國立交通大學

應用數學系

碩 士 論 文

以模型基礎之下臉部追蹤的方法研究–主動外型模式

The Study of Model-based Face Tracking - Active Shape Model and Its Extension

研　究　生：連挺軒

指 導 教 授 ：林松山　教授

中 華 民 國 101 年 7 月

以模型基礎之下臉部追蹤的方法研究-主動外型模式

The Study of Model-based Face Tracking - Active Shape Model and Its Extension

研究生 ： 連挺軒　　Student : Ting-Hsuan Lien

指導教授 ：　林松山 教授　　Advisor : Song-Sun Lin

國立交通大學

應用數學系

碩 士 論 文

A Thesis

Submitted to Department of Applied Mathematics
College of Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master
In
Applied Mathematics
July 2012
Hsinchu, Taiwan, Republic of China

中 華 民 國 101 年 7 月

謝誌

首先要感謝我的指導老師-林松山教授在兩年來不論在做人處事方面，以及研究上給予我很多的指點，在老師常常面帶著微笑教導學生的同時，我也學會如何用微笑面對老師，做起事來特別開心，特別有來勁。不管在研究遇到瓶頸，生活遇到困難，知識遇到匱乏，報告突然辭窮時，老師常常在汪洋中伸出一雙手拉拔我．學生連挺軒在此致上十二萬分的感謝．

感謝何丹期學長協助下完成我人生第一篇論文，要是沒有學長的大力監督下，我現在也無法在這打致謝詞。同時也要感謝我的好朋友，蔡澤弘．楊祥鶴．高瑋琳．詹惠雯吳侑燊．曾茂清，許尚．李姿慧，有你(妳)們的陪伴，與你(妳)們同甘共苦，友難不同當，讓我這兩年來過的非常充實，誠心的祝福你們幸福快樂。也感謝我最親愛的父母，及哥哥在身體健康方面．心理方面給予我最大的支持，今天有這樣的成果，有一半歸於你們。

最後，也祝福我自己，未來的事業上，步步高升，跌跌撞撞，再接再厲，勇敢不放棄的精神，繼續向前衝。

謹以此論文獻給有緣人。　　　　連挺軒 2012.7.11

以模型基礎之下臉部追蹤的方法研究–主動外型模式

研究生：連挺軒 指導教授 ： 林松山

國立交通大學

應用數學系（研究所）碩士班

摘要

影像對應在電腦視覺裡常常被拿來應用，主要是在討論特徵點之間的對應關係。針對影響對應, 近幾年來發展快速, 從 Active contour model, Active shape model, 到 Active appearance model。此篇論文中，將探討主動外型模式方法(Active shape model)如何追蹤影像，並且將其延伸處理三維影像資訊，並討論會遇到的問題，以及提供有效的解決方法。藉由此方法，希望能夠應用在三角網格上的重建, 將來在數學操作上，給予重大的幫助。

The Study of Model-based Face Tracking - Active Shape Model and Its Extension

Student: Ting-Hsuan Lien　　Advisor : Song-Sun Lin

Department of Applied Mathematics

National Chiao Tung University

July, 2012

Abstract

Image correspondence is a fundamental problem which discusses the relation of features between images of the same scene. Various approaches have been proposed to achieve this goal. In this thesis, we discuss the method of Active Shape Model, a straight forward approach for tracking the correspondences between images. The active shape model has been shown to be a practical and robust solution for matching features in 2D images. However, applying this technique to the matching of features in 3D point clouds still fail to produce convincible results. In this thesis, we will review the active shape model by splitting into the preprocessing and run-time stages. discuss the variation in each step, and its extension to the 3D point cloud.

Contents

Chapter 1

Introduction

Shape correspondence, is a fundamental step in many computer vision and image processing applications such as image recognition, 3D reconstruction, object tracking, and image registration. The step has much challenge since we must construct a statistical shape model such as the point distribution model. The approaches for solving image correspondence can be classified into two classes: model-based and feature-based. Feature-based methods extract features in both images and use some higher-order constraint [14] or pair-wise [34] to solve correspondence. Unfortunately, feature-based approaches fail to detect correspondence when there are strong changes in pose and expression. Model-based approaches align appearance features with respect to a model. The model is learned from hand-labeled, and construct a sequence of template model to match new image. But it lacks of generalization to untrained situations. In this section, we will describe the motivation of our study, introduce the overview of the proposed face feature location system, and then give the organization of this thesis.

1.1 Motivation

We found some flaw as we obtained image data from 3D scanner, like holes, or missing data, especially noise, usually we handle it by artificial. Although it is easy to modify image data, but if the more manmade we add, the more image distortion will be happened. In view of above mentioned reasons, we as far as possible extract other image data we have known to recover image. Image registration is one of way that can solve this problem, hence we started surveying on image tracking. First, we consider facial tracking which can make sense on our experiment. Active shape model is a powerful tool to describe the sequence of image, especially face alignment. We expect this tool can be extend to deal with any types of image data.

1.2 Organization

In order to present a systematic overview of the topic, we have divided this article into several parts. In Section 2, we will describe the works related to our study. Section 3 gives the active shape model methods in details and 3D extend. In section 4, we will present an overview of the application of image tracking. The conclusions and future works are given in Chapter 5.

Chapter 2

Related works

In this section, we review the related work about the face tracking in two part, face detection and tracking method.

2.1 Face detection

Before we tracking face, the most important process is to detect the face in an image. Here, we summarize method is in four classification to explain how to extract feature point in object and recent works . Here summarizes methods and representative works for face detection in a single image as shown in Table 2.1.

(A) **Feature invariant approaches:**[47]

This approach aims to find structural features that exist even when the pose, viewpoint, or lighting conditions vary, and then use the structural features to locate faces. These methods are designed mainly for face localization. Facial features such as eyebrows, eyes, nose, mouth are commonly extracted using edge detectors [7]. Researcher try to find the features of faces for detection. Leung et al. [30] use

Approach	Medium	Reference
Feature invariant	Facial Features	Grouping of edges method [30]
	Texture	Space Gray-Level Dependence matrix of face pattern [15]
	Skin Color	Mixture of Gaussian [49]
	Multiple Features	Integration of skin color, size and shape [29]
Template matching	Predefined face template	Shape Template Method [23]
	Deformable templates	Active Shape Model, Active Appearence Model [19]
Appearance-based	Eigenface	Eigenvector decomposition and clustering [46]
	Distribution-based	Gaussian distribution and multilayer perception [43]
	Neural Network	Ensemble of neural networks and arbitration schemes [39]
	Support Vector Machine	SVM with polynomial kernel [35]

Figure 2.1: Categorization of methods for face detection in a single image.

probability to locate a face from a chaotic scene based on local feature detectors. Their idea is to find the arrangement of certain facial features that is most likely to be a face pattern. Yow and Cipolla [51] presented a feature based method that uses a large amount of samples from the visual image and their contextual samples. They use Gaussian filter to obtain the points they interest, and then define the measurement on these point. An image region becomes a valid facial feature candidate if the Mahalanobis distance between the corresponding feature vectors is below a threshold. Subsequently, this approach has been enhanced with active contour models by M. Kass et al. [24]. Augusteijn and Skufca [1] developed a method that deduces the presence of a face through the identification of face-like textures. They use second order statistical features to compute the measurement of textures on the sub-images of 16×16 pixels [21]. Y. Dai et al.[15] used a cascade correlation neural network to classify three types of features: skin, hair, and others.

(B) **Template matching methods:**

In template matching, a standard frontal face pattern is manually predefined or parameterized by a function. Sakai et al.[40] attempt to detect frontal faces in photographs. They set the sub-templates on features of face. Each sub-template is defined in terms of line segments which extracted based on greatest gradient change and then matched against the sub-templates, and we obtained the candidate positions from their correlation. Craw et al. [13] presented a localization method based on a shape template of a frontal view face. After that, Yuille et al. [52] used deformable templates to model facial features that fit an a priori elastic model to facial features. Deformable templates is defined by the minimum energy function to link the edge of image. Later, this conception was extended to the method "Snake" proposed by M. Kass et al. [24]. Lam and Yan [31] used snakes to locate the head boundaries with a greedy algorithm in minimizing the energy function. Lanitis et al. [32] use point distribution model and active shape model to represent image shapes and intensity information. Cootes and Taylor [20] applied a similar approach to localize a face in an image. They define rectangular regions of the image containing instances of the feature of interest. Cootes and Taylor applied a similar approach to localize a face in an image by rectangular regions of the image containing instances of the feature of interest [11].

(C) **Appearance-based methods:**

Appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant feature of face and non-face images. An early case to use eigenvectors in face recognition was done by Kohonen [28]. He proposed a simple neural network to perform face recognition for aligned and normalized face images. These eigenvectors are known as eigenfaces. After that, Turk and Pentland [46] applied principal component analysis [22] to face recognition and detection. Sung and Poggio [43] proposed distribution-based system for face detection. This system consists of distribution-based models for face or non-face patterns and a multi-inductor classifier. Neural networks have been applied successfully in many pattern recognition problems, such as optical character recognition, object recognition, and autonomous robot driving. Agui et al. [2] used hierarchical neural networks. They used parallel sub-networks in which the inputs are intensity values from an original image and intensity values from Sobel filter on image. Sobel filter is used in image processing, particularly within edge detection algorithms. The first used Support Vector Machine to face detection was worked by Osuna et al. [35] support vector machines re supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis (Quote from Wikipedia). The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the input, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

2.2 Model-based image correspondence

Model-based methods are able to solve the correspondence in difficult situations since we can construct prior knowledge of the expected shape and appearance of an object class.

(A) **Active contour models:**

The active contour model, or snake, proposed by M. Kass et al. [24] is an energy-minimizing spline guided by external constraint forces and influenced by image forces that pull it toward features such as lines and edges. Snakes do not solve the entire problem of finding contours in images. The most advantage of snake is that it is active and information from a higher level process can be used. Changing in high-level interpretation can carry out forces on a snake as it continues its minimization. This approach can represent the boundary, internal features, such as facial expression. The method Active contour model of Kass et al. [24, 36] is similar as active shape model. It describes flexible contour models which are attracted to image feature. Their main idea is that model evolution is driven by two energies: an external energy that adapts the model to the image data and an internal energy that stabilizes its shape based on general smoothness.

(B) **Active shape models:**

Active shape model (ASM) is statistical model of the shape of object which iteratively deforms to fit to an example of the object in a new image [20, 41]. The shapes are constrained by a statistical shape model to vary only in ways seen in a training set of labeled examples. The shape model in ASM is given by the principal components of vectors of landmark points. The gray-level appearance model is limited to the border of the object and consists of the normalized of profiles centered at each landmark that run perpendicular to the object contour. The The fitting procedure is an replacement of landmark location and model fitting in a multi-resolution framework [12]. Since PCA is linear for shape models, Sozou et al. [44] try to use nonlinear models that can be more suitable. After compute the Mahalanobis distance to search new possible position, Behiels et al. [4] used dynamic programming to find new positions for the landmarks, instead of moving each point to the position with the lowest distance. After that, Cootes et al. [45] countered the the proposed landmark displacement with using weighted fitting of the shape model.

(C) **Active appearance models:**

Active Appearance models (AAM), proposed by Cootes et al. [8]. This method is also a statistical model of the shape of object. It is a powerful generative method for modeling deformable objects. The model divided the shape and the texture variations of objects, which is followed by an efficient gradient-based model fitting method. Due to the flexible and simple framework, AAM has been widely applied in the fields of computer vision. However, difficulties are met when it is applied to various practical issues, which lead to a lot of robust improvements to the model. The AAM is a widely used method for model based vision showing excellent results in a variety of applications. Comparing with Active shape model method, ASM match a set of model points to an image constrained by a statistical model of shape, the scope is restricted on profile composed by feature points we labeles and AAM match both the position of the model points and a representation of the texture of the object to an image. AAM could control a full model of appearance, which represents both shape variation and the texture of the region covered by the model.

This can be used to generate full synthetic images of objects.

(D) **3D morphable models:**

3D morphable face model [6] is a well-known computer graphics technique based on a vector space that is constructed by any convex combination of shape and texture vectors. The main idea behind the morphable face model approach is that given a sufficiently large database of 3D face models so that any arbitrary face can be generated by morphing between the ones in the database.

2.3 Feature-based image correspondence

Feature-based image matching extracts distinct features from images then identifies those features that correspond to one another. Features may a global property of the image like the average gray value or a local property like points, edges or circles within the image. The detected features should be invariant to geometric transformation. Here we reviews two method on feature-based image correspondence.

Scale-invariant feature transform:

Scale-invariant feature transform (SIFT) is an algorithm in computer vision to detect and describe local features in images proposed by D.Lowe [33]. Lowes SIFT is one of powerful method to construct geometric invariant features to match rigid objects. Its extensions [5, 34] have been successfully applied to many problems.

Chapter 3

Active shape model

3.1 Overview

The *Active Shape Model* (ASM) is first proposed by Cootes and Taylor [9] to find and trace the specific structure feature in image. It is based on the statistic method that first define the interesting shape feature and then find the features in target images. The ASM consists of two stages: the preprocessing stage and the run-time stage. In the preprocessing stage, the specific shape feature is defined by learning the mean shape of it and some parameters to reveal the variations in a given data set using statistic methods. The specific shape feature in each sample of the data set can be extracted by either user specify or by some shape descriptors. In this thesis, we assume that such shape features are known. In the run-time stage, the mean shape is served as a template model to search the similar shapes in the target images.

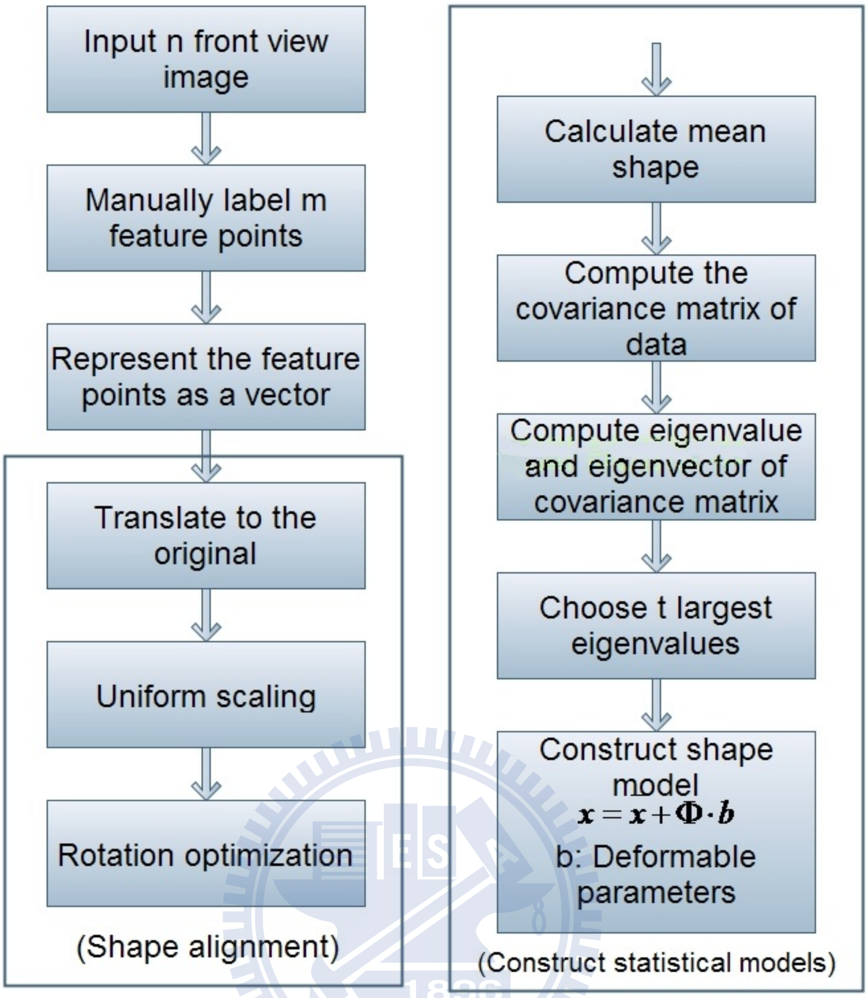


Figure 3.1: Training process.

3.2 Preprocessing

To trace the shape feature using ASM, we first need to define what shape we focus on. In the preprocessing stage, the shape feature we are interest is learned by a training process. A mean shape follows by the parameters to denote the shape variation is learned from the training process which describes the average of the shapes in the data set. In this section, we describe the methods of how the mean shape is obtained and how to align the shapes. It can be divided into two parts in generally, landmark feature points and construct the shape model. Fig. 3.1 shows the flow chart of the preprocessing process.

3.2.1 Landmark feature point

In order to locate a structure of interesting object, we must first build a model for it. The shape model for ASM is base on the statistic analysis. Before a statistical shape model is built to represent the feature, a set of annotated images of typical examples is required. The first step is to landmark the feature points of the shape we want in each of the training image of the data set. We begin by the definition of feature point and shape representation in the ASM method. Therefore we define the shape representation and some property so that easy to allows us to determine the direction of the boundary at a given point.

Definition(Feature point)
A feature point is a point on the image that represents some characteristics of the object in the image.

Definition(Shape representation)
A structural shape descriptor formed by the spatial relation of a set of n feature points, which can be represented by a $2n$ dimension vector.

Given some image from camera, we have to represent the profile of object by a set of points. First, we manually label some feature points from boundary of object in image. Each labeled point represents a particular position of the object as shown in Fig. 3.2. Bookstein [3] and Kendall [25] mentioned three criteria to select the feature points.

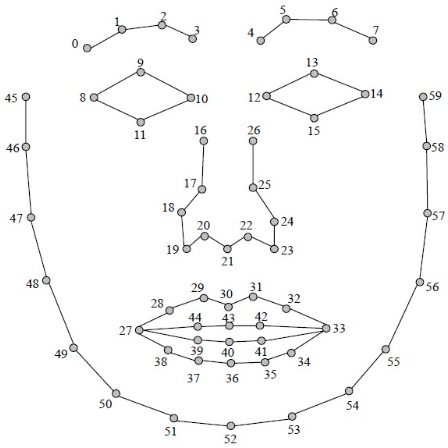


Figure 3.2: Label the feature points on a face

1. Outstanding points on the image, such as the sharp corners of boundary or tip of the mouth.

2. Independent points, such as high curvature points or isolated points.

3. Interpolated point from 1 & 2, this is used for generating the reference points so that make more sample points which can be consistently match from one image to the other.

3.2.2 Shape alignment

After the feature points on all training images are labeled, the next step is to derive a mean shape from all the shape representations. Since the shape in each of the training image may be well aligned, we need to transform all the shapes into a common coordinate frame. Finding the transformation between shape of different training images requires solving a minimization problem that minimize the sum of distance between the two shapes through the transformation as in Eq. 3.1.

$$D(s_i, m) = \sum_{i \in \mathbb{N}} \|T(s_i) - m\|) \tag{3.1}$$

where T is the aligning transformation, s_i is a shape in training set, and m is the mean shape.

Such problem is generally difficult to be solved since it consists of a rigid transformation and a isomorphic scaling. The Procrustes analysis [17, 18, 38, 41] is a simple and efficient way to approximate the transformation. The Procrustes analysis is a rigid shape analysis that decompose the process into three steps. First, a translation transformation is performed for each shape that translate the geometric center to the origin of its local coordinate, as shown in Eq. 3.2.

$$(x'_i, y'_i) = (x_i - \bar{x}, y_i - \bar{y}) \tag{3.2}$$

where (x'_i, y'_i) is the new position of shape, $\bar{x} = \frac{x_1+x_2+...+x_n}{n}$, $\bar{y} = \frac{y_1+y_2+...+y_n}{n}$ is the gravity of shape.

Then, an isomorphic scaling transformation is obtained by mean square distance, as shown in Eq. 3.3.

$$(x''_i, y''_i) = (\frac{x'_i}{S}, \frac{y'_i}{S}) = (\frac{x_i - \bar{x}}{S}, \frac{y_i - \bar{y}}{S}) \tag{3.3}$$

where $S = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}}$ is the root mean square distance. Finally, the rotation matrix is obtained by minimizing the square distance between the shapes. Consider two shape with same number of feature points which the scale and translation have been aligned. Let

$$X_1 = (x_{11}, y_{11}, ..., x_{1n}, y_{1n}), X_2 = (x_{21}, y_{21}, ..., x_{2n}, y_{2n})$$

where X_1, X_2 are two shapes in training set. Assume

$$X_1' = (x_{11}', y_{11}', ..., x_{1n}', y_{1n}') = T(X_1)$$

where X_1' is the new shape after aligned, and T is rotation operator we optimize θ such that $\|X_2 - X_1{}'\|_2$ is minimum.

$X_2 \approx X_1' = [\mathrm{R}]X_1$, which [R]=T is rotation matrix, taking the derivative of $(x_{21} - x_{11}')^2 + (y_{21} - y_{11}')^2 + ... + (x_{2n} - x_{1n}')^2 + (y_{2n} - y_{1n}')^2$ with respect to θ and equal to zero (Least square method). We can imply

$$\theta = \tan^{-1}\left(\frac{\sum_{i=1}^{n} x_{2i}y_{1i} - y_{2i}x_{1i}}{\sum_{i=1}^{n} x_{2i}x_{1i} + y_{2i}y_{1i}}\right). \tag{3.4}$$

Although the Procrustes analysis works well for finding the transformation between shape in different images. However, once the variance of shape in training set is too large, the distortion of shape will be occur. Hence, Cootes et al. [9] proposed a weighted procrustes analysis to solve this problem, and can achieve a stabler results.

Let $x_1 = (x_{11}, x_{12}, ..., x_{1n})^T$ and $x_2 = (x_{21}, x_{22}, ..., x_{2n})^T$ be two shape vector with respect to two similar shapes with n feature points.

Define

$$M(s,\theta)\begin{bmatrix} x_{2j} \\ y_{2j} \end{bmatrix} = \begin{pmatrix} (s\cos\theta)x_{2j} - (s\sin\theta)y_{2j} \\ (s\sin\theta)x_{2j} + (s\cos\theta)y_{2j} \end{pmatrix} \tag{3.5}$$

θ: Rotation, s: Scale.

Then we minimize the weighted sum

$$E = (x_1 - M(s,\theta)[x_2] - t)^T W (x_1 - M(s,\theta)[x_2] - t). \tag{3.6}$$

$t = (t_x, t_y, ..., t_x, t_y)^T$: The translation of shape.

W : A diagonal matrix of weight for each point. For the kth point, $w_k = (\sum_{l=1}^{n} V_{R_{kl}})^{-1}$. R_{kl} is the distance between points k and l, $V_{R_{kl}}$ is the variance in this distance.

Let $a_x = s\cos\theta$, $a_y = s\sin\theta$, we acquire four equation from $E = 0$,

$$\begin{cases} a_x \sum_{i=1}^{n} w_i x_{2i} - a_y \sum_{i=1}^{n} w_i x_{yi} + \sum_{i=1}^{n} w_i t_x = \sum_{i=1}^{n} w_i x_{1i} \\ a_y \sum_{i=1}^{n} w_i x_{2i} + a_x \sum_{i=1}^{n} w_i x_{yi} + \sum_{i=1}^{n} w_i t_y = \sum_{i=1}^{n} w_i y_{1i} \\ a_x \sum_{i=1}^{n} w_i ({x_{2i}}^2 + {y_{2i}}^2) + t_x \sum_{i=1}^{n} w_i x_{2i} + t_y \sum_{i=1}^{n} w_i y_{2i} = \sum_{i=1}^{n} w_i (x_{2i} x_{1i} + y_{2i} y_{1i}) \\ a_y \sum_{i=1}^{n} w_i ({x_{2i}}^2 + {y_{2i}}^2) - t_x \sum_{i=1}^{n} w_i y_{2i} + t_y \sum_{i=1}^{n} w_i x_{2i} = \sum_{i=1}^{n} w_i (y_{1i} x_{2i} - x_{1i} y_{2i}) \end{cases} \tag{3.7}$$

$$\Leftrightarrow \begin{pmatrix} X_2 & -Y_2 & W & 0 \\ Y_2 & X_2 & 0 & W \\ Z & 0 & X_2 & Y_2 \\ 0 & Z & -Y_2 & X_2 \end{pmatrix} \begin{pmatrix} a_x \\ a_y \\ t_x \\ ty \end{pmatrix} = \begin{pmatrix} X_1 \\ Y_1 \\ C_1 \\ C2 \end{pmatrix}, \tag{3.8}$$

where

$$
\begin{array}{ll}
X_1 = \sum_{i=1}^{n} w_i x_{1i}, & Y_1 = \sum_{i=1}^{n} w_i y_{1i}, \\
X_2 = \sum_{i=1}^{n} w_i x_{2i}, & Y_2 = \sum_{i=1}^{n} w_i y_{2i}, \\
W = \sum_{i=1}^{n} w_i, & Z = \sum_{i=1}^{n} w_i ({x_{2i}}^2 + {y_{2i}}^2), \\
C_1 = \sum_{i=1}^{n} w_i (x_{1i} x_{2i} + y_{1i} + y_{2i}), & C_2 = \sum_{i=1}^{n} w_i (y_{1i} x_{2i} - x_{1i} + y_{2i}).
\end{array}
$$

Then, a_x, a_y, t_x, t_y can be solved by inverse matrix. Hence, we obtain θ, s, t.

Algorithm. 1 describes the process of shape alignment. Fig. 3.2.2 illustrates the shapes in different images before and after alignment.

3.2.3 Construct the active shape model

Suppose now we have n shapes which are aligned into a common coordinate frame. If we can model this distribution, we can generate new examples. An effective approach is to apply *Principal component analysis* to reduce the dimensionality of the training set. Our goal is to construct a statistical shape model consists of a mean shape to describe the global structural feature and some parameters for representing the detail variations from a collection of training samples.

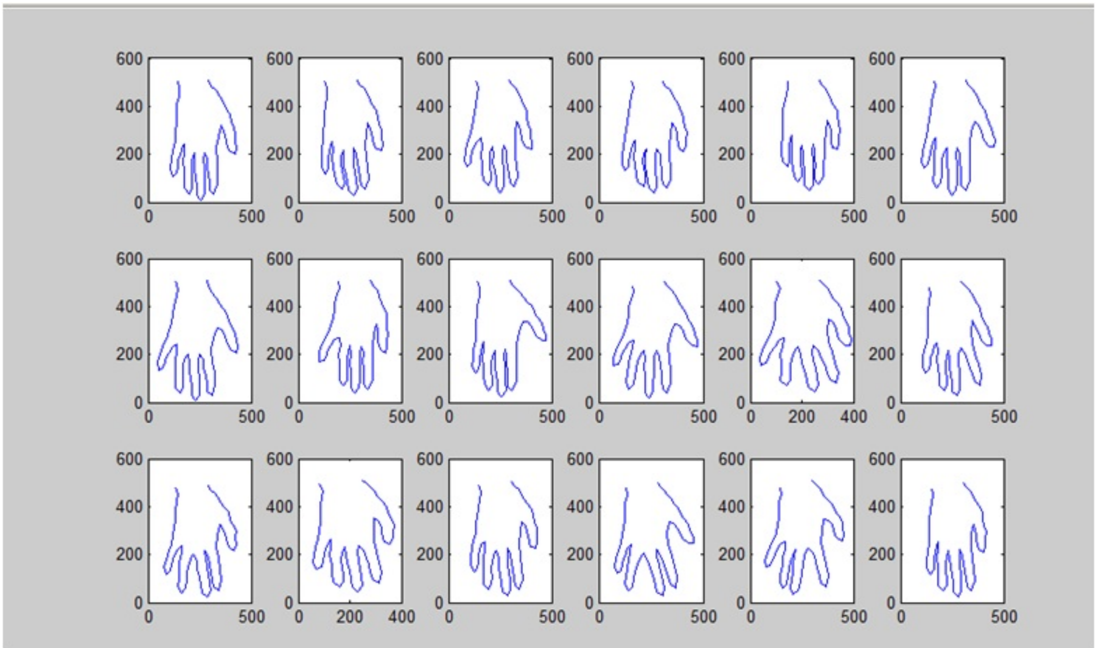
We denote the *Mean Shape* as an average shape of the n aligned shapes in the training images, which can be defined by Eq. 3.9.

$$
\bar{x} = \frac{1}{n} \Sigma_{i=1}^{n} x_i, \tag{3.9}
$$

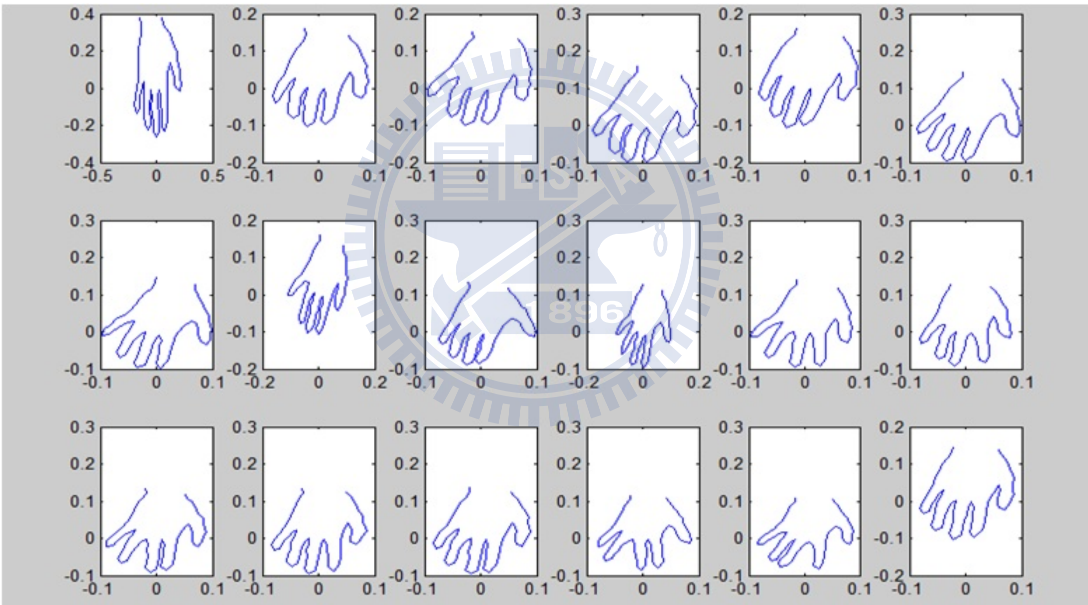
where x_i is the shape after aligned.

The mean shape is an average shape for representing the structural feature of shape. However, matching the shape we obtained using the mean shape may not be robust since the obtained shape may different from the mean shape. We must take the variation of shape into account.

The variation of shape is a vector in n dimension coordinate. Matching shape



(a) Before alignment



(b) After alignment

Figure 3.3: Alignment

in such high dimension coordinate is time and storage consuming. Since variations of shape are similar between different training samples, it is possible to reduce the need of variation defined in such high dimension. The *Principal Component Analysis* (PCA) [22] is a technique to reduce the dimensionality of the training set.

Let

$$\begin{aligned} x_1 &= (x_{11}, x_{12}, ..., x_{1m}) \\ x_2 &= (x_{21}, x_{22}, ..., x_{2m}) \\ &\vdots \\ x_n &= (x_{n1}, x_{n2}, ..., x_{nm}) \end{aligned} \tag{3.10}$$

be n m-dimension points.

Let

$$m = \frac{1}{n} \sum_{k=1}^{n} x_k. \tag{3.11}$$

Define

$$\begin{aligned} J_0(x_0) &\equiv \sum_{k=1}^{n} \|x_0 - x_k\|^2 \quad \text{(Square error criterion function)} \\ &= \sum_{k=1}^{n} \|x_0 - m + m - x_k\|^2 \\ &= \sum_{k=1}^{n} \|x_0 - m\|^2 - 2\sum_{k=1}^{n} (x_0 - m) \cdot (x_k - m) + \sum_{k=1}^{n} \|x_k - m\|^2 \\ &= \sum_{k=1}^{n} \|x_0 - m\|^2 - 2(x_0 - m) \cdot \underbrace{\sum_{k=1}^{n} (x_k - m)}_{=0} + \sum_{k=1}^{n} \|x_k - m\|^2 \\ &= \sum_{k=1}^{n} \|x_0 - m\|^2 + \underbrace{\sum_{k=1}^{n} \|x_k - m\|^2}_{\text{independent with } x_0} . \end{aligned} \tag{3.12}$$

We imply $x_0 = m$ as $J_0(x_0)$ is minimum. Assume there is a straight line L pass through x_0 such that the square error from x_k to L is minimum. Set the correspondence $x_k{}'$ on L of x_k. i.e $L : x_k{}' = x_0 + a_k e$, where a_k is coefficient and e is an unit vector.

$$J_1(a_1, a_2, ...a_k, e) \equiv \sum_{k=1}^{n} \|x_k{}' - x_k\|^2 \tag{3.13}$$
$$= \Sigma_{k=1}^{n} \| \underbrace{x_0}_{=m} + a_k e - x_k\|^2$$
$$= \Sigma_{k=1}^{n} \|a_k e\|^2 - 2\sum_{k=1}^{n} a_k e \cdot (x_k - m) + \underbrace{\sum_{k=1}^{n} \|x_k - m\|^2}_{\text{independent with } a_k}$$

Assume e has been known, let

$$J_1{}' = \sum_{k=1}^{n} a_k{}^2 - 2\sum_{k=1}^{n} a_k e \cdot (x_k - m) \tag{3.14}$$
$$\Rightarrow a_k = e \cdot (x_k - m) \text{ as } J_1{}' \text{ is minimum.}$$

$$J_1 = \sum_{k=1}^{n} a_k{}^2 \|e\|^2 - 2\sum_{k=1}^{n} \|a_k\|^2 + \sum_{k=1}^{n} \|x_k - m\|^2 \tag{3.15}$$
$$= \sum_{k=1}^{n} a_k{}^2 + \sum_{k=1}^{n} \|x_k - m\|^2$$
$$= -\sum_{k=1}^{n} [e \cdot (x_k - m)]^2 + \sum_{k=1}^{n} \|x_k - m\|^2$$
$$= -\sum_{k=1}^{n} e^t (x_k - m)(x_k - m)^t e + \sum_{k=1}^{n} \|x_k - m\|^2$$
$$= -e^t S e + \sum_{k=1}^{n} \|x_k - m\|^2$$

,where S is covariance matrix.

Use Lagarange multipliers method to find e such that $e^t Se$ is maximum. Let

$$u = e^t Se - \lambda(e^t - 1) \tag{3.16}$$

$$\Rightarrow \frac{\partial u}{\partial e} = 2Se - 2\lambda e = 0 \tag{3.17}$$

$$\Rightarrow Se = \lambda e \tag{3.18}$$

$\Rightarrow e$ is the eigenvector of covariance matrix S.

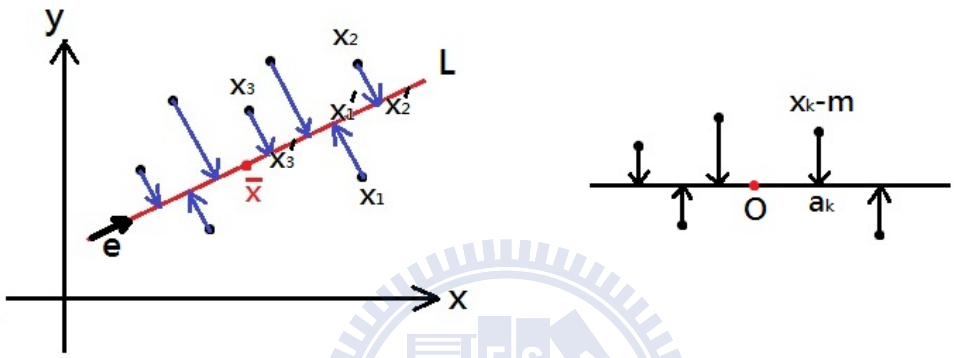


Figure 3.4: Apply a PCA on 2D shape vector, we obtain line L is the principal axis, then x_k can be approximated by the nearest point on L. b is the distance along the axis from the mean of the closest approach to x.

Applying to the shape, suppose x_k is k-th aligned shape for $k = 1, .., n$ and $\bar{x}$ is the mean shape, given by

$$\bar{x} = \frac{1}{n}\sum_{k=1}^{n} x_i, \tag{3.19}$$

and S is covariance of the data, given by

$$S = \frac{1}{n-1}\sum_{k=1}^{n}(x_k - \bar{x})(x_k - \bar{x})^T. \tag{3.20}$$

We can generate new examples by using PCA, the model describes as

$$x \approx \bar{x} + \Phi \mathbf{b}. \tag{3.21}$$

where $\Phi = (\phi_1, \phi_2, ..., \phi_t)$ is the $n \times t$ matrix of eigenvectors of the covariance matrix. The number of eigenvectors to retain, t, can be chosen by the proportion of variation captured by the corresponding eigenvectors.

$$\sum_{i=1}^{t} \lambda_i \geq f_v V_T, \tag{3.22}$$

where f_v is proportion of variation captured by the corresponding eigenvectors, λ_i is the ith eigenvalue, and V_T is total variance in the training data is the sum of all the eigenvalues. We order the eigenvalue λ_i of S, and keep first t large eigenvalue number, the column of Φ are the eigenvectors corresponding to the largest eigenvalues of S. The remaining eigenvalues represent noise in the form of numerical errors. $\mathbf{b}$ is a t dimension vector defined a set of parameters of a deformable model. We usually constrain the value of $\mathbf{b}$ within $\pm m\sqrt{\lambda_i}$ when fitting the model to a set of points, where m is between 2 and 3.

$$\mathbf{b} = \Phi^T (x_k - \bar{x}). \tag{3.23}$$

Each mode b_i describes one way in which the shapes in the training set tend to vary from the mean.

Algorithm. 2 describes how to construct the shape model.

Algorithm 1 Shape alignment

Input: n samples of 2D image.

1: Label m feature point manually in each image.
2: Translating to shape vector $x_i = (x_{i1}, y_{i2}, ...x_{im})^T$, for $i = 1, 2, ..., n$
3: Translating the gravity of each x_i to original

$$(x_{ij}, y_{ij}) \leftarrow (x_{ij} - \frac{1}{m}\sum_{k=1}^{m} x_{ik}, y_{ij} - \frac{1}{m}\sum_{k=1}^{m} y_{ik})$$

4: Select one shape as reference shape denoted x_R, and scale this shape to unit size (Initial mean shape $\bar{x_0}$)

$$\bar{x_0} = \frac{x_R}{\|x_R\|} = \frac{x_R}{\sqrt{\sum_{k=1}^{m}(x_{R,k})^2 + (y_{R,k})^2}}.$$

Rotate,scale,and translate each shape to align with reference shape.

5: Calculate the mean shape from the aligned shape.
6: Use current mean shape align with reference shape, denoted $\bar{x_0}'$
7: If converge (mean shape does not change in threshold), end process. Else, $\bar{x_0} \leftarrow \bar{x_0}'$ return step 6.

Output: Aligned training set.

Algorithm 2 Construct shape model.

Input: n aligned shapes.

1: Compute the mean shape of the aligned shapes $\bar{x} = \sum_{i=1}^{n} x_i$.
2: Compute the covariance matrix of the aligned shapes,

$$S = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T.$$

3: Compute and order(decreasing) the eigenvector Φ_i with respect to corresponding eigenvalue λ_i of S.
4: Select the t large eigenvalues such that $\sum_{i=1}^{t} \lambda_i \geq f_v \sum_{i=1}^{2} n\lambda_i$, where $0 < f_v \leq 1$, the proportion of variation. $\Phi = [\Phi_1|\Phi_2|...|\Phi_t]$.

Output: Shape model.

3.3 Run-time

In the preprocessing stage, a mean shape and its variation is constructed from a given set of training data. Such mean shape is served as a template shape model for matching the features new images. To find the shape features on the new image, we must find the parameters of the shape variations which best fit the template shape model to the image. Since there is no information about the shape on new image, we are unable to generate new profile model directly.

Now the problem is how to derive the useful information from the shape model we generated. Researchers usually construct measurements to seek where the boundary of object is. If we assume that the shape model represents boundaries and strong edges of the object, a useful measure is the distance between a given model point and the nearest strong edge in the image. The best approach is to sample along the profile normal to the boundary in the training set, and building an intensity model from the grey-level structure. we observe that edge of testing image will occur at instantaneous rate of change in intensity model.

Fig. 3.5 illustrates the flow chart of the run-time process of ASM. The run-time process starts by inputting a new image that contains the shape we interest. Then, setting a shape from shape model we constructed before and align to same coordinate frame with new image. After that, we compute Mahalanobis distance to estimate the feature points on the new image. Once the feature points have found, we can define the adaptive affine transform T between two cluster of feature points data and obtain new image shape. Finally, we inverse image shape to model shape and then calculate average error and maximum error whether converge or not. If not, choose shape variation b by Eq. 3.23 and regenerate a new shape from shape model, then restarting the process until converge.

In the following subsections, we describe the processes of fitting the template shape model to the new images by dividing into two steps : the feature extraction step and the model fitting step.

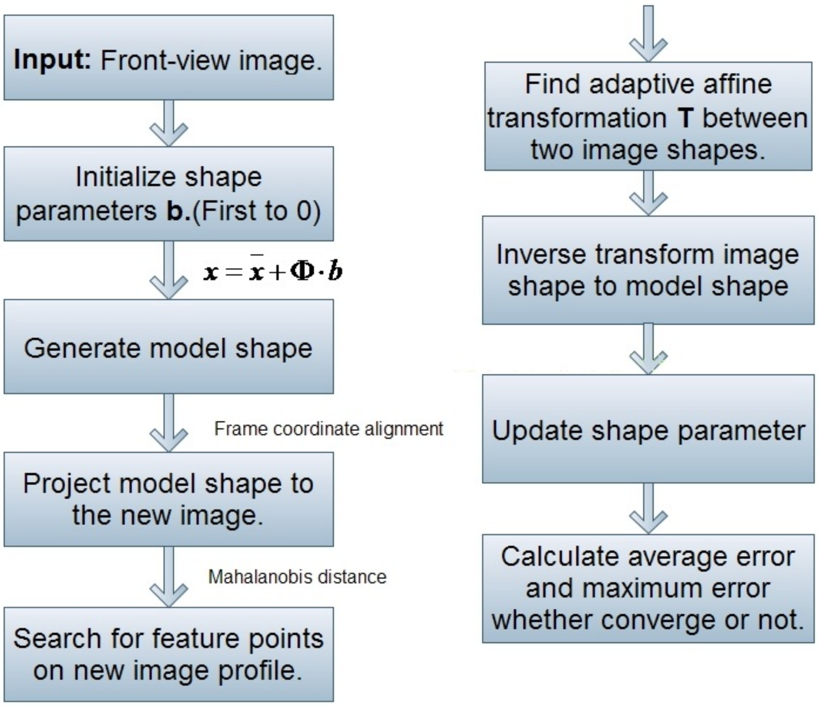


Figure 3.5: Model fitting process

3.3.1 Feature Extraction

The shape on the image can be identified by looking at the gray-level color distribution of the image. The gray-level appearance model describes the shape structure around each landmark by analyzing the strength of the gray-level pixel color gradient to be boundary. Cootes and Taylor [10] used the normalized first derivatives of these profiles to build the gray-level appearance model. To form the profile vector at a landmark, we first sample k pixels on each side of model point perpendicularly, as shown in Fig. 3.6. We define a model point (x_i, y_i) with direction perpendicular by rotating the vector of (x_i, y_i) and (x_{i-1}, y_{i-1}) over 90^o. Hence, we have $2k+1$ grey-level values on each model point. To reduce the effects of global intensity changes, we normalize these gray-level values by dividing the sum of absolute element values which denoted by vector g_i in i-th training image. After that, the Mahalanobis distance [37] is given by

$$f(g_i) = (g_i - \bar{g})^T S_g (g_i - \bar{g}) \tag{3.24}$$

where $\bar{g}$ is mean, and S_g is covariance matrix of g_i, respectively. The Mahalanobis distance is a metric that can be used to measure the dissimilarity between two vectors. Minimizing $f(g_i)$ is equivalent to maximizing the probability that g_s comes from the distribution. Finally, we obtain the approximate feature point of new image.

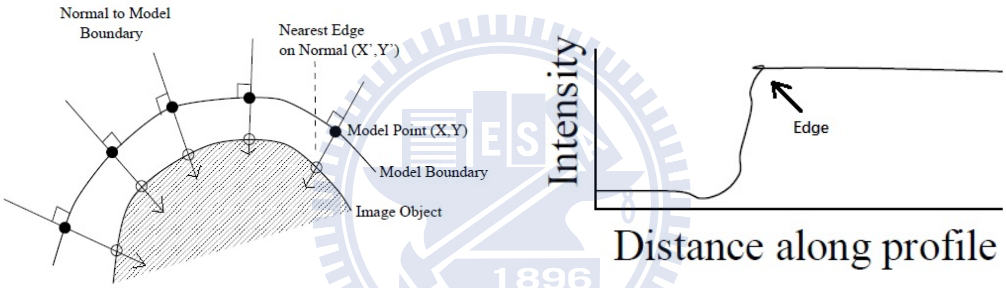


Figure 3.6: Left: At each model point sample along a profile normal to the boundary and searching for strong edges. Right(Line chart): The intensity respect to distance along profile. [10]

3.3.2 Model Fitting

Once we approximate features on new image by using gray-level appearance model, our next step is to find the new affine transformation T between each new image shape and template shape model. The transformation T on the 2D image can be described by Eq. 3.25.

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{bmatrix} S_c\cos\theta_c & S_c\sin\theta_c \\ S_c\sin\theta_c & S_c\cos\theta_c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{3.25}$$

where x_c and y_c are the translation vector, S_c is the isomorphic scaling value, and θ_c is the rotation angle.

The problem of finding the transformation T is the same as the shape alignment process, which can be calculated the Procrustes analysis. After that, we determine the average error and maximum error to control converge threshold.

Algorithm 3 describes the process of modeling fitting.

Algorithm 3 Model fitting.

Input: A front-view image from scanner, and Φ.

1: Initial the shape deformable parameters $\mathbf{b}$ (First to zero).
2: Generate the shape model by $x = \bar{x} + \Phi\mathbf{b}$. ($\bar{x}$ mean shape)
3: Project x to the image $x_{image} = T_1(x)$. (T_1: Operator of coordinate translation)
4: Search for the best image shape y_{image} according Mahalanobis distance profile model or the distance of crisscross profile model.
5:
6: Find x_c, y_c, S_c, θ_c between x_{image} and y_{image}.
7: Construct adaptive affine transformation T_2,

$$T_2\begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{bmatrix} S_c\cos\theta_c & S_c\sin\theta_c \\ S_c\sin\theta_c & S_c\cos\theta_c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

8: $y \leftarrow {T_1}^{-1}(T_2(x_{image}))$.
9:
10: $\mathbf{b}' \leftarrow \Phi^T(y - \bar{x})$.
11:
12: Estimate metric of b,

$$e_1 = \frac{1}{t}\sqrt{\sum_{i=1}^{t}(b_i - b_i')},$$

$$e_2 = max\|b_i - b_i'\|,$$

where e_1 denotes average error, e_2 denotes maximum error.
13: If not converge, $b \leftarrow b'$, return step 2. Else, process end.

Output: Profile model.

3.4 Extend to 3D

The ASM method we described is based on the 2D coloring image. However, such technique can also be applied to handle the range images obtained from the 3D scanner. In this section, describe the general process of how to extend the 2D ASM method to handling the range images. The range image is a 2D image which each pixel stores the distance of captured surface point from the observer. Thus, 3D surface of the observed object can be reconstructed from the range image. The general process of matching features in range images using ASM is the same as the way in 2D coloring images. However, due to the different meaning of pixel values, there still face lots of difficulties in matching features in range image using ASM. We describe the ASM method for matching the features on range image by first review the overall process of ASM on the range image, and then point out the challenges in finding the featured landmarks in range image and the transformation in 3D space.

we extend this technique to handle 3D image data. The active shape model we mentioned before can only handle rotations which are present in the manually landmark training 2D images. If we turn into 3D image data, although 3DASM is no more than rotation, translation and scale parameters, as presented in [26], it will not be robustness. For example, as we considering on facial tracking, a problem arises when the head turns to one side and only a profile view is visible. Then the number of feature points on two image will different and cause out of correspondence, we call this trouble as occlusion or data missing. Moritz Kaiser, et al.[27] propose brightness constancy assumption to decrease error norm of image correspondence. Ying Chen, et al.[50] propose active conditional models, it learns the conditional relation between a reference view of the object and other view points. In future, dense real-time 3D face tracking and 3D deformation modeling will be a challenging study for researcher.

Chapter 4

Future work

Face correspondence is a challenging and interesting problem in computer vision, the active shape model is one of the simplest methods for image correspondence. In this thesis, we arrange the development of active shape method in the last two decades, include in 2D and 3D spaces. Even though there is no significant effect in facial recovery, there is still work to be done. Future work includes how to handle the image constructed by triangular mesh and combine other model-based method to give the best correspondence results for a class of shapes.

Bibliography

[1] M. F. Augusteijn, T. L. Skujca, *Identification of human faces through texture-based feature recognition and neural network technology,* in Proc. IEEE Conf. Neural Networks, San Francisco, CA, Mar.28-Apr.1, 1993, pp.392-398.

[2] T. Agui, Y. Kokubo, H. Nagashashi, T. Nagao, *Extraction of face recognition from monochromatic photographs using neural networks,* in Proc. 2nd Int. Conf. Automation, Robotics, and Computer Vision, vol.1, Singapore, Sep.15-18, 1992, pp.1881-1885.

[3] F. L. Bookstein, *Morphometric Tools for Landmark Data: Geometry and Biology.* Cambridge University Press, 1997.

[4] G. Behiels, D. Vandermeulen, F. Maes, P. Suetens, and P. Dewaele, *Active shape model-based segmentation of digital X-ray images,* in Lecture Notes in Computer Science. Berlin, Germany: Springer-Verlag, 1999, MICCAI 99, pp. 128V137.

[5] H. Bay, T. Tuytelaars and L. V. Gool, *SURF: Speeded Up Robust Features,* in European Conference on Computer Vision, 2006, pp. 404V417.

[6] V. Blanz and T. Vetter, *A Morphable Model for the Synthesis of 3D Faces,* Computer Graphics Proc. SIGGRAPH 99, pp. 187-194, 1999.

[7] J. Canny, *A Computational Approach to Edge Detection,* IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679-698, June 1986.

[8] T. F. Cootes, G. J. Edwards, C. J. Taylor, *Active appearance models,* In: H. Burkhardt, B. Neumann (Eds.), Proc. ECCV, vol. 2. 1998.

[9] T. F. Cootes, C. J. Taylor, D. H. Cooper, J. Graham, *Active shape models - their training and application.* Computer Vision and Image Understanding, 61(1):38−59, 1995.

[10] T. F. Cootes, C. J. Taylor, *Statistical models of appearance for computer vision.* Wolfson Image Anal. Unit, Univ. Manchester, U.K., Tech. Rep., 1999.

[11] T.F. Cootes and C.J. Taylor, *Locating Faces Using Statistical Feature Detectors,* Proc. Second Intl Conf. Automatic Face and Gesture Recognition, pp. 204-209, 1996.

[12] T. F. Cootes, C. J. Taylor, A. Lanitis, *Active shape models: Evaluation of a multiresolution method for improving search,* In: Proc. British Machine Vision Conference. BMVA Press. 1994.

[13] I. Craw, H. Ellis, and J. Lishman, *Automatic extraction of face features,* Pattern Recognition Letters, vol.5, pp.183-187, 1987.

[14] O. Duchennel, F. Bach, I. Kweon, J. Ponce, *A Tensor-Based Algorithm for High-Order Graph Matching.* in Proceedings IEEE Conf. on Computer Vision and Pattern Recognition, 2009, pp. 1980V1987.

[15] Y. Dai, Y. Nakano, *Face-texture model based on SGLD and its application in face detection in a color scene*, Pattern Recognition, vol.29, no.6, pp.1007-1017, 1996.

[16] I. L. Dryden, K. V. Mardia, *Statistical Shape Analysis.* John Wiley Sons, 1998.

[17] J. C. Gower, *Generalized Procrustes analysis.* Psychometrika 40, 33V51,1975.

[18] C. Goodall, *Procrustes methods in the statistical analysis of shape.* J. Roy. Stat.Soc. B 53 (2), 285V339, 1991.

[19] X. Gao, Y. Su, X. Li, D. Tao, *A Review of Active Appearance Models*, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICSXPART C: APPLICATIONS AND REVIEWS, VOL. 40, NO. 2, MARCH 2010.

[20] A. Hill, T. F. Cootes, C. J. Taylor, *Active shape models and the shape approximation problem.* Image and Vision Computing, 14(8):601V607, Aug. 1996.

[21] Haralick, R. M., K. Shanmugam, and I. Dinstein, *Texture features for image classification,* IEEE Trans. Systems, Man, and Cybernetics, vol.3, no.6, pp.610-621, 1973.

[22] I. T. Jolliffe, *Principal Component Analysis.* second ed. Springer 2002.

[23] M. J. Jones, J. M. Rehg, *Statistical color models with application to skin detection*, in Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol.1, Fort Collins, Colorado, Jun.23-25, 1999, pp.274-280.

[24] M. Kass, A. Witkin, D. Terzopoulos, *Snakes: Active Contour Models.* First International Conference on Computer Vision 259-268, 1987.

[25] D. G. Kendall, *A survey of the statistical theory of shape.* Stat. Sci. 4 (2), 87V120, 1989.

[26] M. Kaiser, D. Arsic, S. Sural, G. Rigoll, *Tracking of facial feature points by combining singular tracking results with a 3d active shape model.* in VISAPP, 2010.

[27] M. Kaiser, D. Arsic, S. Sural, G. Rigoll, *Robust tracking of facial feature points with 3D active shape models.* ICIP pp.3829 3832 2010.

[28] T. Kohonen, *Self-Organization and Associative Memory,* Springer-Verlag, New York, 1989.

[29] R. Kjeldsen, J. Kender, *Finding skin in color images*, in Proc. 2nd Intl Conf. Automatic Face and Gesture Recognition, Killington, VT, Oct.14-16, 1996, pp.312-317.

[30] T. K. Leung, M. C. Burl, P. Perona *Finding faces in cluttered scenes using random labeled graph matching*, in Proc. Fifth IEEE Intl Conf. Computer Vision, Cambridge, Massachusetts, Jun.20-23, pp.637-644.

[31] K.Lam, H. Yan, *Fast algorithm for locating head boundaries,* J. Electronic Imaging, vol.3, no.4, pp.351-359, 1994.

[32] A. Lanitis, C. J. Taylor, and T. F. Cootes, *An automatic face identification system using flexible appearance models,* Image and Vision Computing, vol.13, no.5, pp.393-401, 1995.

[33] D. Lowe, G. David , *Object recognition from local scale-invariant features,* Proceedings of the International Conference on Computer Vision. 2. pp. 1150V1157, 1999.

[34] J. M. Morel, G. Yu, *ASIFT-A New Framework for Fully Affine Invariant Image Comparison.* SIAM Journal on Imaging Sciences, vol. 2, 2009, pp. 438V469.

[35] E. Osuna, R. Freund, F. Girosi,*Training support vector machines: an application to face detection*, in Proc. IEEE Conf. Computer Vision and Pattern Recognition, San Juan, Puerto Rico, Jun.17-19, 1997, pp.130-136.

[36] Y. G. Papagrigorakis, *Real Time Face Tracking Using Active Contour Models.* MSIN 2004.

[37] C. Rao, *Linear Statistical Inference and Its Applications.* New York: Wiley, 1973.

[38] A. Ross, *Procrustes Analysis.* Department of Computer Science and Engineering, University of South Carolina, 2004.

[39] H. Rowley, S. Baluja, T. Kanade,*Neural network-based face detection*,IEEE Trans. Pattern Analysis and Machine Intelligence, vol.20, no.1, pp.23-38,Jan. 1998.

[40] Sakai, T., M. Nagao, and S. Fujibayashi, *Line extraction and pattern detection in a photograph,* Pattern Recognition, vol.1, pp.233-248, 1969.

[41] M. B. Stegmann, D. D. Gomez, *A Brief Introduction to Statistical Shape Analysis.* Tech. Rep., Infomatics and Mathematical Modeling.

[42] K. K. Sung, T. Poggio,*Example-based learning for view-based human face detection*,IEEE Trans. Pattern Analysis and Machine Intelligence, vol.20, no.1, pp.39-51, Jan. 1998.

[43] K. K. Sung, *Learning and Example Selection for Object and Pattern Detection,* Ph.D. dissertation, Department of Brain and Cognitive Sciences, Massachusetts Inst. of Technology, MA, 1996.

[44] P. Sozou, T. F. Cootes, C. J. Taylor, E. DiMauro, *A nonlinear generalization of point distribution models using polynomial regression,* Image Vis. Computing, vol. 13, no. 5, pp. 451V457, 1995.

[45] P. P. Smyth, C. J. Taylor, J. E. Adams, *Automatic measurement of vertebral shape using active shape models,* in Proc. Br. Machine Vision Conf., 1996, pp. 705V714.

[46] M. Turk, A. Pentland,*Eigenfaces for recognition*, J. Cognitive Neuroscience, vol.3, no.1, pp.71-86, 1991.

[47] M. H. Yang, , D. Kriegman, N. Ahuja, *Detecting faces in images: a survey.* IEEE Trans. on Pattern Analysis and Machine Intelligence, vo1.24, no.1, pp.34-58, Jan., 2002.

[48] G. Yang, T. S. Huang, *Human face detection in complex background*, Pattern Recognition, vol.27, no.1, pp.53-63, 1994.

[49] J. Yang, A. Waibel *A real-time face tracker*, in Proc. Third Workshop Applications of Computer Vision, Sarasota, FL, Dec.2-4, 1996, pp.142-147.

[50] C. Ying, F. De la Torre, *Active condition models.* IEEE Conference on Automatic Face and Gestures Recognition (FG) 2011.

[51] K. C. Yow, R. Cipolla, *Feature-based human face detection,* Image and Vision Computing, vol.15, no.9, pp.713-735, 1997.

[52] A.Yuille, P. Hallinan, D. Cohen, *Feature extraction from faces using deformable templates,* Intl J. Computer Vision, vol.8, no.2, pp.99-111, 1992.