables calculation of the intersecting points between the projection ray from 2D image points and the ground floor. The closest heel strike to the floor is considered as the final heel strike position, thereby filtering the invalid positions. Figure 4.4 shows a result of the filtering process. The invalid points in figure 4.4(a) are removed to give the final result in figure 4.4(c). Figure 4.5 shows the example of the candidate filtering. Figure 4.5(a) is the sample of

candidate extraction when the gait trajectory is the highest and the lowest in a gait cycle. Figure 4.5(b) describes the filtering method using the relationship between 3D coordinate and 2D image plane. The white crosses in figure 4.5(a) are the heel strike candidates.



(a) Heel strike candidates (b) Result of first filtering (c) Result of second filtering

Figure 4.4: Verification process



(a) Candidates filtering using two group's candidates (b) Candidate filtering using 3D projection from 2D image plane

Figure 4.5: Candidate filtering

Once 3D heel strike positions have been calculated we can estimate the direction and the position of walking. First, we define a walking direction vector ($\mathbf{i}_k$) between k^{th} heel strike position $\mathbf{H}_k$ and $k+1^{th}$ heel strike position $\mathbf{H}_{k+1}$

$$\mathbf{i}_k = (\mathbf{H}_{k+1} - \mathbf{H}_k) / \left\| \mathbf{H}_{k+1} - \mathbf{H}_k \right\| \tag{4.2}$$

Then, the position at ground plane $\mathbf{P}_k$ ($Z = 0$ plane) is

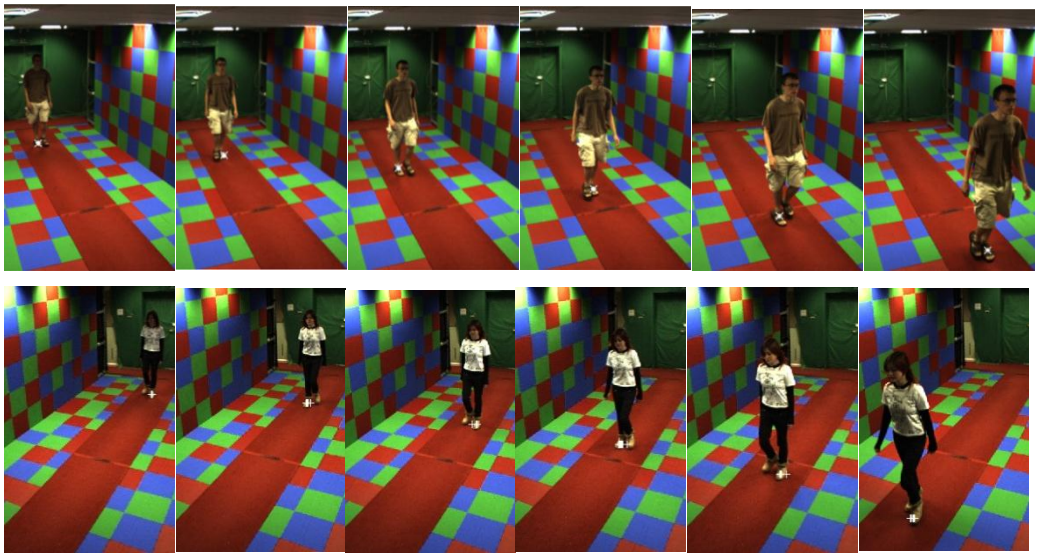
$$\mathbf{P}_k(t) = \mathbf{i}_k \times (stride\,length / sample\,number) \times t \tag{4.3}$$

4.4 Experimental Results

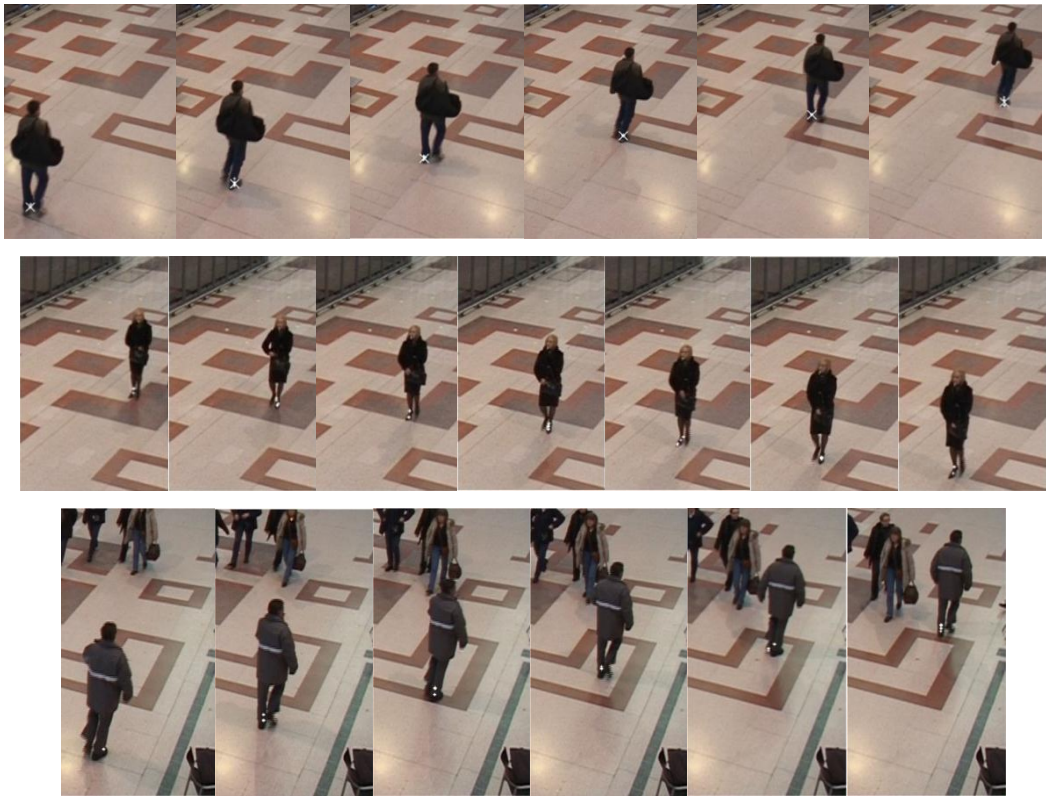
To evaluate the heel detection system, we test three different environments offered by visual surveillance databases. One is a controlled environment (the Biometric tunnel database [21]) and the others are uncontrolled databases (the PETS 2006 database [46] and the CAVIAR dataset [45]). The test set from the Biometric tunnel data consists of 25 samples (18 males and 7 females, with around 130 images in each sequence) and each sample has two views of image sequences. The database was recorded in a controlled environment (such as lighting illumination, walking direction and camera calibration). We choose 10 samples from an image sequence of the PETS 2006 dataset which is recorded at an indoor train station. In the dataset, each sample has a different walking direction with around 80 images in each sample sequence. The test set from CAVIAR dataset consists of 15 samples (11 males and 4 females, with around 100 images in each sequence) and are resized to 640×480 pixels. Each sample has a random walking direction. This database was recorded in a shopping centre corridor. To calculate a camera projection matrix we estimate (perspective) corresponding points based on provided ground truth position (15 pairs of corresponding points are used).

Figure 4.6 shows the detection results for three different environments. The white crosses in the figure represent the points detected as the heel strike positions. As shown in figure 4.6 the proposed method can detect the heel strike position regardless of the camera view since the method uses the basic characteristic of gait: its periodic factor. In the cases of samples from the Biometric tunnel data and the PETS 2006 data (figure 4.6 (a), (b)) the subjects walk in a series of straight lines. To confirm that our method works when the walking direction changes, the image sequences from the CAVIAR data are tested, where the subject is walking randomly along a corridor (figure 4.6 (c)). Even when the walking direction changes the new method still follows the position of heel strike.

Table 4.1 shows the test environment and the detection rate. A total of 470 heel strikes from 75 subjects were tested. For the evaluation of the performance we measured the distances between the test results and the ground truth. For the Biometric tunnel, given calibration, the detected 3D heel strike positions were compared with the ground truth. Unlike the Biometric tunnel, the PETS 2006 data does not provide camera information. Therefore, the detection rate is calculated based on 2D image coordinates. For the CAVIAR dataset, the camera projection matrix can be constructed using the provided coordinates in 3D space, but it does not provide ground truth in 3D space. For this, we applied the two filtering methods (Section 4.3.2, the 2D distance and 3D geometry measure) to the Biometric tunnel data and the CAVIAR data to remove the invalid candidates and to obtain 3D heel strike position. For the PETS2006 data we applied one filtering method which uses the 2D distance between the candidates of two groups, given the lack of camera calibration information. Besides that, to calculate the detection rate, we consider as ‘detected’ when the distance between 3D heel strike positions of the test results and the ground truth is within ±10 cm for the Biometric tunnel case. The others were counted as ‘detected’ when the distance is within ± 5 pixels in the 2D image coordinate.

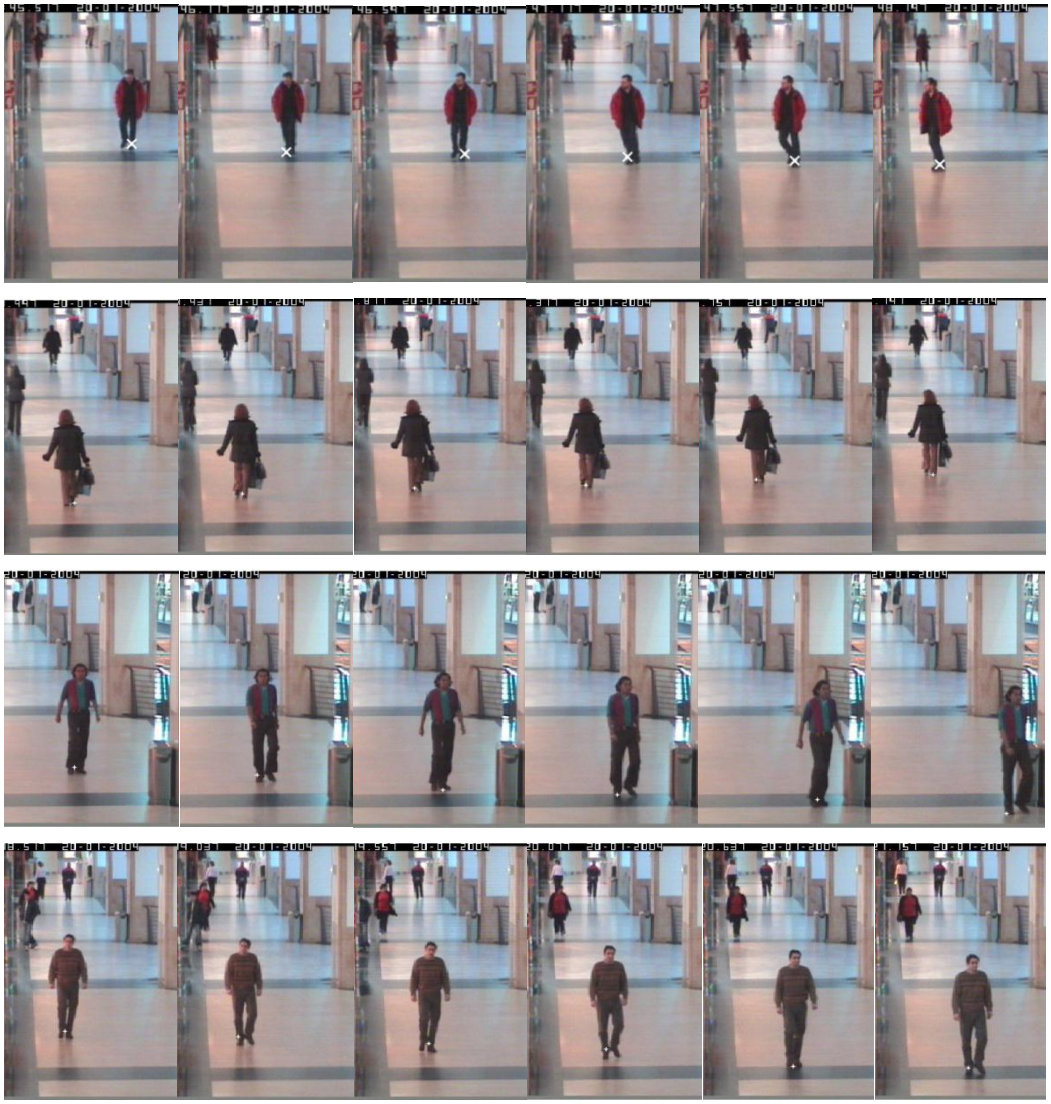


(a) Heel strike detection results of the Biometric tunnel data



(b) Heel strike detection results of the PETS 2006 data

(b) (*Continued*.) Heel strike detection results of the PETS 2006 data



(c) Heel strike detection results of the CAVIAR data

Figure 4.6: Heel strike detection results within different environments

TABLE 4.1: Test results of the different databases in heel strike detection

Database	Biometric Tunnel	PETS 2006	CAVIAR
# of subject	25× 2 views	10	15
Walking direction	Straight	Straight & Random	Straight & Random
Image size (pixels)	640×480	720×576	Resized to 640×480
Filtering method	2D & 3D filtering	2D filtering only	2D & 3D filtering
Environment	Controlled tunnel	Train station	Shopping centre corridor
Decision method	± 10 cm in 3D	± 5 pixels in 2D	± 5 pixels in 2D
Detection rate (%)	95.6 (285/298)	93.4 (57/61)	93.7 (104/111)

Table 4.1 shows the detection rate for all the databases. The detection rate for the Biometric tunnel database is slightly higher than for the PETS 2006 database and the CAVIAR dataset since the environment of the Biometric tunnel is more controlled. The overall detection rate is 94.9% (446/470).

4.5 Conclusions

To deploy automatic gait recognition in unconstrained environments, we need to develop new techniques for analysis. This section describes new techniques for heel strike estimation which are robust to self-occlusion and change in camera view. The approach to heel strike estimation combines human walking analysis with characteristics of the heel strike. The approach has been demonstrated on visual surveillance databases, and on one for biometrics, with a heel strike detection rate exceeding 94%. We make a strong assumption that the walking speed is constant and that the foot area can be seen. Also, since this method is based on the silhouette accumulator for all frames, it could be affected by the camera inclination. If the angle between the walking direction and the camera

view is large, the system error reduces. This model can be applied in many facilities such as corridors, lobbies, and the entrances of building. Also, it is a basic step in improving visual surveillance. As such, heel strike analysis can be used for basic gait analysis and derivation of walking direction estimation, and this approach provides a new and more generalised approach for surveillance environments.

Chapter 5

Face Region Estimation by 3D Gait Trajectory

5.1 Introduction

Determining the position of the head is a basic step in automatic face recognition and can also be used in privacy-aware surveillance (to mask the head region) or to improve person location in tracking and behaviour analysis applications. In visual surveillance there are many constraints on the ability to detect and recognise people. Depending on which CCTV technology is used, the illumination conditions, the frame rate, and the resolution can differ. In addition, if the Region of Interest (ROI) is considerably smaller than the captured image, conventional approaches could fail. Even though Viola's method [57] is well known for its ability to detect faces, the above constraints could result in possible failure when detecting people in a visual surveillance environment. An alternative is to use gait, and this is immediately beneficial since the body comprises a larger ROI than the human face.

There are many approaches to detect a human in visual surveillance. Face region detection is one sub-category of a human detection. Previous human detection methods can be classified into two main approaches. The first is a 3D-assisted approach which uses view geometry and a 3D human shaped model. Mohedano et al. [58] built a multi-camera geometry-based 3D tracker which uses multi-dimensional background subtraction and human template correlation. This method detected people even when they were occluded by static foreground objects. Li and Leung [59] defined the Human Perspective Context according to the camera tilt angle. Then, using Model Estimation-Data Tuning the human shape and head/foot positions were detected. Saboune and Laganiere [60] generated a human upper body 3D model and a likelihood function. Then, Explorative Particle Filtering was applied to detect people and for 3D tracking. In another approach, Jean et al. [61] used the 2D trajectories of both feet and the head extracted by using the silhouettes. After that, the fronto-parallel normalised view trajectory was generated from a homography transformation based on the 3D walking plane.

An alternative approach is a local feature-based approach which uses an object's information such as the pattern of a face or skin colour. Li et al. [62] estimated the number of people in surveillance scenes using a mosaic image difference-based foreground segmentation and Histograms of Oriented Gradients for head-shoulder detection. Yang et al. [63] detected basic human actions such as placing objects and pointing using a set of motion edge history images and tree-structured boosting classifiers. Leykin and Hammound [64] tracked a subject's body and estimated the visual attention field from head-pose estimation by combining a skin colour detector with the direction of motion. Chen and Chen [65] proposed a novel cascaded structure called meta-state to boost the performance of the AdaBoost detection algorithm.

Our approach has a different starting point. We focus on detecting the head region based on the characteristics of a human walking. In other words, we propose a gait-based face region detection method. First, we calculate the potential head trajectory between frames by using a Homography relationship. Wavelet decomposition is deployed to detect the component which contains a specific frequency of human walking. By analysing this component the gait cycle can be calculated. Based on the gait period and the (known) camera projection matrix, the heel strike position and walking direction in 3D are calculated by using the method in Chapter 4. After that, we define an objective function which can be used to fit the 3D gait trajectory model with the actual data by comparing a 2D potential gait trajectory with a projected 3D gait trajectory which minimises the error of objective function. In this way, we can estimate the region of the head in 3D space.

5.2 Gait Period Estimation

The gait period provides key information to generate heel strike position and 3D trajectory correction. In Chapter 4, we described the human walking characteristic; when a person walks there is conspicuous sinusoidal head movement in the vertical plane. The highest point in a gait cycle is when both feet cross and the lowest point is when the gait stride is the largest. Therefore, finding the highest and lowest points in the gait trajectory enables us to calculate the gait period. We use the same way of calculating the potential gait trajectory as in Chapter 3. First, the Homography relationship between the adjacent images is calculated and then the potential trajectory is extracted. By wavelet packets analysis [78], different frequency signals can be decomposed. These contain a signal with the same gait period as the original signal.

The Fast Fourier Transform (FFT) can analyse the main frequencies within a signal; although it can only be localized in frequency, not in time. In our assumption the walking speed can change so that the main walking frequency can be varied in the frequency domain. In other Autocorrelation is generally applied to find repeating patterns, such as a periodic signal corrupted by noise. This method is also not suitable here because the gait trajectory contains a non-linear scaling factor. Given a model, the frequency can be found using curve fitting. However, this method needs an accurate model in advance. Wavelet decomposition can be localised in both time and frequency so we use this method to detect the gait period. Since the walking period changes with the speed of walking, a more specific method is needed for generalised analysis. Ladetto et al. [66] reconstructed the gait walking signal using the wavelet decomposition to refine the signal. In this research we analyze more about the walking signal and use the method to detect the gait period.

Wavelet packets analysis can be considered as a combination of high frequency filter banks **h**(*k*) and low frequency filter banks **g**(k) [78]. A signal **S** can be divided into approximation coefficients $\mathbf{cA_j}$ and detail coefficients $\mathbf{cD_j}$ by the convolution of high and low frequency filters as shown in figure 5.1 where **c** is a constant. The 'meyer' type of mother wavelet [67] is used to construct the filter banks.

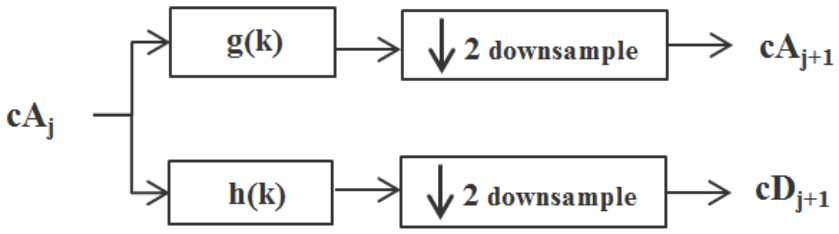
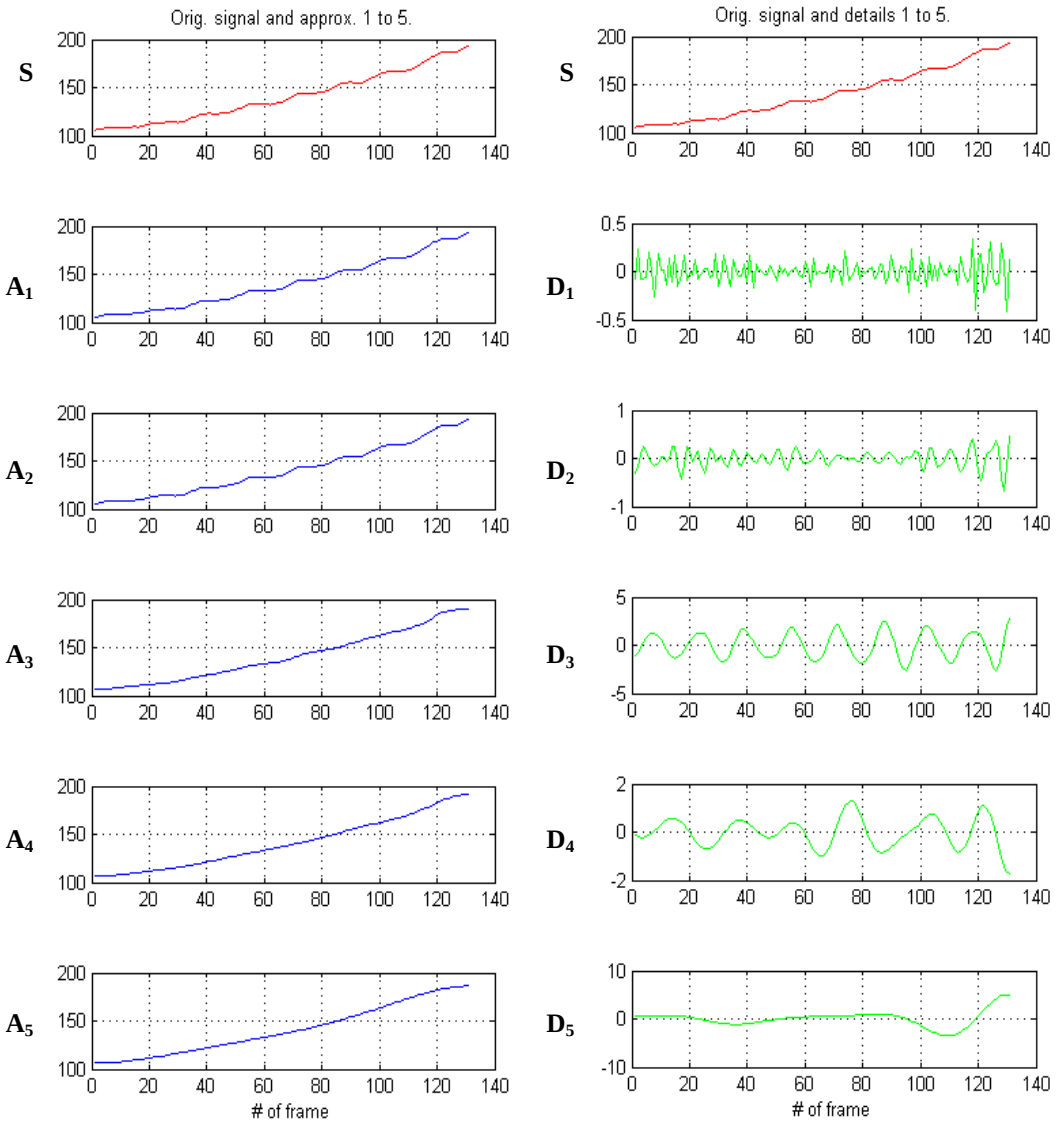


Figure 5.1: Wavelet packets analysis



(a) Approximate coefficients (b) Detail coefficients

Figure 5.2: Five scale full wavelet packets analysis

The signals $\mathbf{A_1}$-$\mathbf{A_5}$ and $\mathbf{D_1}$-$\mathbf{D_5}$ are the decomposition results of the original signal (signal **S**). As shown in figure 5.2, as the decomposition step becomes higher, the extracted signals become smoother. For example, the detail coefficient $\mathbf{D_1}$ contains the highest frequency and $\mathbf{D_2}$ has the second highest frequency

component. Generally, as previously mentioned, the average adult walking velocity on level surfaces is approximately 80 *m*/*min*. For men, it is about 82 *m*/*min*, and for women, about 79 *m*/*min*. Therefore, the walking signal has a fixed range of frequency so that the signal including the walking frequency can be extracted by wavelet decomposition. In this research, the frame number is chosen as a key frame when the highest and lowest points of the gait trajectory occur. Therefore, the gait period can be detected by finding the peak position of the signal in a particular frequency band which is suitable for passing the gait frequency.

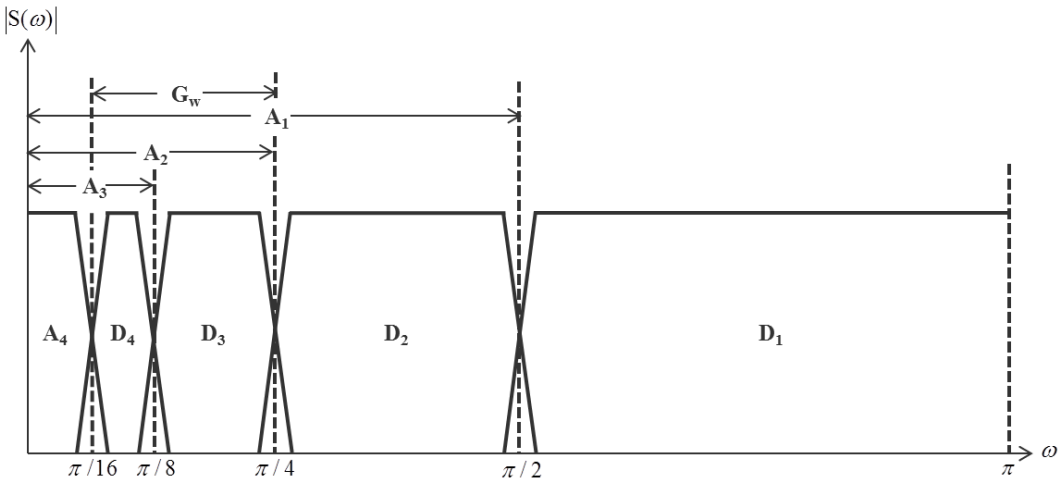
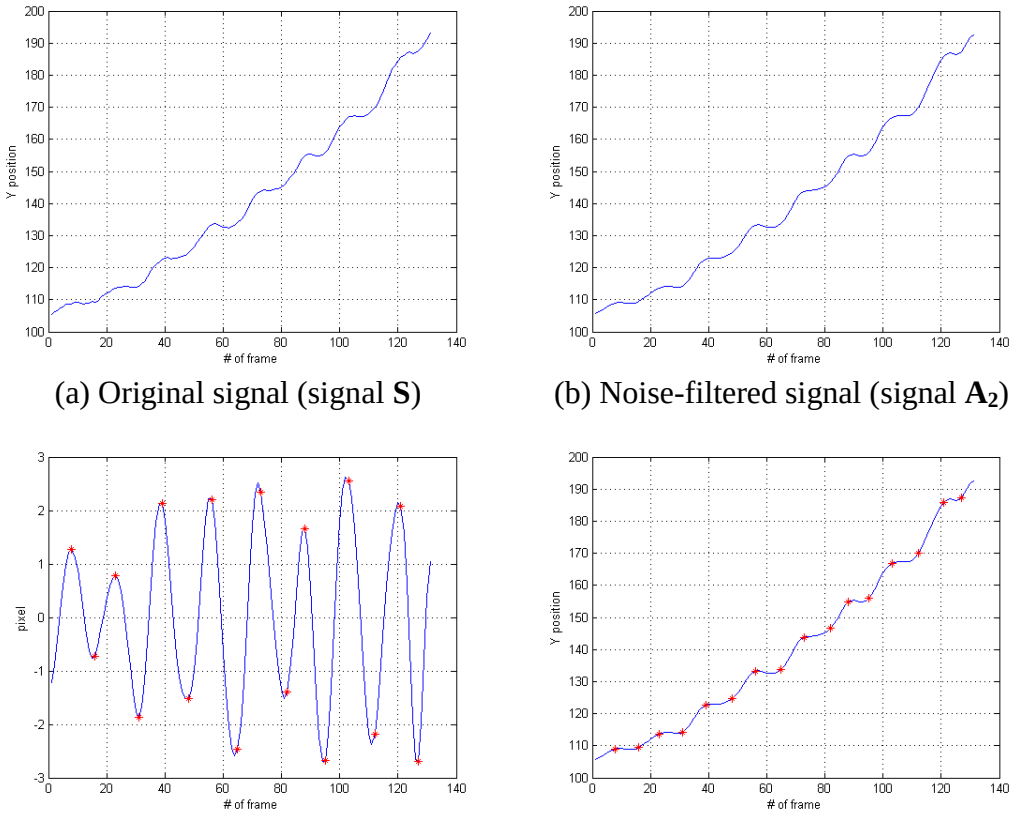


Figure 5.3: Spectrum splitting of gait trajectory

Figure 5.3 shows the results of the four-scale filter bank. From the original signal **S**, the sub-band signal can be divided. In this experimental environment the frame rate is 30 FPS and the human walking has a fixed range of speed. By experiments we can infer that a gait frequency $\mathbf{G_w}$ belongs to the sub-band frequency between signals $\mathbf{D_3}$ and $\mathbf{D_4}$ as shown in figure 5.3. Therefore, the gait period can be detected as follows. First, high frequency components (detail coefficients $\mathbf{D_1}$ and $\mathbf{D_2}$ in figure 5.2(b)) are removed from an original signal (signal **s**) since the original signal contains high frequency noise components. Then, the signal which contains the gait frequency is calculated. The signal (figure 5.4(c)) is calculated

by subtraction between the noise-filtered signal (signal $\mathbf{A_2}$) and the low frequency signal (signal $\mathbf{A_4}$). Thus, it only contains the gait frequency. Finally, from the signal, we can detect the maximum and minimum points per cycle when the gradient is zero since the signal is smooth and continuous. Figure 5.4 shows the procedure of the gait period detection from the original signal to the detection results. The red crosses in figure 5.4(c) depict the detected maximum and minimum points in a cycle. Figure 5.4(d) shows the points that correspond with those in figure 5.4(c).



(a) Original signal (signal **S**)

(b) Noise-filtered signal (signal $\mathbf{A_2}$)

(c) High frequency component (signal $\mathbf{A_2}$ - $\mathbf{A_4}$)

(d) Gait period per each cycle

Figure 5.4: Gait period detection procedure

To confirm the performance of the gait detection period we tested 30 samples from the Biometric tunnel database [21]. Each sample consists of around 120 frames. The camera used has a resolution of 640 × 480 pixels and captures at a rate of 30 FPS as shown in figure 1.2(c). The average distance between the ground truth and the calculated data is measured (the distance unit is a frame). We choose the ground truth at the moment when the feet cross and the stride is largest in a cycle. The average error of all samples is 1.48 frames. This means that the time difference between the actual heel strike and the calculated heel strike is 48.76 *ms* (the frame rate is 30 FPS). Considering that the heel strike stays on the floor for around five frames, the results infer that the method can be a way of calculating the gait period. This method is the first use of wavelet decomposition in calculating the gait period. Table 5.1 shows the time errors for each subject from the Biometric tunnel database.

TABLE 5.1: Time gaps between the actual heel strike and the calculated results

Subject	Frame difference (frame)	Time (*ms*)	Subject	Frame difference (frame)	Time (*ms*)
395	2.00	66.0	2584	1.38	45.5
1603	0.75	24.8	2589	0.64	21.1
2345	0.70	23.1	2635	1.00	33.0
2359	3.58	118.1	5019	1.50	49.5
2393	1.40	46.2	5080	1.58	52.1
2415	1.75	57.8	5181	4.13	136.3
2431	1.17	38.6	5194	1.42	46.9
2441	1.64	54.1	5252	1.80	59.4
2452	1.42	46.9	5303	1.60	52.8
2465	2.42	79.9	5562	1.67	55.1
2481	1.08	35.6	6005	1.58	52.1
2498	1.00	33.0	6052	1.17	38.6
2512	0.88	29.0	6929	1.17	38.6
2522	1.17	38.6	6932	0.64	21.1
2575	1.10	36.3	8440	1.00	33.0

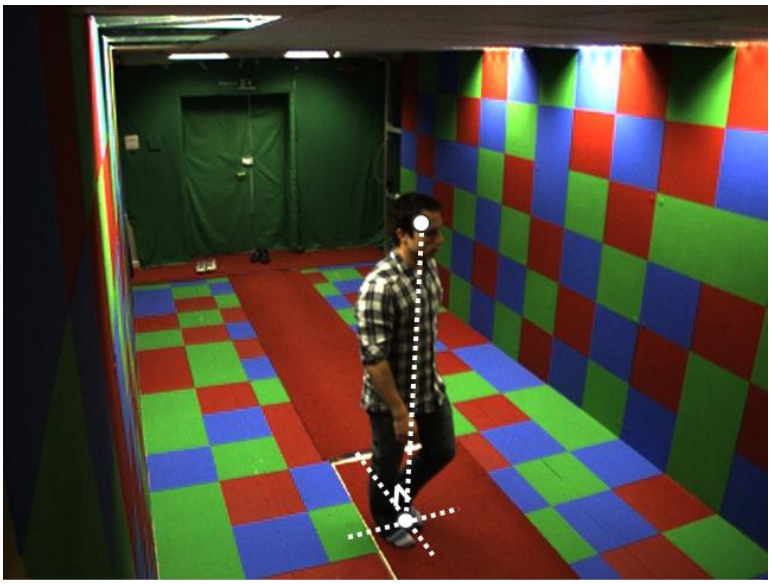
5.3 Trajectory Calculation in 3D Space

5.3.1 Head Position While Changing Walking Direction

The gait trajectory in 2D space contains a non-linear factor; the looming effect [68] which can affect the accuracy of the gait trajectory. Therefore, the domain is changed from the 2D image plane into 3D space using the camera projection matrix since in 3D space there are many advantages such that the height of walking person is constant and the looming effect can be ignored. In addition, it can provide not only the walking direction but also the walking speed. Therefore, in this section, a 3D gait trajectory model is built based on the pre-calculated 3D heel strike position and the gait period. Then, this model is fitted to actual data in order to estimate the 3D position of head using Levenberg-Marquardt optimisation.

Before defining the 3D gait trajectory, we need to investigate the relationship between walking direction and head position because the walking trajectory is not always the same as the trajectory of the head. For example, if a person changes the walking direction, a half gait cycle gap takes place between walking trajectory and head trajectory.

Generally, the head position is located directly above the heel strike position at the moment the feet cross. Figure 5.5(a) shows a sample of this relationship. In this figure, the lower white dot is the detected heel strike position and the higher white dot represents a potential centre position of the head. As shown in figure 5.5(a), the angle between the ground floor and the line of the two points is perpendicular. Figure 5.5(b) presents the head trajectory from all frames. This is one of the results discussed in the next section, where a 3D gait trajectory model is constructed and the 3D gait trajectory is projected into 2D space. The trajectory shows sinusoidal movement with a fixed height.



(a) Positions of head and heel strike

(b) Head trajectory

Figure 5.5: 3D head position when walking in different direction

t+80 t+85 t+90 t+95

Figure 5.6: Head position and walking direction change

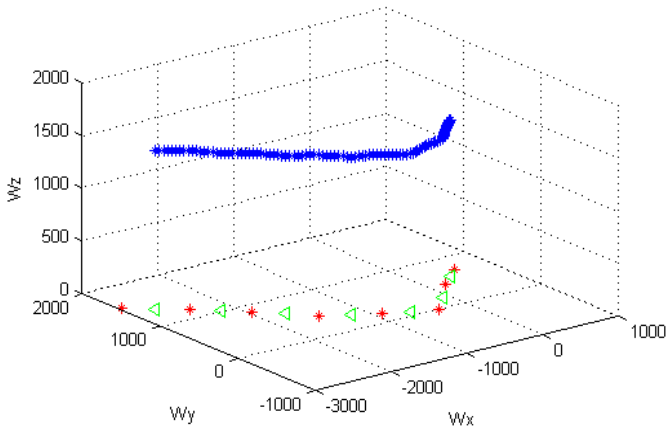


Figure 5.7: Heel strike position and the middle position

Another factor to be considered is when people change their walking direction. Figure 5.6 shows the moment of change in a walking direction. As shown in this figure, first, a person looks at the direction of walking. Then, a half gait cycle later, the direction of the feet changes. In other words, the moment that a head turns is when the gait stride is the largest. Therefore, the middle position of the 3D heel strike is used (assuming the middle position of heel strike is when a gait stride is the largest). Figure 5.7 presents a sample of the heel strikes and the middle positions. The blue asterisk point represents the 3D head position. The red asterisk and green triangle point are 3D heel strike position and the middle of heel strike, respectively.

5.3.2 3D Gait Trajectory Model

Given the gait periods and the 3D heel strike positions, a gait trajectory can be modeled by a series of simple sinusoidal waves which have the same magnitude and height. The potential 2D trajectory could be inaccurate since error could occur within the Homography matrix and the camera projection matrix. Therefore, the gait trajectory needs to be corrected. In this Section, we introduce a 3D gait trajectory model to fix the gait trajectory in the 3D space. Further, the head region can be estimated by fitting the model with actual data using the non-linear optimisation method.

The 3D gait trajectory model can start from equations (4.2) and (4.3). In Section 4.3 we define a walking direction vector $\mathbf{i}_k$ and position $\mathbf{P}_k$ at ground plane. Here, the world coordinate system is shown in figure 5.8(a). The vertical direction is *Z* direction and the main walking direction is along the *X* axis.

The vertical position can be defined as

$$Z_k(t) = C_1 \sin(2\pi f_k t + \theta_k) + C_2 \tag{5.2}$$

Let $\mathbf{G}_{3D,k} = (\mathbf{P}_k, Z_k)$

$$\mathbf{T}_{3D}(t) = \sum_{k=1}^{Num\ of\ Heel\ Strikes} \mathbf{G}_{3D,k}(t) \cdot R_k(t) \tag{5.3}$$

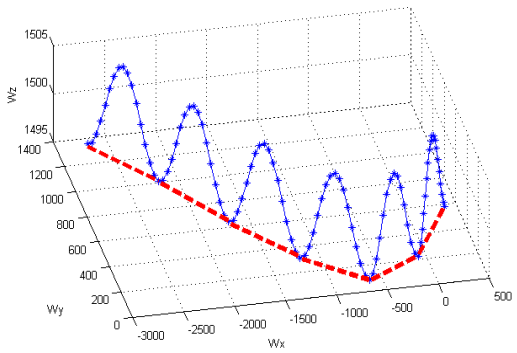
$$R_k(t) = \begin{cases} 1 & \sum_{n=1}^{k-1} p_n \le t < \sum_{n=1}^{k} p_n \quad when\ k \ge 2 \\ 0 & otherwise \end{cases} \tag{5.4}$$

The objective function is

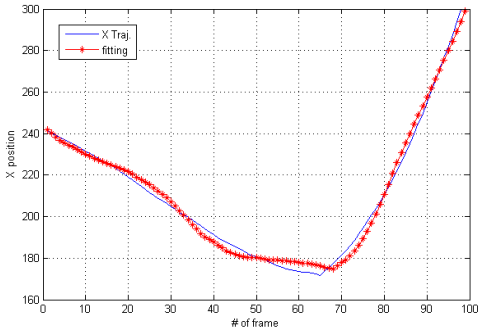
$$\arg\min_{C_1, C_2} \left\| \mathbf{T}_{2D}(t) - \mathbf{K} * \mathbf{T}_{3D}(t) \right\| \tag{5.5}$$

First, we define a 3D gait trajectory per gait period $\mathbf{G}_{3D,k}$ which consists of *x*, *y* position $\mathbf{P}_k(t)$ and *Z* position $Z_k(t)$ based on the gait period f_k and the middle heel

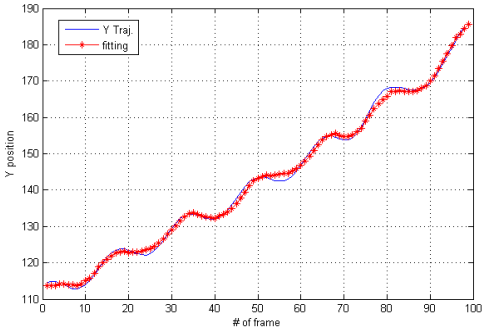
strike position $\mathbf{H}_k$. Then, the 3D gait trajectory $\mathbf{T}_{3D}$ is constructed by adding each gait trajectory per cycle. Equations (4.2), (4.3), and (5.2)-(5.4) represent the details of this procedure where p_n is a period of each gait cycle and R_k is a windowing function. The objective function is equation (5.5) where $\mathbf{T}_{2D}(t)$ is the potential gait trajectory in 2D image plane, $\mathbf{T}_{3D}(t)$ is the 3D gait trajectory from the model, and $\mathbf{K}$ is a camera projection matrix. The purpose of equation (5.5) is to calculate the magnitude C_1 and height C_2 in equation (5.2) which minimise the value of the objective function. The objective function is an error function between 2D potential trajectory and the projected trajectory from the 3D gait trajectory model. To calculate the parameters of the model in equation (5.5), the Levenberg-Marquardt algorithm is applied. After finding the magnitude C_1 and the height C_2 the 3D gait trajectory can be reconstructed.



(a) 3D gait trajectory model



(b) *x* position fitting



(c) *y* position fitting

Figure 5.8: 3D gait trajectory model fitting

Figure 5.8 displays the sample of the fitting results using the 3D gait trajectory model. Figure 5.8 (a) shows the reconstructed 3D gait trajectory. The blue line is the trajectory of the head and the red line is the direction of head movement. Figure 5.8 (b) and (c) show 2D image plane fitting after the optimisation procedure. The blue curve is the ground truth data which is selected manually and the red line is the fitting result.

5.4 Experimental Results

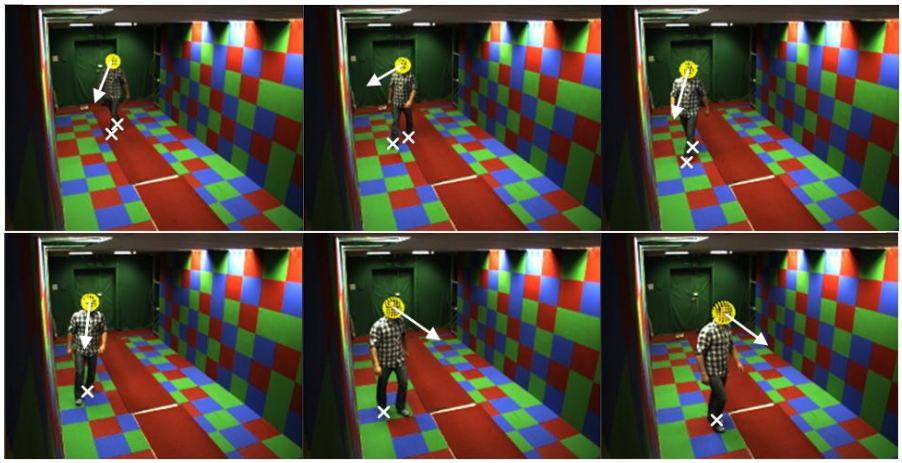
To evaluate the proposed 3D head region estimation method, we analysed the samples from the CAVIAR dataset [45]. The dataset consists of 23 samples (17 males and 6 females, with around 100 images in each sequence) and are resized to 640 × 480 pixels. Each sample has a random walking direction and speed. To calculate the camera projection matrix we estimate (perspective) corresponding points based on provided ground truth position (15 pairs of corresponding points are used).

Figure 5.9 shows the detection results with different environments; the Biometric tunnel [21] and a shopping centre from the CAVIAR dataset [45]. The yellow region is a 2D projected face region from the 3D estimated face region. The white cross is the result of back projection from the 3D heel strike position and the white arrow represents the walking direction. As shown in figure 5.9, the proposed method can estimate the head region regardless of the walking direction and speed since the method uses the basic characteristic of gait: heel strike position. Table 5.2 shows the errors between the potential 2D gait trajectory and the projected gait trajectory from 3D trajectory extracted by 3D gait trajectory model for 23 samples. We use Root Mean Square Deviation (RMSD) to evaluate the performance in equation (5.6). The average error in the x, y axis is 3.73 pixels

and 1.59 pixels, respectively as shown in table 5.1. Compared to the original image sizes, this value is under 1%, suggesting good accuracy.

5.5 Conclusions

This chapter describes new techniques for head region estimation in 3D space, which are less constrained than previous approaches and can handle any direction of walk, even away from the camera. The approach to head region estimation combines 3D geometry information with human walking characteristics. In other words, from the movement of the head we can estimate the heel strike, the walking direction, and the 3D head region. The new approaches have been demonstrated with the samples from the CAVIAR database. The results show the head region estimation method is accurate even with changes in walking direction and speed. As such, gait analysis can be used to derive the head region invariant to the view of the camera, leading to a more versatile method of finding the human head in surveillance applications.



(a) Sample results from the Biometric tunnel database

(b) Sample results from the CAVIAR database

Figure 5.9: Detection result with different walking direction

$$RMSD(\theta_1,\theta_2)=\sqrt{E((\theta_1-\theta_2)^2)}=\sqrt{\frac{\sum_{i=1}^{n}(x_{1,i}-x_{2,i})^2}{n}} \tag{5.6}$$

TABLE 5.2: Fitting errors

Object	X pos. error (pixels)	X pos. / image width (%)	Y pos. error (pixels)	Y pos. / image height (%)
Obj1	6.2780	0.9809	1.8997	0.3958
Obj2	5.9324	0.9269	2.3385	0.4872
Obj3	4.5952	0.7180	2.7985	0.5830
Obj4	2.6370	0.4120	1.8765	0.3909
Obj5	3.0777	0.4809	1.9592	0.4082
Obj6	4.8614	0.7596	1.5488	0.3227
Obj7	2.5864	0.4041	0.8252	0.1719
Obj8	8.2737	1.2928	1.5397	0.3208
Obj9	4.0387	0.6311	1.5106	0.3147
Obj10	2.8128	0.4395	1.1381	0.2371
Obj11	1.5620	0.2441	1.3598	0.2833
Obj12	1.7742	0.2772	1.1616	0.2420
Obj13	2.4818	0.3878	1.2760	0.2658
Obj14	3.0018	0.4690	2.3580	0.4913
Obj15	3.6817	0.5753	1.7399	0.3625
Obj16	2.2598	0.3531	0.9040	0.1883
Obj17	4.0278	0.6294	1.1021	0.2296
Obj18	5.8818	0.9190	1.8274	0.3807
Obj19	4.5899	0.7172	1.2580	0.2621
Obj20	2.0599	0.3219	1.6304	0.3397
Obj21	3.5919	0.5612	1.2405	0.2584
Obj22	3.8329	0.5989	1.7415	0.3628
Obj23	1.9018	0.2972	1.5309	0.3189
Average	3.7279	0.5825	1.5898	0.3312

Chapter 6

Frontal Face Extraction

6.1 Introduction

Face image extraction is a basic step in automatic face recognition. In the previous Chapters, we extracted the approximate face regions and calculated the position of the face based on the 3D ellipsoidal model and the gait trajectory model. Deployment depends on the imaging scenario. For example, the Boston database was recorded under uniform illumination and at fixed distance. Even though the variation of head pose is large, the frontal face can be extracted since every image sequence contains the front pose of the face image. Thus, in this database, the head inclination is the main factor to consider when registering face image. In the case of the Biometric tunnel database where a person walks towards the camera, the image resolution of the face changes with time. Normally, when the person walks, the head rotates less than ± 15° from the walking direction. Thus, the most important consideration is the resolution of the extracted face image rather than face rotation.

To remove the invalid face images from image sequences we apply a method of head pose estimation in Chapter 2. Based on the head pose, the invalid face

images over ±10° from axis of view of the camera are removed. Usually, in surveillance video, the resolution of the extracted face image is low so that the filtered face image within ±10° from the axis of view of the camera can be used to generate a HR face image which is suitable for face recognition.

To improve the resolution of the face images we have deployed Super Resolution (SR) methods. Basically, SR can be divided into single and multiple-frame based methods. Given the Biometric tunnel environment, we shall focus on the latter because the face image sequences are taken from a video. The SR process consists of three parts: registration, interpolation, and deblurring. In registration, the scene motion should be estimated for each image with reference to one particular image to align following images. Since the shifts in Low Resolution (LR) image frames are unknown, interpolation is necessary to match up High Resolution (HR) grid. Lastly, the reconstructed HR image usually could be blurred so that a deblurring process is needed. In many surveillance images the face region occupies less than one tenth of the whole image so that the SR method is necessary to clarify the face pattern and texture for human interpretation. The SR method can be used to magnify the input image so that pre-processing algorithms, such as the noise filtering and the background extraction which need enough resolution to obtain the accurate data, can work properly. Also, SR can be deployed to extract more face features from the resized face image and to reconstruct an accurate face image from the synthesised frontal face images which are generated by the view change in the 3D face models.

Considering all possibilities, in this Chapter we will describe the three-step pre-processing extraction methods leading to face recognition analysis.

6.2 Pose-based Face Image Filtering

There are many factors which must be considered in face extraction such as the resolution of the face image, facial expression, illumination, or facial obstacles. In this research, we assume that there are no illumination changes, no facial obstacles, and no facial expressions. We are only concerned with the relationship between head pose, image resolution, and face recognition. Ideally, a best sample for face recognition is a face image in which the face is looking directly towards the camera and the distance is the closest among all the images. In other words, the three axis angles (θ_{3Dx}, θ_{3Dy}, θ_{3Dz}) are near zero and translation (t_{3DZ}) in the z axis direction is the smallest in equation (2.16).

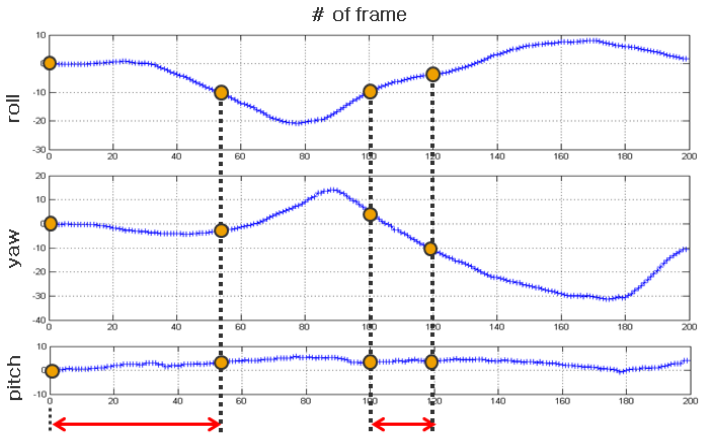


Figure 6.1: Pose-based image selection

Figure 6.2: Examples of the reconstructed frontal face images

Figure 6.1 shows sample motion data from the Boston face database. In this sample, the roll and yaw directions show a great deal of variation, but the pitch direction only shows small variation. Here, we choose the frame numbers in which the rotation along the three axes is within ±10° as being suitable for registration, choosing samples from 1 to 56 frames and 103 to 121 frames. Distance is not considered since its variation according to frame number is minimal. In this way, we can remove the face images which are invalid for face recognition.

Next, the SR algorithms are applied to enhance the quality of the image after pose-based filtering. The LR image sequences in the visual surveillance environments could be used to reconstruct the HR image. However, there could be a movement of the face and/or changes in illumination. Therefore, a robust SR method is necessary.

In Chapter 2, we describe a way of calculating the motion vector using the 3D ellipsoidal model. Here, we extract and reconstruct the frontal face image which was projected into the 2D image plane from the fitted 3D ellipsoidal model. Basically, if there is a large change in head pose, the reconstructed face image might become greatly distorted since our model is not based on an actual face model such as 3DMM. Thus, we only consider the face images with pose within ±10°, consistent with walking subjects, and where the frontal face can be reconstructed. Figure 6.2 shows the results of the reconstructed frontal face images for images which have pose variation within ±10° from the first frame. The total number of images is around 40 images per subject. As seen in figure 6.2, the sharpness and pattern of the images become clear from the left to the right images since they are taken when a person walks towards a camera. Among the reconstructed face images, the first 10 frames are used as an input to a SR algorithm.

6.3 High Resolution Image Reconstruction

SR is the process of combining a sequence of LR noisy blurred images to produce a HR image or sequence. The common notations are used to formulate the SR problem in the pixel domain [39] shown as follows. Basically, this is an inverse problem where HR image is generated from multiple LR frames.

$$Y_k = D_k H_k^{cam} F_k X_k + V \qquad k = 1, ..., N \tag{6.1}$$

$$\hat{X} = ArgMin\left[\sum_{k=1}^{n} \left\| D_k H_k F_k X - Y_k \right\|_p^p\right] \qquad 1 \le p \le 2 \tag{6.2}$$

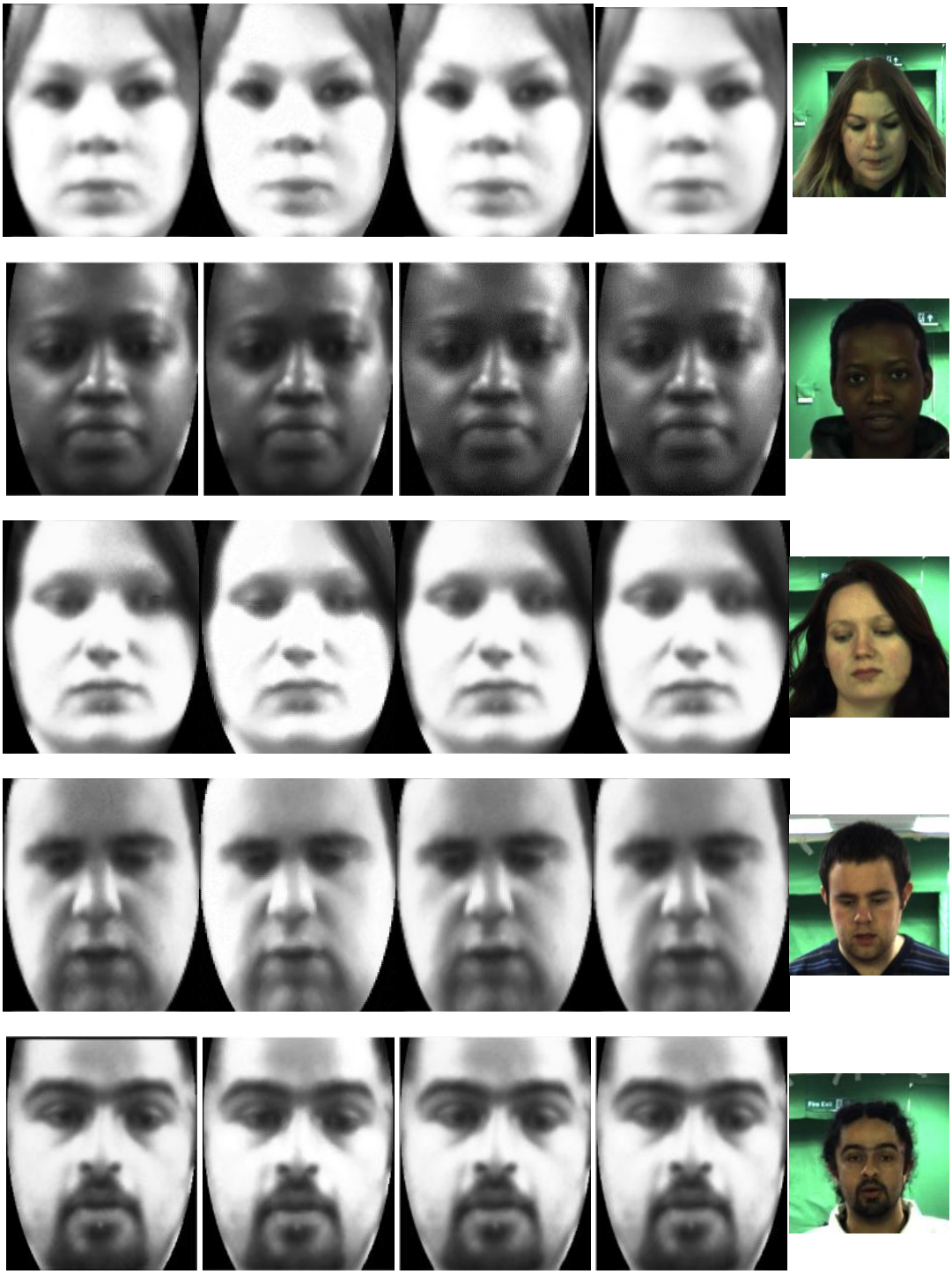
where F_k is the geometric motion operator between the HR frame X and k^{th} LR frame Y_k which are rearranged in lexicographic order. The camera's point spread function (PSF) is modelled by the blur matrix H_k^{cam}, and D_k represents the decimation operator. V_k is the system noise and N is the number of available LR frames.

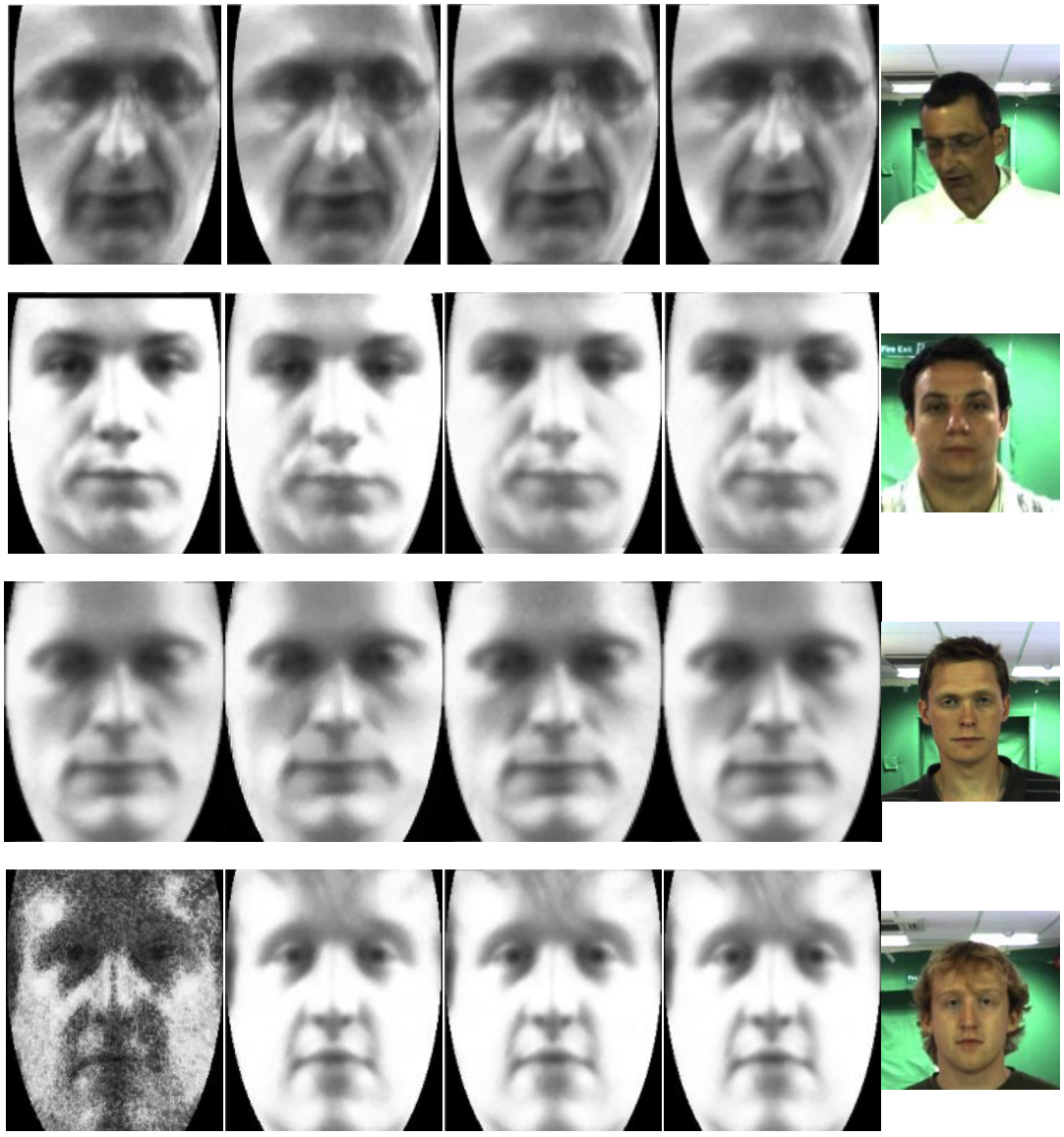
Figure 6.3: HR image quality comparison from normal images (1st row) and pose corrected images (2nd row) using IterNorm1 method [37]

To ensure that our method is reasonable we apply the SR method [37] to the approximate face region from Chapter 3 (figure 3.7) and the pose-corrected face images from the previous section (figure 6.2). As seen in figure 6.3, there are distinct differences between the results. Unlike the second row images, the upper images are blurred and the texture is not clear. In addition, the background is not distinguishable. The amount of blur and clarity depends on the variation of the head pose between frames. The left sample in the figure does not have much head movement in frames. Thus, the HR image is of similar quality to the original image. On the other hand, the middle subject has slight head movement in the vertical direction; the subject on the right has much head movement in the roll and pitch directions. Despite this, the SR reconstructions appear reasonably accurate. This result indicates that the reconstructed image quality is similar when there is not significant head movement. However, other cases show that the HR image which comes from the pose-corrected images shows less blurring and more accurate face pattern. Normally, there is a variation of head movement when the person walks so a method which can handle this variation is highly desirable.

To synthesise a HR image from the LR images, we applied four of the existing SR methods to the LR images: Bilateral Shift and Add (BSA), Iterative Norm 1 (IterNorm1), Median Shift and Add (MSA), Shift and Add (SA) [37-39].

The SA method involves calculation of the differential shifts of the images. The images are then shifted back to a common centre and added together. This provides an image with higher resolution. Here, we deploy SA method from [38]. The BSA is similar to the SA, with an additional pre-processing outlier detection/ removal algorithm from [39]. The MSA is conducting the SA using the Median operator from [39]. The IterNorm1 method is using L_1 norm minimization criterion to solve equation 6.2 with regularization factor and iteration from [36].





(a) BSA (b) IterNorm1 (c) MSA (d) SA (e) Last frame

Figure 6.4: Results of the HR images and the last frame of the image sequences; HR images which are generated by (a) Bilateral Shift and Add method, (b) Iteration Norm 1 method, (c) Median Shift and Add method, and (d) Shift and Add method (e) Last frame of the image sequence

The size of the LR images is approximately from 70 × 90 to 100 × 130 pixels (the distance between eye centres is around 30 to 40 pixels). They contain changes in illumination and pose variations. To ease alignment all LR images are resized to be 100 ×130 pixels. The resulting HR images are two times larger in each direction than the original LR images.

Figure 6.4 shows the synthesised HR images for each method and the approximate face region in the last frame (figure 6.4(e)). Since these HR images are made by the pose-corrected face images, there is little difference between the synthesised HR images, except in the bottom left sample is generated improperly due to disarrangement of the LR images. Figure 6.4(e) is the highest resolution image acquired from the sequence as the subject is closest to the camera, and it was not used to derive the HR images. The last frame is shown for comparison between HR images and a real image. As shown in figure 6.4(e), some of subjects look forward towards the camera, while others look to the right, or downwards, and some of the subjects even close their eyes. These images in which the subjects do not face the camera cannot be used for recognition since depending on head pose the recognition rate can vary [76]. However, the pose-corrected HR image can be used for recognition. As seen in this result, the synthesised HR image has a very similar pattern to that in the last frame.

6.4 Experimental Results

Basically, it is not an easy task to evaluate the quality of the HR image since it is much bigger than a LR image. Also, SR methods use de-convolution to remove image noise. Thus, texture and image illumination could be changed and it is unknown whether this has an adverse effect or not on recognition. Also, there is no ground truth to compare this with. In addition, the normalisation method can be different so that it affects the quality of image. Accordingly, in this study,

indirect measures are used. We evaluate the possibility of comparing the HR faces to the LR ones for automatic face recognition.

To evaluate recognition performance, we analysed one sequence of around 40 images for each of 34 subjects. The first test aims to explore whether the HR image can represent the characteristics of LR images. Accordingly, we then evaluate the face recognition capability between each HR image which is generated by different SR algorithms (BSA, IterNorm1, MSA, and SA method) and the 10 LR image frames used to construct each HR image. So the gallery set consists of 136 HR images and probe set consists of 340 LR images for 34 subjects (each subject has four HR images and 10 LR images). We test four methods such as Principal Component Analysis (PCA), Speeded Up Robust Features (SURF) [30], Scale Invariant Feature Transform (SIFT) [40], and Cross Correlation. Figure 6.5 shows the recognition results. Note that several methods can classify all faces correctly; the SIFT was generally of the lowest performance and Cross Correlation was of the highest performance. All the correct classification methods represent the characteristics of LR images sufficiently for good recognition capability. The average rates in BSA, IterNorm1, MSA, and SA are 97.1%, 97.1%, 97.8%, and 97.8%, respectively. Thus, this confirms that there is little difference between them as observed previously.

The second test aims to explore whether the HR image can capture the face structure in images which were not used in HR reconstruction. In this case, we use PCA-based recognition only as a test basis since it explores the recognition capability. Other approaches could be used or the PCA method could be refined, but our purpose is to demonstrate whether there is an advantage in this new HR from gait method. PCA-based recognition was also one of the best performing algorithms in the previous image comparison. We compare the HR image with the rest of the LR images from the sequence which are not used to generate the HR

images. The gallery set consists of 502 pose-corrected images extracted from the same video and the probe set contains 136 HR images. Figure 6.6 shows the resulting recognition rate between the HR images and the LR images. Of the super resolution techniques, generation by SA shows the highest recognition rate (88.2%) and generation by MSA shows the lowest recognition rate (79.4%). The average recognition rate is 84.6%. This indicates that the HR images can be successfully generated from the LR images with fidelity that is sufficient for recognition purposes.

In order to confirm that using our approach is more effective for face recognition than using the LR images, we test the recognition rate between 340 LR images which are used to generate the HR images and the remaining 502 LR images which are not used to generate the HR images. The column furthest on the right in figure 6.6 shows the resulting recognition rate. The resulting recognition is 58.8% which is around 30% lower than that for HR images, confirming the advantages of this new approach.

The average recognition rate is over 84% for the HR images. Considering constraints such as large changes in illumination and low resolution, the recognition rate is high. In addition, there are no directly comparable results generated by other techniques since this is the first approach to apply SR techniques using a 3D face model and gait from image sequences in a surveillance environment.

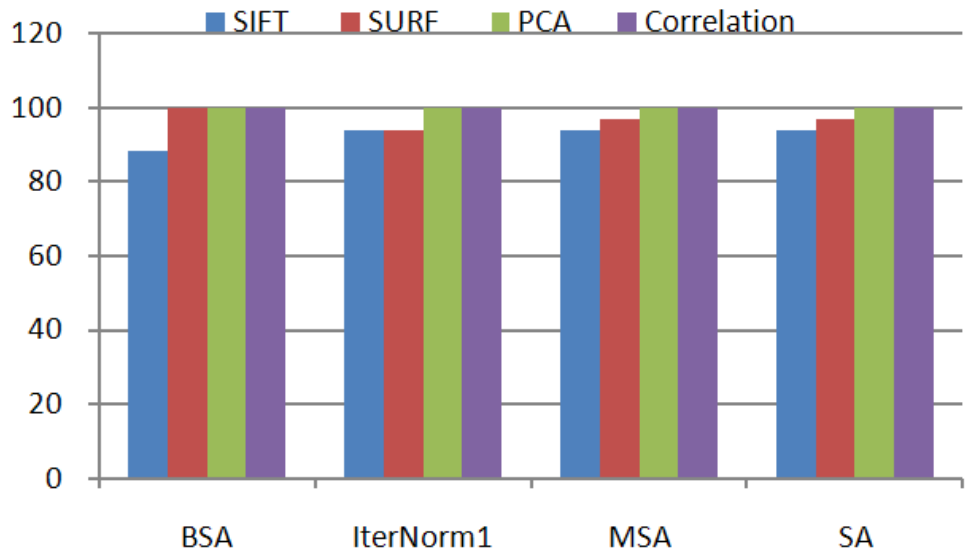


Figure 6.5: Recognition rate between four HR images made by different SR methods and the LR images used to build the HR images.

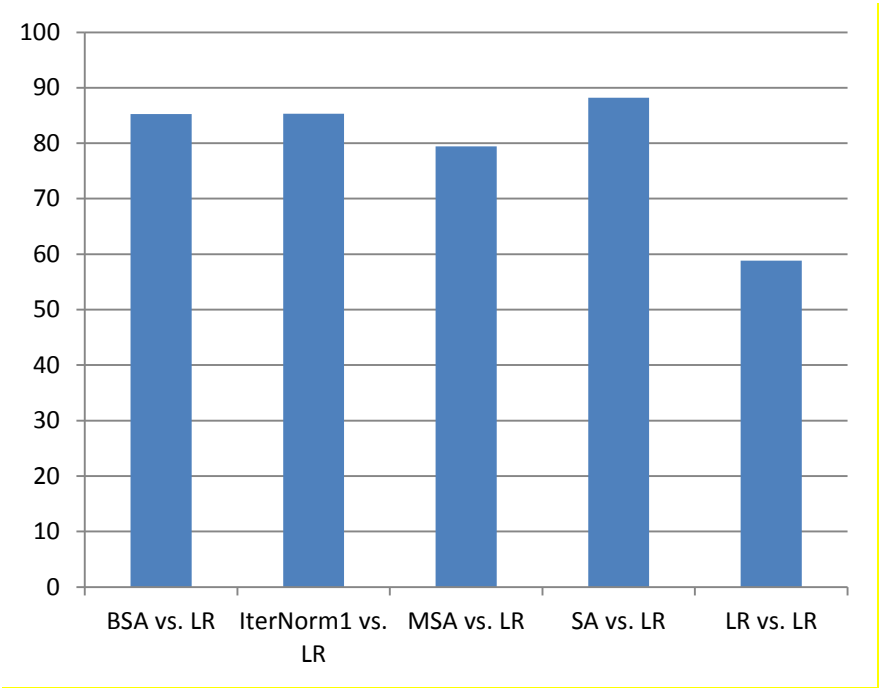


Figure 6.6: Recognition rate between four HR images made by different SR methods, and the rest of LR images which were not used to build the HR images, using PCA based recognition

6.5 Conclusions

In this chapter, we showed a way of extracting the frontal face from image sequences. Based on head position information we filtered the invalid face images. Then, using the pose-corrected LR images, an HR face image was generated. We adapted the existing four SR algorithms to synthesise the HR face image. In addition, the quality of HR images was evaluated by using the indirect approach; PCA-based recognition. The result indicates that the generated HR image is much more suitable for face recognition than the LR image. However, in this method, we only consider face images within ±10° from the camera view since our method is based on an approximate model (3D ellipsoidal model).

Chapter 7

Overall Conclusions

In this thesis we have shown how head movement can be modeled by gait motion to allow for accurate face extraction when a subject walks towards a camera. This thesis describes new techniques for head pose estimation and for gait trajectory analysis. The new approach to head pose estimation combines an ellipsoidal model with SIFT-based low-level feature extraction. In addition, we modeled the gait trajectory in 2D and 3D space to extract an approximate face region including analysis of a looming effect. The usefulness of the proposed methods was demonstrated using a variety of databases. For the controlled environments the Biometric tunnel database [21] and the Boston face dataset [24] were used. For the uncontrolled environments, we used the CAVIAR database [45] and PETS 2006 data [46].

Three basic modules were developed and tested: 3D head pose estimation, gait trajectory modeling, and frontal face extraction. First of all, the 3D head pose estimation method was tested using the Boston face database and a 3D ellipsoidal model applied to fit a face. We formulated the objective function between actual 3D SIFT points and 2D SIFT points based on the 3D ellipsoidal model. Then, Levenberg-Marquardt optimisation was applied for obtaining head motion in 3D

space. The test result showed that average errors in roll, yaw, and pitch direction are less than 4°. The next step was to undertake gait trajectory modeling. The gait trajectory model was constructed, and proved the effectiveness of tracking a walking object's movement by 2D and 3D gait trajectory analysis. The fit of the model with actual data was verified by Levenberg-Marquardt optimisation. This showed that the fitting result was over 98%. Also, invalid face images are filtered based on the position of a head. Then, to improve the resolution of LR images, SR methods of monochrome image sequences were applied to reconstruct a HR frontal face images. PCA-based recognition was used as a measurement to evaluate the quality of the HR images. By the new method the overall recognition performance is over 84% whereas it was around 54% for the images from which the super-resolution images were derived.

Our main contribution in this research has been to present a new method of extracting the frontal face image within visual surveillance environment. This thesis evolved from a simple observation that when a person walks there is sinusoidal movement in the vertical direction. Analysing this observation we can extract the face region, detect the heel strike position, and reconstruct the 3D region of the face. Based on that, we can generate the frontal view face image by adding 3D pose estimation and adapting SR method. This is the first use of gait information to enhance the face extraction process.

In the future we aim to translate these approaches to analyse surveillance data in order to provide a high resolution face image from a surveillance video's data in which a subject's movement is unconstrained. To achieve this there are many modules to develop a more generalised automatic system for analysing video scenes.

For pre-processing, a silhouette extraction method which is robust to changes in illumination needs to be built in order to apply our approach to outdoor

environments. Also, multiple object tracking should be one of modules in the system. In the thesis, we assume that there is only one subject so that a Homography relationship based on local feature matching can be calculated. However, to generalise our system this factor should be considered.

For head motion estimation, a more accurate face model needs to be built. In this thesis, we use a 3D ellipsoidal model to estimate the head position in 3D space. Even though it can track the head motion precisely, the frontal face image cannot be derived directly from a fitting model since the model is an approximate one. If the head motion is significant, a synthesised frontal face image could be distorted. Therefore, an actual face model such as 3DMM or 3D AAM should be considered. Moreover, automatic initialisation is another factor that needs to be considered when building an automatic system.

For gait trajectory extraction, 3D data of the human can be used to enhance modeling of gait motion. In this research, all data are 2D-based data even though the analysis was conducted by 3D models such as the 3D ellipsoidal face model and the 3D gait trajectory model. We seek to obtain 3D information from 2D planar imaginary. Therefore, the initialisation and fitting process is very important in order to avoid the potential tracking error. Given 3D data of human motion, such as voxel data, the 3D gait trajectory can be calculated more accurately.

For frontal face extraction, another goal is to develop a method which can evaluate the illumination quality from an image sequence. During SR processing, it shows that the LR images, which are highly affected by changes in illumination, decrease the recognition rate. Using this method, the invalid LR images can be removed before the SR process, which could help to improve recognition performance. Further, a more accurate 2D frontal face image needs to be generated. From each valid LR image which is selected by the image quality module, a 3D actual model can be applied to each LR image. Then, we can analyse each actual 3D face model

and synthesise the more standard actual 3D face model for each person which does not have an illumination effect. Accordingly, more accurate 2D face models can be generated from the projection of the actual 3D face models which are suitable for face recognition purposes.

Further, extending these methods, a full model of the human motion can be modeled under visual surveillance environments. Based on new methods introduced in this thesis and in suggestions for future work, the surveillance video can be analysed in unconstrained environments. This analysis can be applied to human computer interaction, surveillance monitoring, access control system and gesture recognition system.

In summary, future work should concentrate both on improvement of the automation of the system and development of the image quality with the standard actual 3D model under the visual surveillance environment. Also, we seek to extend this technique to, and evaluate its potential in, practical applications such as forensics research.

References

[1] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe, "FRVT 2006 and ICE 2006 large-scale results," *NIST IR 7408*, March 2007.

[2] G. Medioni, J. Choi, C. H. Kuo, and D. Fidaleo, "Identifying noncooperative subjects at a distance using face images and inferred three-dimensional face models", *IEEE Trans. Syst. Man, Cybern. A, syst. and humans*, **39**(1), pp. 12-24, Jan. 2009.

[3] F. W. Wheeler, X. Liu, and P. H. Tu, "Multi-frame super-resolution for face recognition," in *Proc. IEEE Conf. Biometrics, Theory, Appl., Syst*., 2007, pp 1-6.

[4] P. Mortazavian, J. Kittler, and W. Christmas, "A 3-D assisted generative model for facial texture super-resolution", in *Proc. IEEE Conf. Biometrics, Theory, Appl., Syst*., 2009, pp. 452-458.

[5] K. Jia and S. Gong, "Generalized face super-resolution", IEEE Trans. Image process., **17**(6), pp. 873-886, June. 2008.

[6] P. H. Hennings-Yeomans, S. Baker, and B. V. K. V. Kumar, "Simultaneous super-resolution and feature extraction for recognition of low-resolution faces", in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1-8.

[7] W. W. W. Zou and P. C. Yuen, "Learning the relationship between high and low resolution images in kernel space for face super resolution", in *Proc. Int. Conf. Pattern Recog.*, 2010, pp. 1152-1155.

[8] B. Li, H. Chang, S. Shan, and X. Chen, "Low-resolution face recognition via coupled locality preserving mappings", *IEEE Signal Process. Lett.,* **17**(1), pp. 20-23, Jan. 2010.

[9] K. W. Bowyer, K. Chang, and P. Flynn, "A survey of approaches and

challenges in 3D and multi-modal 3D+2D face recognition," *Comput. Vis. Image Underst*., **101**(1), pp. 1-15, Jan. 2006.

[10] J. Kittler, A. Hilton, M. Hamouz, and J. Illingworth, "3D assisted face recognition: a survey of 3D imaging, modeling and recognition approaches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog*., 2005, pp. 1-11.

[11] U. Park and A. K. Jain, "3D model-based face recognition in video," in *Proc. IEEE/IAPR Int. Conf. Biometrics*, 2007, pp.1085-1094.

[12] S. V. Duhn, M. J. Ko, L. Yin, T. Hung, and X. Wei, "Three-view surveillance video based face modeling for recognition," in *Proc. Biometric Symposium*, pp. 1-6, 2007.

[13] X. Lu, A. K. Jain, and D. Colbry, "Matching 2.5D face scans to 3D models," *IEEE Trans. Pattern Anal. Mach. Intell*., **28**(1), pp. 31-43, Jan. 2006.

[14] V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," *IEEE Trans. Pattern Anal. Mach. Intell*., **25**(9), pp. 1063-1074, Sept. 2003.

[15] C. T. Liao, S. F. Wang, Y. J. Lu, and S. H. Lai, "Video-based face recognition based on view synthesis from 3D face model reconstructed from a single image," in *Proc. IEEE Int. Conf. Multimedia, Expo*, 2008. pp 1589-1592.

[16] R. Stiefelhagen, K. Bernardin, H. K. Ekenel, and M. Voit, "Tracking identities and attention in smart environments – contributions and progress in the CHIL project," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog*., 2008, pp. 1-8.

[17] A. Kale, A. K. Roychowdhury, and R. Chellappa, "Fusion of gait and face for human identification," in *Proc. IEEE Int. Conf. Acoustics Speech, Signal Process*., 2004, pp. 901-904.

[18] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer, "The humanID gait challenge problem: data sets, performance and analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, **27**(2), pp. 162-177, Feb. 2005.

[19] G. Shakhnarovich, L. Lee, and T. Darrell, "Integrated face and gait recognition from multiple views," in *Proc. IEEE Conf. Comput. Vis.*

Pattern Recog., 2001, pp. 439-446.

[20] X. Zhou and B. Bhanu, "Integrating face and gait for human recognition at a distance in video," *IEEE Trans. Syst., Man, Cybern. B, Cybern*., **37**(5), pp. 1119-1137, Oct. 2007.

[21] R. D. Seely, S. Samangooei, L. Middleton, J. N. Carter, and M. S. Nixon, "The university of Southampton multi-biometric tunnel and introducing a novel 3D gait dataset," in *Proc. IEEE Conf. Biometrics, Theory, Appl., Syst*.,2008, pp. 1-6.

[22] I. Matthews and S. Baker, "Active appearance models revisited," *Int. J. Comput. Vis.*, **60**(2), pp.135-164, Feb. 2004.

[23] G. D. Hager and P. N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Trans. Pattern Anal. Mach. Intell*., **20**(10), pp. 1025-1039, Oct. 1998.

[24] M. La Cascia, S. Sclaroff, and V. Athitsos, "Fast, reliable head tracking under varying illumination: An approach based on robust registration of texture-mapped 3D models*," IEEE Trans. Pattern Anal. Mach. Intell*., **22**(4), pp. 322-336, Apr. 2000.

[25] S. Basu, I. Essa, and A. Pentland, "Motion regularization for model-based head tracking," in *Proc. Int. Conf. Pattern Recog*., 1996, pp. 611-616.

[26] J. Xiao, T. Kanade, and J. Cohn, "Robust full-motion recovery of head by dynamic templates and re-registration techniques," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog*., 2002, pp. 156-162.

[27] J. S. Jang and T. Kanade, "Robust 3D head tracking by online feature registration," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog*., 2008, pp. 1-6.

[28] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical introduction to robotic manipulation*, CRC Press, 1994.

[29] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision (2nd ed.)*, Cambridge University Press, 2004.

[30] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," *Comput. Vis. Image Underst*., **110**(3), pp. 346-359, Jun. 2008.

[31] T. K. M. Lee, M. Belkhatir, and S. Sanei, "Fronto-normal gait incorporating accurate practical looming compensation," in *Proc. Int. Conf. Pattern Recog*., 2008, pp 1-4.

[32] G. Cheung, T. Kanade, J. Bouquet, and M. Holler, "A real time system for robust 3d voxel reconstruction of human motions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2000, pp. 714-720.

[33] Aristotle, On the Motion of Animals, B.C. 350 (available at http://classics.mit.edu/Aristotle/motion_animals.html, 15/4/2004)

[34] M. S. Nixon and J. N. Carter, "Automatic recognition by gait," in *Proc. IEEE*, **94**(11), pp.2013-2024, Nov. 2006.

[35] J.M. Burnfield and C. M. Powers, *Normal and Pathologic Gait, in Orthopaedic Physical Therapy Secrets*, 2nd edition, Hanley & Belfus, 2006.

[36] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Process*., **13**(10), pp. 1327-1344, Oct. 2004.

[37] M. Elad and A. Feuer, "Restoration of a single super-resolution image from several blurred, noisy and down-sampled measured images," *IEEE Trans. Image Process*., **6**(12), pp. 1646-1658, Dec. 1997.

[38] M. Elad and Y. Hel-Or, "A fast super-resolution reconstruction algorithm for pure translational motion and common space invariant blur," *IEEE Trans. Image Process*., **10**(8), pp. 1187-1193, Aug. 2001.

[39] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Robust shift and add approach to super-resolution," in *Proc. Appli. Digital Signal, Image Process*., 2003, pp. 121-130.

[40] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis*., **60**(2), pp. 91-110, Jan. 2004.

[41] K. Liu, Y.P. Luo, G. Tei, and S.Y. Yang, "Attention Recognition of Drivers Based on Head Pose Estimation," in *Proc. IEEE Vehicle Pow. Propul. Conf.*, 2008, pp. 1-5.

[42] K.Mikolajczyk et al. "A comparison of affine region detectors," *IJCV*, **65**(1), pp. 43-72, 2005.

[43] K.Mikolajczyk and C.Schmid, A performance evaluation of local descriptors, *IEEE Trans. PAMI*, **27**(10), pp. 1615-1630, 2005.

[44] R. D. Seely, On a three-dimensional gait recognition system, *Ph.D. thesis*, 2010.

[45] *The CAVIAR Test Case Scenarios: http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/.*

[46] PETS 2006 Benchmark Data, *http://www.cvg.rdg.ac.uk/PETS2006/data.html*

[47] K. H. An and M. J. Chung, "3D head tracking and pose-robust 2D texture map-based face recognition using a simple ellipsoid model," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots, Syst.*, 2008, pp. 307-312.

[48] I. Bouchrika, M. S. Nixon, "Model-based feature extraction for gait analysis and recognition," in *Proc. Int'l Conf. on Computer Vision/ Computer Graphics Collaboration Techniques and Applications, LNCS 4418*, 2007, pp.150-160.

[49] A. F. Bobick, A. Y. Johnson, "Gait recognition using static, activity-specific parameters," in *Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, 2001, pp.423-430.

[50] C. BenAbdelkader, R. Cutler, L. Davis, "View-invariant estimation of height and stride for gait recognition," in *Proc. Int'l European Conf. on Computer Vision Workshop: Biometric Authentication*, 2002, pp. 155-167.

[51] F. Jean, R. Bergevin, A. B. Albu, "Body tracking in human walk from monocular video sequences," in *Proc. Canadian Conf. on Computer and Robot Vision*, 2005, pp. 144-151.

[52] J. Vignola, J. F. Lalonde, R. Bergevin, "Progressive human skeleton fitting," in *Proc. Int'l Conf. on Vision Interface*, 2003, pp. 35-42.

[53] Z. Zhou, A. Prugel-Bennett, R. I. Damper, "A Bayesian framework for extracting human gait using strong prior knowledge," *IEEE Trans. Pattern Anal. Mach. Intell*., **28**(11), pp. 1738-1752, Nov. 2006.

[54] L. Sigal, M. J. Black, "Measure locally, reason globally: occlusion-sensitive articulated pose estimation," in *Proc. IEEE Int'l Conf. on*

Computer Vision and Pattern Recognition , 2006, pp. 2041-2048.

[55] J. Zhang, J. Luo, R. Collins, Y. Liu, "Body localization in still images using hierarchical models and hybrid search," in *Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, 2006, pp. 1536-1543.

[56] A. Sundaresan, R. Chellappa, "Model-driven segmentation of articulating humans in laplacian eigenspace," *IEEE Trans. Pattern Anal.Mach. Intell.*, **30**(10), pp1771-1785, Oct. 2008.

[57] P. Viola and M. Jones, "Robust real-time object detection", *Int'l Journal of Computer Vision*, **57**(2), pp. 137-154, Feb. 2001.

[58] R. Mohedano, C. R. del-Blanco, F. Jaurequizar, L. Salgado, and N. Garcia, "Robust 3D people tracking and positioning system in a semi-overlapped multi-camera environment", In *Proc. Int'l Conf. on Image Processing*, 2008, pp. 2656-2659.

[59] L. Li and M. K. H. Leung, "Unsupervised learning of human perspective context using ME-DT for efficient human detection in surveillance", in *Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, 2008, pp. 1-8.

[60] J. Saboune and R. Laganiere, "People detection and tracking using the explorative particle filtering", In *Proc. Int'l Conf. on Computer Vision Workshop*, 2009, pp. 1298-1305.

[61] F. Jean, A. B. Albu, and R. Bergevin, "Towards view-invariant gait modeling: computing view-normalized body part trajectories", *Pattern Recog.*, **42**(11), pp. 2936-2949, Nov. 2009.

[62] M. Li, Z. Zhang, K. Huang, and T. Tan, "Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection", In *Proc. Int'l Conf. on Pattern Recognition*, 2008, pp. 1-4.

[63] M. Yang, F. Lv, W. Xu, K. Yu, and Y. Gong, "Human action detection by boosting efficient motion features", In *Proc. Int'l Conf. on Computer Vision Workshop*, 2009, pp. 522-529.

[64] A. Leykin and R. Hammoud, "Real-time estimation of human attention field in LWIR and color surveillance videos", In *Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition Workshop*, 2008, pp. 1-6.

[65] Y. T. Chen and C. S. Chen, "Fast human detection using a novel boosted cascading structure with meta stages", *IEEE Trans. Image Processing*, **17**(8), pp. 1452-1464, 2008.

[66] Q. Ladetto, V. Gabaglio, B. Merminod, P. Terrier, and Y. Schutz, "Human Walking Analysis Assisted by DGPS", *Proc. Global Navigation Satellite System,*2000, pp. 1-6.

[67] Y. Meyer, *Wavelets and operators, Cambridge Studies in Advanced Mathematics*, Cambridge University Press, 1992.

[68] S. U. Jung and M. S. Nixon, "On using gait biometrics to enhance face pose estimation", In *Proc. IEEE Conf. Biometrics, Theory, Appl., Syst*., 2010, pp. 1-6.

[69] C. Harris and M. Stephens, "A combined corner and edge detector," In *Proc. the Alvey Vision Conference,* 1988, pp. 147-151.

[70] K. Mikolajczyk and C. Schmid, Indexing based on scale Invariant interest points, *Proc. 8th ICCV*, pp. 525-531, 2001.

[71] T. Kadir and M. Brady, Scale, saliency and image description, *IJCV*, **45**(2), pp.83-105, 2001.

[72] F. Jurie and C. Schmid, Scale-invariant shape features for recognition of object categories, *Proc. IEEE CVPR*, Vol.2, pp. 90-96, 2004.

[73] The XM2VTS face database, http://www.ee.surrey.ac.uk/CVSSP-/xm2vtsdb/

[74] R. Keys, Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoustics, Speech, and Signal Processing,* **29**(6), pp. 1153-1160, 1981.

[75] The Sheffield face database, http://www.shef.ac.uk/eee/research/iel/-research/face.html

[76] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, Face recognition: A literature survey, *ACM Computer Surveys*, **35**(4), pp. 399-458, 2003.

[77] K. P. C. Edwin and H. Z. Stanislaw, *An introduction to optimization (2nd ed.)*, John wiley & Sons, Inc., 2001.

[78] R. C. Gonzalez and R. E. Woods, *Digital image processing (2nd ed.),*

Prentice Hall, 2002.

[79] OpenCV library: *http://opencv.willowgarage.com/wiki*/

Appendix A

More Experimental Results

A.1 Full Image Sequence Experiment

In the main body of this thesis, we presented sections of the results from full sequence experiments. In this Chapter, we will show the full sequence results from Chapters 2 to 5.

First three figures (figures A.1-A.3) are one of full image sequence results presented in Chapter 2 based on the Boston face database. The frame rate is 30 FPS. The middle three figures (figures A.4-A6) are the results from Chapter 3, which show the approximate face region extracted and 3D head-pose tracking by a 2D gait trajectory model. The frame rate of this database is 10 FPS. The last two figures (figures A.7 and A.8) are the results from Chapter 5. The results present the extracted approximate face region by 3D gait trajectory model. The frame rate is around 30 FPS.

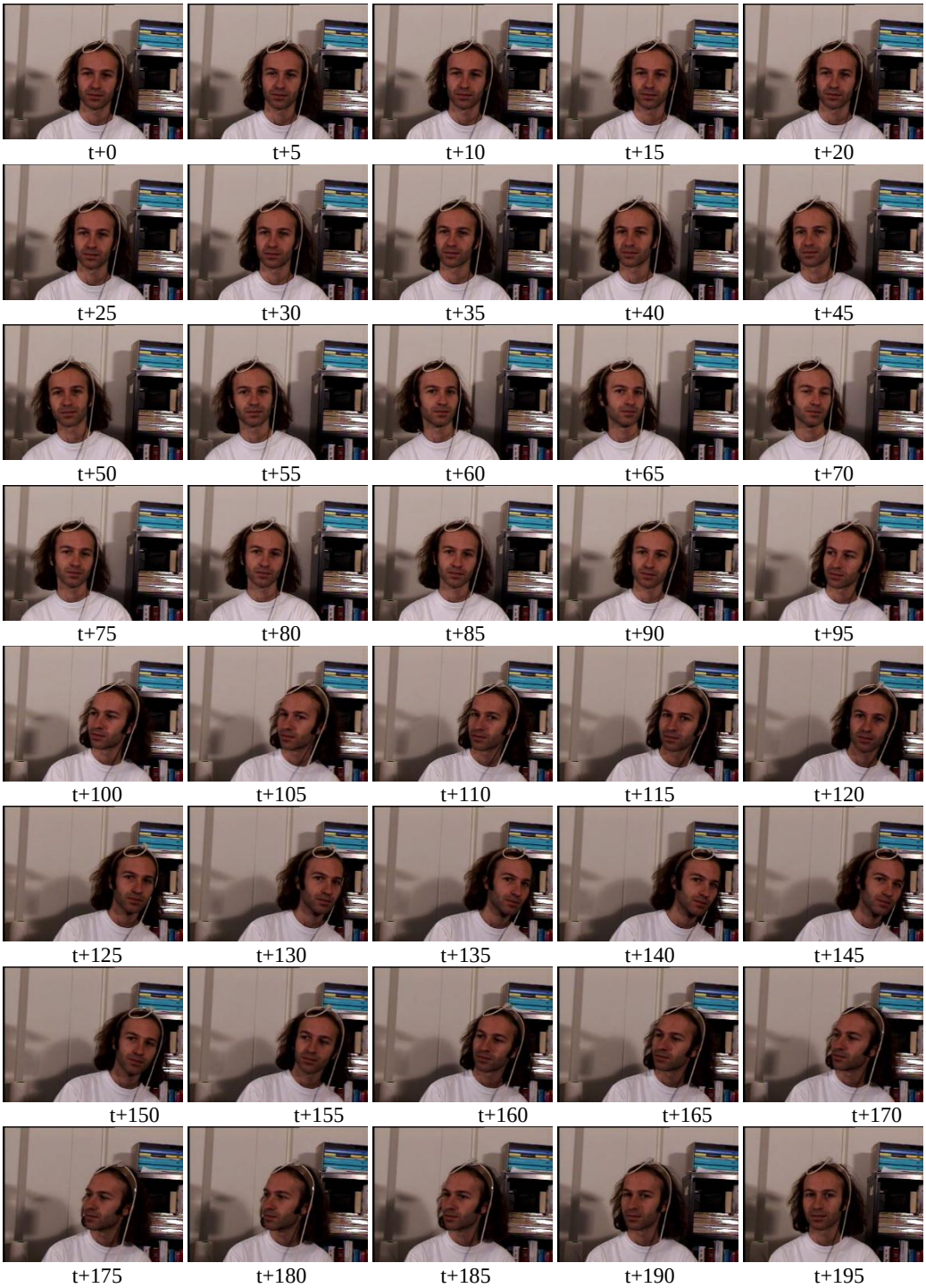


Figure A.1: An original image sequence from the Boston face dataset

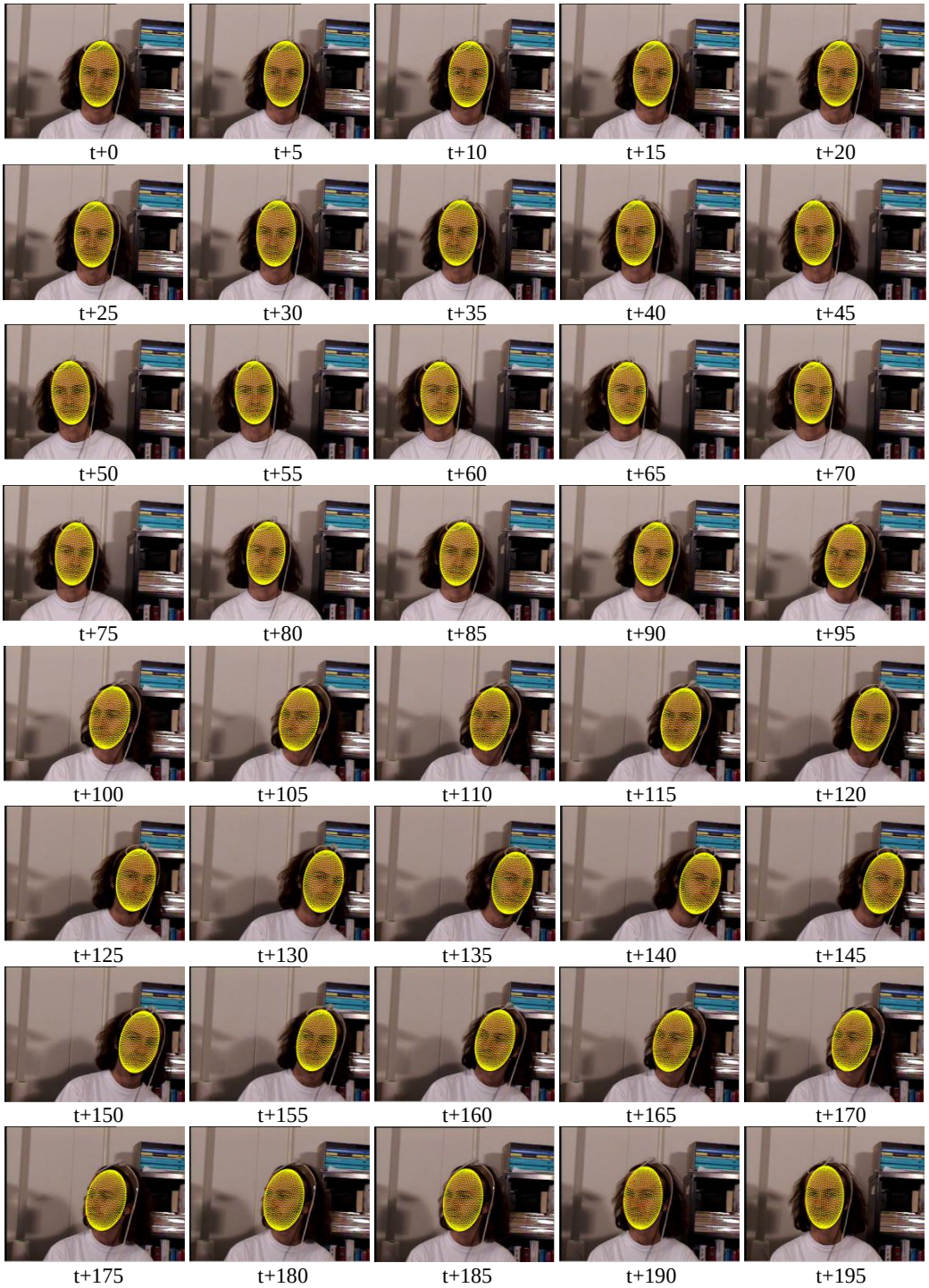


Figure A.2: Face tracking results in the Boston face dataset

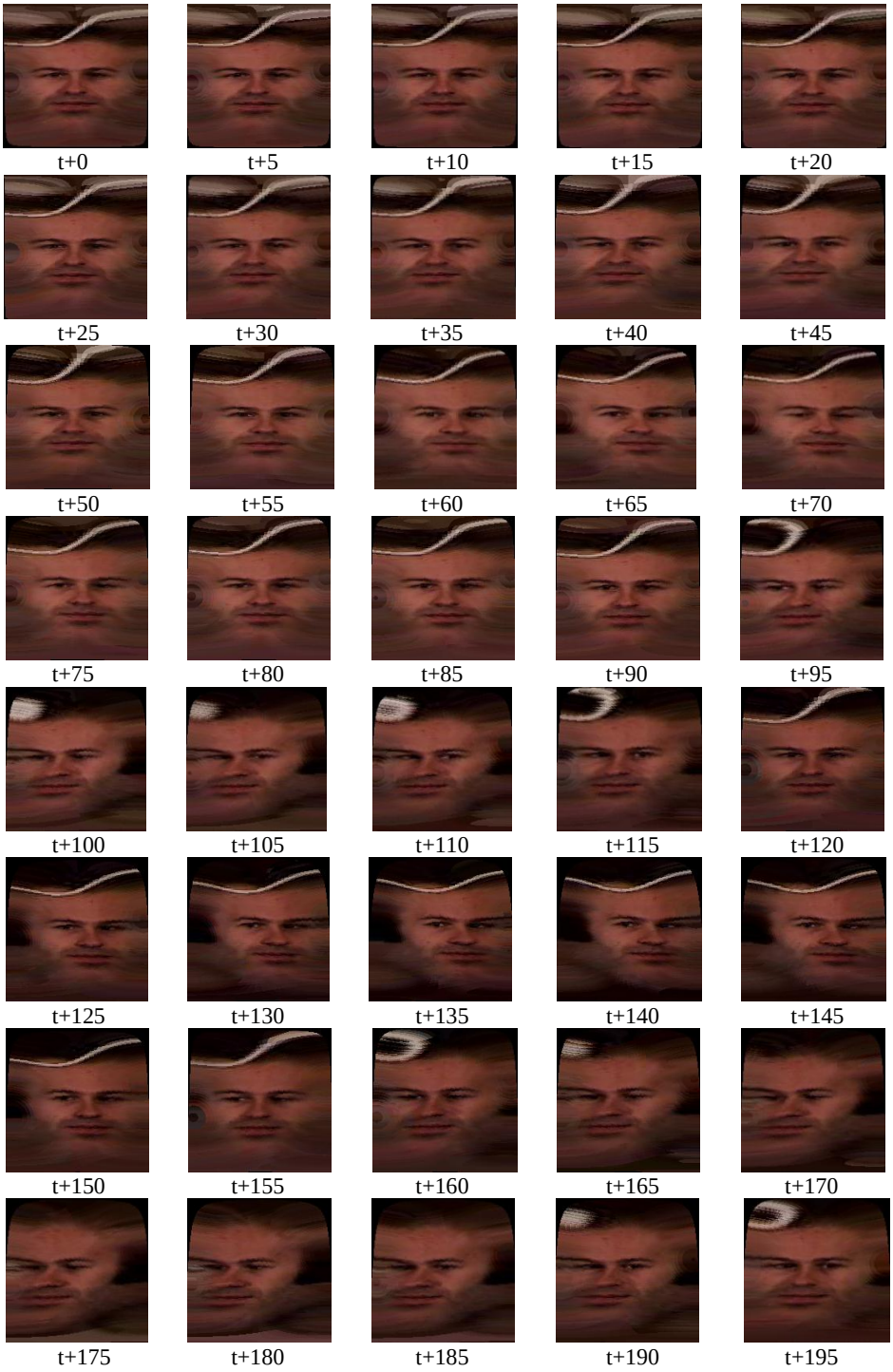


Figure A.3: Unfolded images for each frame

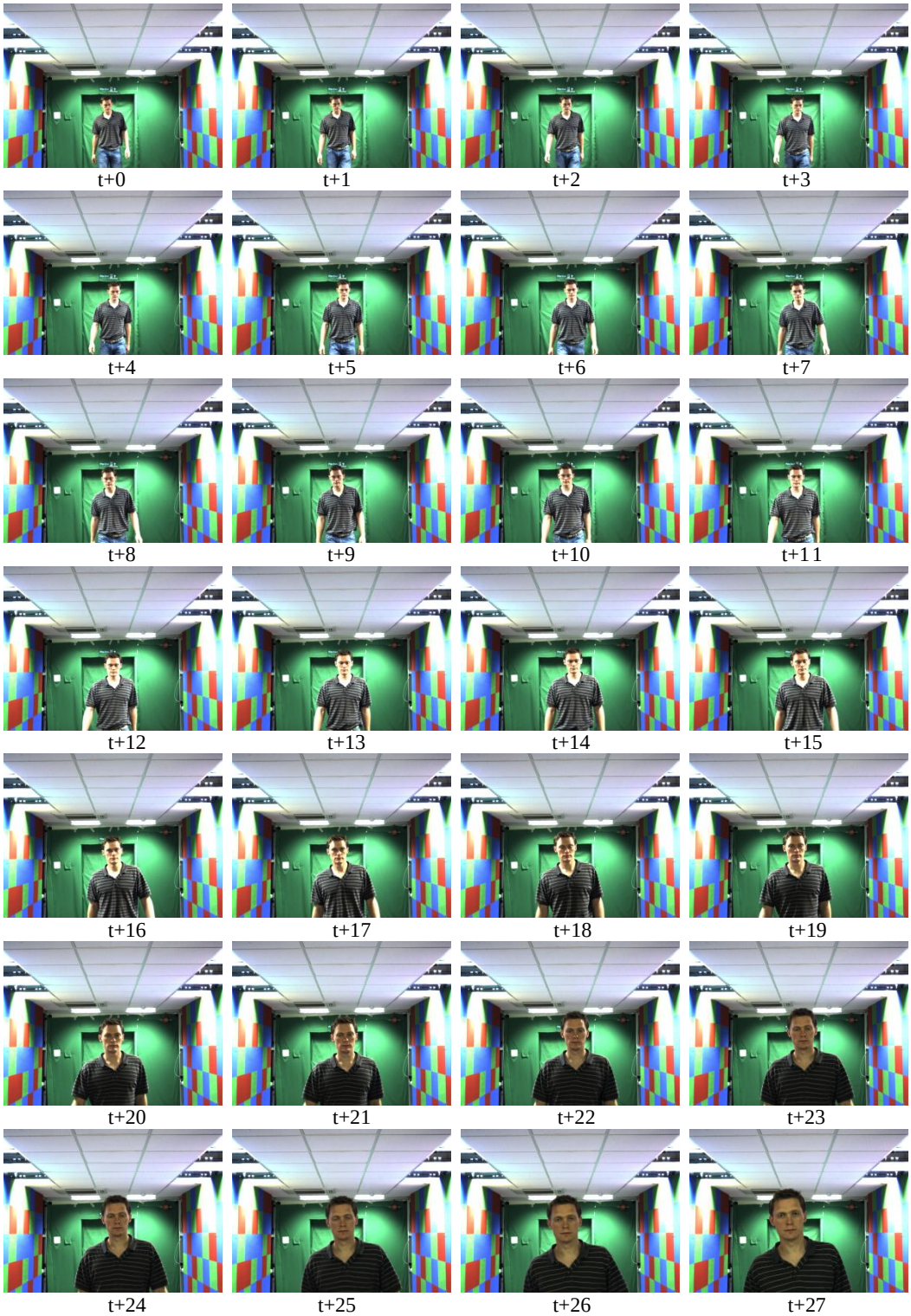


Figure A.4: An original image sequence from the Biometric tunnel database

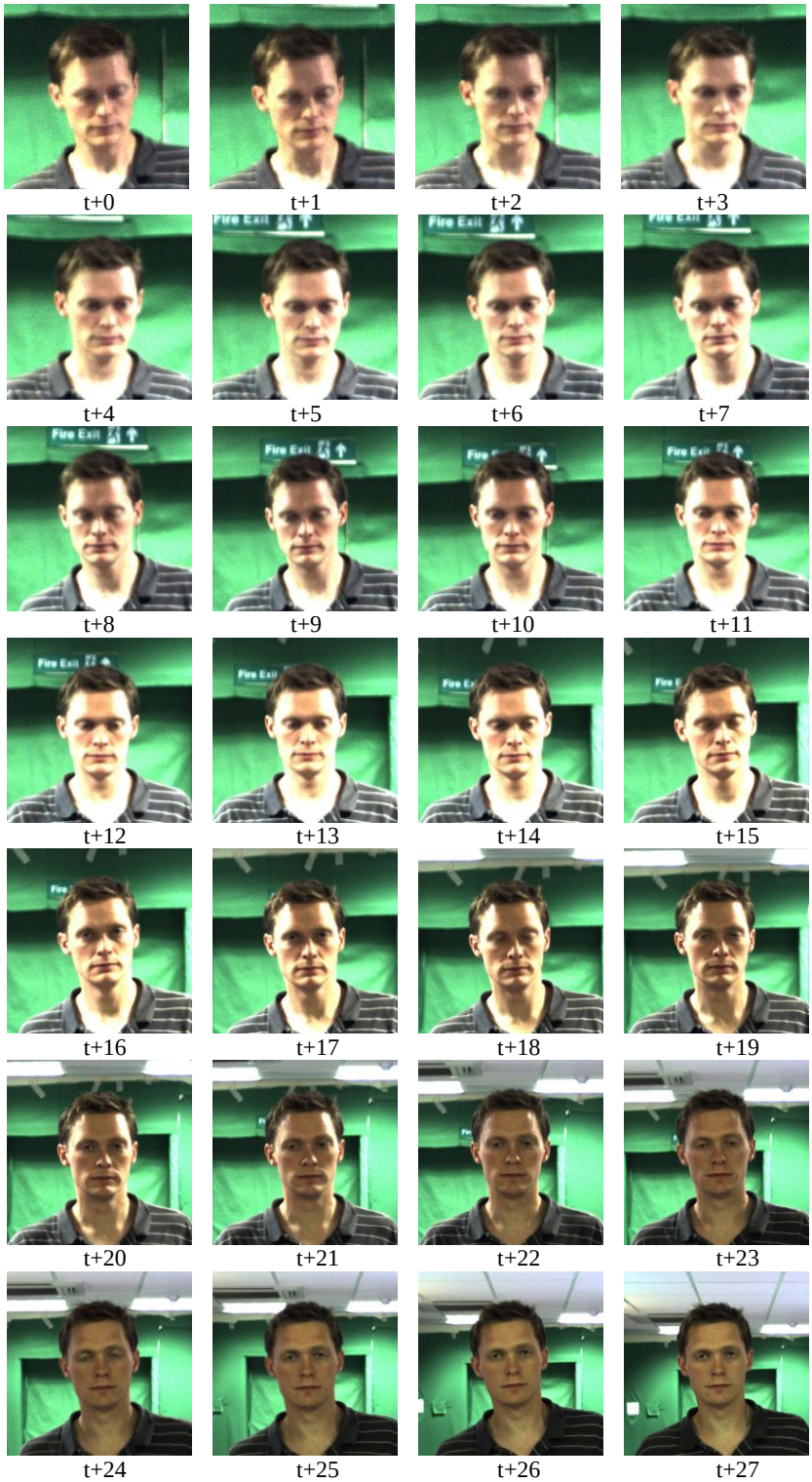


Figure A.5: Approximate face regions by a 2D gait trajectory

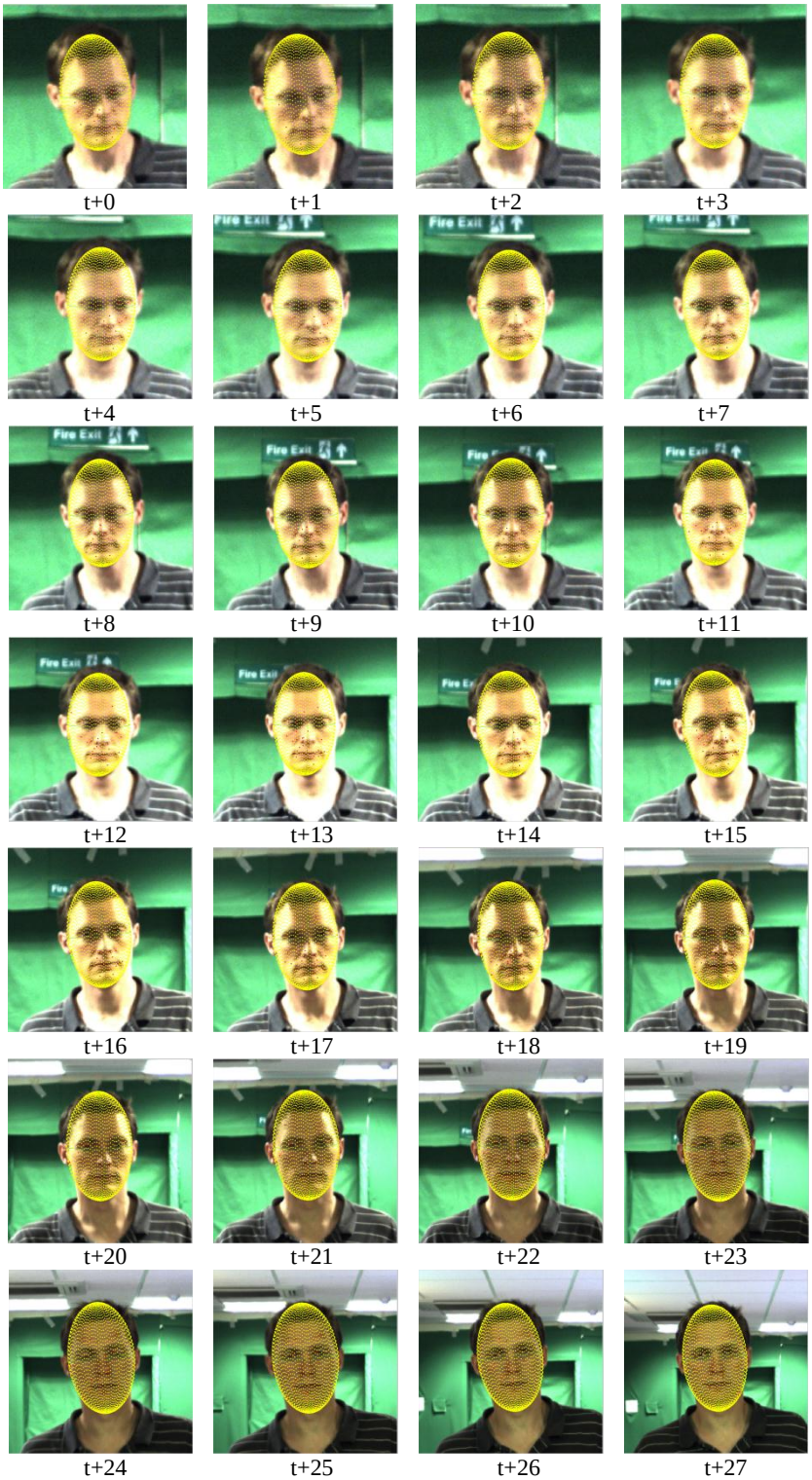


Figure A.6: Face tracking results in the Biometric tunnel dataset

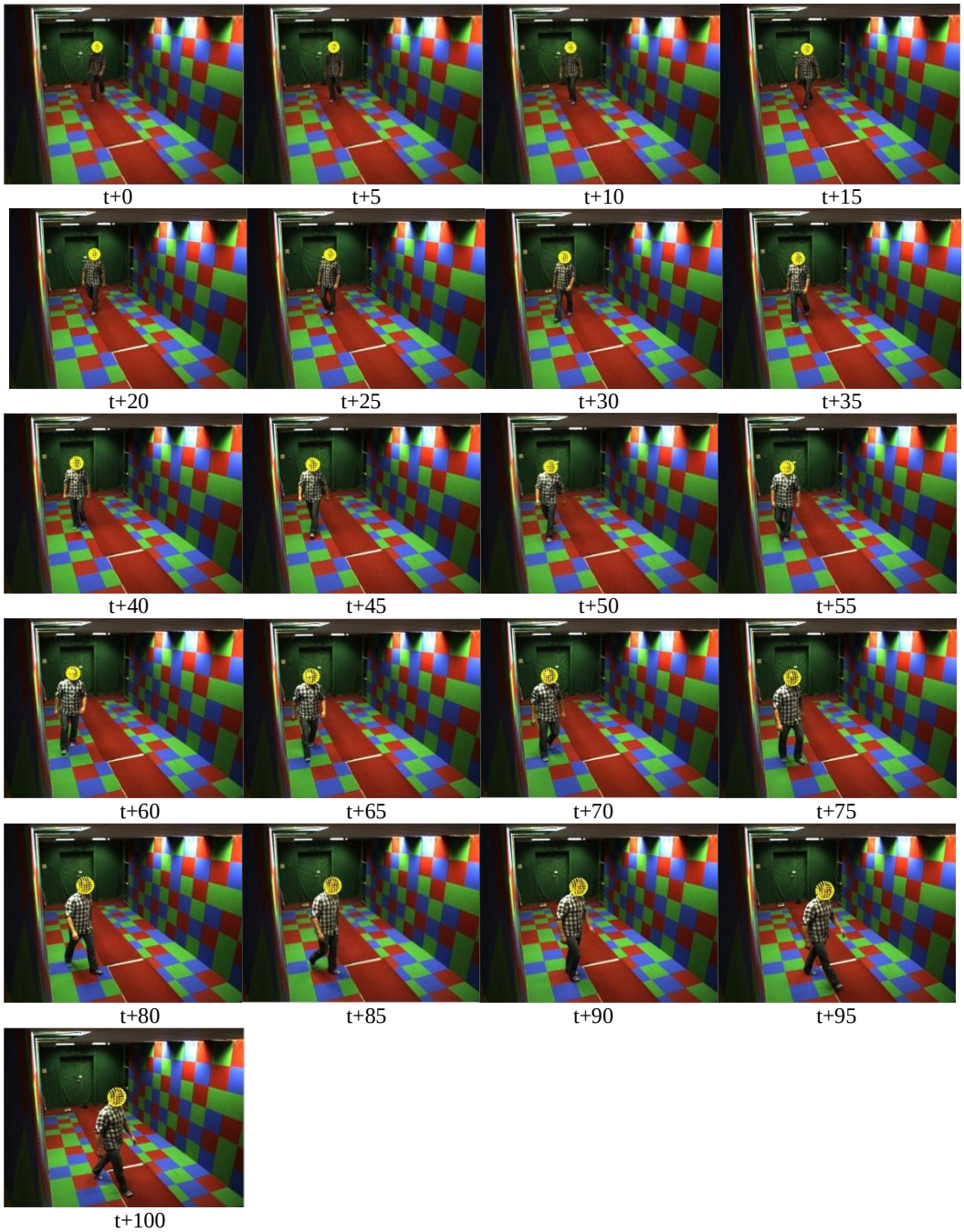


Figure A.7: Face region estimation by a 3D gait trajectory in the Biometric tunnel dataset

Figure A.8: Face region estimation by a 3D gait trajectory in the CAVIAR dataset