To Appear in the *IEEE International Conference on Computer Vision and Pattern Recognition*, Wisconsin, June 2003

# Bayesian Tangent Shape Model: Estimating Shape and Pose Parameters via Bayesian Inference*

Yi Zhou**
*Microsoft Research Asia*
*SMS, Peking Univ.*
*yizhou@msrchina.research.mic-rosoft.com*

Lie Gu
*Computer Science Department*
*Carnegie Mellon University*
*gu+@cs.cmu.edu*

Hong-Jiang Zhang
*Microsoft Research Asia*
*hjzhang@microsoft.com*

## Abstract

*In this paper we study the problem of shape analysis and its application in locating facial feature points on frontal faces. We propose a Bayesian inference solution based on tangent shape approximation called Bayesian Tangent Shape Model (BTSM). Similarity transform coefficients and the shape parameters in BTSM are determined through MAP estimation. Tangent shape vector is treated as the hidden state of the model, and accordingly, an EM based searching algorithm is proposed to implement the MAP procedure. The major results of our algorithm are: 1) tangent shape is updated by a weighted average of two shape vectors, the projection of the observed shape onto tangent space, and the reconstruction of shape parameters. 2) Shape parameters are regularized by multiplying a ratio of the noise variations, which is a continuous function instead of a truncated function. We discussed the advantages conveyed by these results, and demonstrate the accuracy and the stability of the algorithm by extensive experiments.*

## 1. Introduction

The geometrical description of an object can be decomposed into two parts: the geometrical transform and the shape. A common vision task is to recover both pose parameters and low-dimensional representations of the underlying shape from observed images. This procedure is usually referred as "shape analysis" or "shape registration".

Shape analysis has been advanced in both the literature of statistics and vision. The statistical theory of general shape space began with the work of Kendall [6] in 1977. Kendall described shape distribution in a Riemann manifold which is highly curved and nonlinear. Statistical techniques were first introduced to analyze the probabilistic distribution of shape in this manifold. Subsequent developments [7][5][2] have led to several practical statistical approaches to analyzing objects using probability distributions of shape and likelihood based inference. A comprehensive survey can be found in Small [5]. General shape space has been proved to be highly nonlinear. However, as for a set of concentrated data, tangent space provides a good linear approximation to general shape space. More importantly, modeling shape in tangent space can convert statistical shape analysis to standard multivariate analysis [7].

In image analysis literatures, practical parametric deformable models [3][10][11] have been developed to deal with the problems like segmentation or feature points localization. These models are generally capable of incorporating prior knowledge with observations directly derived from image data. In particular, *Active Shape Model* [3] proposed by Cootes et.al. in 1992 attracts a wide range of attention. ASM consists of a point distribution model capturing shape variations of valid object instances, and a set of grey gradient distribution models, which describe local texture of each landmark point. Cootes developed an iterative searching algorithm to actively update the model parameters according to the observed image. The major advantage of ASM is that the model can only deform in the ways leant from the training set. That is, it can accommodate considerable variability and it is still specific to the class of object it intends to represent. Specifically, in ASM the principle component analysis (PCA) technique is used to model both 2D shape variations and local grey level structures.

In this paper, we address the problem of shape analysis from two aspects. First, shape analysis problem is formulated in Bayesian framework. Specifically, we describe the prior model of tangent shape vectors, the likelihood model and the posterior of model parameters. Second, an EM based searching algorithm is given to estimate tangent shape and other model parameters. The

---

* The work presented in this paper is performed in Microsoft Research Asia and Carnegie Mellon University.

** This author is partially supported by NSFC grant #10171005.

derived updating rules highlight the advantages of BTSM shape registration

The rest of the paper is organized as follows: we begin with the description of tangent space approximation and the probabilistic formulation of shape registration. We describe the parameter estimation algorithm and compare the updating rules of ASM searching and BTSM searching in Section 3. Section 4 provides experimental results. We discuss some related problems and draw the conclusions and in Section 5 and 6.

## 2. A Bayesian Formulation to Shape Registration

The probabilistic formulation of shape registration problem contains two models: one denotes the prior shape distribution in tangent shape space and the other is a likelihood model in image shape space. Based on these two models we derive the posterior distribution of model parameters. The MAP estimation of the parameters can be obtained using the EM algorithm.

### 2.1 Tangent Space Approximation

Assuming that a planar shape is described by *N* landmark points in the image, we can represent it by a *2N*-dimensional vector $s_i$. The difference between two planar shapes is usually measured by their Procrustes distance [12]. Furthermore, given a set of training shape vectors $\{s_i\}_{i=1}^{L}$, the most popular way to align them into a common co-ordinate frame is Generalized Procrustes Analysis (GPA) [1]. The procedure essentially equals to minimize a quadratic loss function defined by $L(\mu)=\sum_{all\ i}\| T_i(s_i)-\mu \|^2$, where $T_i(s_i)$ is a 2D similarity transform of $s_i$. See [12] for the details of GPA.

The tangent space is a linear approximation of the general shape space in the vicinity of the mean shape vector. More specifically, the tangent space $\mathbb{C}_\mu$ is defined as the space normal to $T(\mu)$ and passing through $\mu$. The Euclidean distance in the tangent space is a good approximation to the Procrustes distance, if most of shape instances are close. $s_i$ can be transformed onto $\mathbb{C}_\mu$ by aligning $s_i$ with $\mu$ as $\{x_i \in \mathbb{C}_\mu : x_i = T_i(s_i)\}_{i=1}^{L}$. $x_i$ is often referred as "tangent shape vector" and represented as a *2N*-dimensional vector. The residuals are computed as $\{t_i = x_i - \mu\}_{i=1}^{L}$ in tangent space instead of image space, to remove the difference introduced by similarity transform $T_i$.

Note that the dimension of $\mathbb{C}_\mu$ is $2N-4$, where the degenerated dimensionality is corresponding to the degree of freedom of similarity transformation in a *2d* Euclidean space. Furthermore, since any transformed shape vector from $\mu$ can be represented by a linear combination of $\{e, e^*, \mu, \mu^*\}$ [1], the complement space of $\mathbb{C}_\mu$ is spanned by $\{e, e^*, \mu, \mu^*\}$. Therefore, the covariance matrix of tangent shape,

$$Var(X)=\frac{1}{L-1}\sum_{i=1}^{L}(x_i-\mu)(x_i-\mu)^T \qquad (1)$$

will has at least four zero eigenvalues with corresponding eigenvectors $\{e, e^*, \mu, \mu^*\}$. In other words, the tangent shape variances in this complement space must be zero.

### 2.2. Prior Tangent Shape Model

We apply a probabilistic extension of traditional PCA to model tangent shape variation, which is similar to PPCA proposed by Tipping and Bishop [4]. The model can be written as

$$\Phi^T(x-\mu)=\begin{pmatrix} I_r \\ 0_{(2N-4-r)\times r} \end{pmatrix} b+\varepsilon \qquad (2)$$

a) $\Phi^T : (2N-4)\times 2N$ is the tangent projection matrix whose row vectors are the eigenvectors of $Var(X)$. $\Phi_r : 2N\times r$ consists the first *r* columns of $\Phi$.

b) *b*, the shape parameter, is a *r*-dimensional vector distributed as multivariate Gaussian $N(0, \Lambda)$, where $\Lambda = diag(\lambda_1,...,\lambda_r)$. $\lambda_i$ is the *ith* eigenvalue and *r* is the number of modes to retain in PCA.

c) $\varepsilon$ denotes an isotropic noise in the tangent space. It is a *2N-4*-dimensional random vector which is independent with *b* and distributes as $p(\varepsilon) \sim \exp\{-\| \varepsilon \|^2/(2\sigma^2)\}$ ($\sigma^2 = \frac{1}{2N-4}\sum_{i=r+1}^{2N-4}\lambda_i$).

After some simple algebra the model (2) can be rewritten as:

$$x = \mu + \Phi_r b + \Phi\varepsilon \qquad (3)$$

By adding an isotropic Gaussian noise term we associate PCA with a probabilistic explanation, thereby allowing to compute the posterior of model parameters.

Each item of *b* reflects a specific variation along the corresponding principle component (PC) axis. Instead of using all modes and *2N-4*-dimensional shape parameters, we only select a subset of them to reconstruct the shape with shape variations we concern about. The fewer the modes are used, the more compact the model will be, and the smoother the reconstructed shape tends to be. On the other hand, more modes are involved in describing shape,

---

[1] $e = (1,0,1,0,...,1,0)^T$; $x^*$ is obtained by rotating planar shape *x* by 90°, i.e. $x = (x_1, x_2 ... x_{2N})^T \Rightarrow x^* = (-x_2, x_1, ..., -x_{2N}, x_{2N-1})^T$

Figure 1: Shapes reconstructed by the first three PCs: in each row the middle one is the mean shape. Else are obtained by varying corresponding PC from $-3\sqrt{\lambda_i}$ to $3\sqrt{\lambda_i}$.

more flexible the model is. Shape variation along the first three PCs is visualized in Figure 1. The interpretation of PCs is not straight forward. A possible interpretation is that the first PC describes variations in vertical direction, the second PC may explain the variation on mouth, and the third PC may account for out-of-plane rotation.

The tangent space noise $\varepsilon$ can also be viewed as a compensation of missed shape variation during PCA projection. When the number of modes is larger, more variation is retained in PCA model and the noise variance $\sigma^2$ is smaller.

### 2.3. Adaptive Likelihood Model

To incorporate image evidence into the Bayesian framework one requires a likelihood $P(I \mid x, \theta)$ which is usually a probability distribution of the grey levels conditional on the underlying shape. However, directly parameterizing $P(I \mid x, \theta)$ may not be a good idea, because $I$ and $X$ are not in a same physical coordinate system, and the parametric form of $P(I \mid x, \theta)$ is usually complex and nonlinear. In BTSM, we redefine the likelihood as $P(y \mid x, \theta)$. Assume $y^{old}$ is the shape estimated in the last iteration, by updating each landmarks of $y^{old}$ with its local texture we obtain *y*, which is called "observed shape vector". The distance between observed shape *y* and the true shape can also be model as an adaptive Gaussian as (4). By adaptive we mean the variance of the model is determined by the distance between *y* and $y^{old}$ in each iteration step.

$$y = sU_\theta x + c + \eta \tag{4}$$

a) *y*: observed shape vector,

b) *s*: scale parameter;

$U_\theta = I_N \otimes \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$: rotation matrix;

$c = 1_N \otimes \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$: translation parameter.

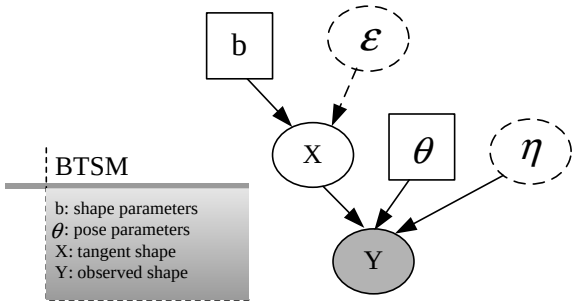


Figure 2. A graphical illustration of Bayesian tangent shape model: circles stand for variables, dashed circles denote noise terms and rectangles denote model parameters.

($\otimes$ denotes Kronecker product[2].)

c) $\eta$: isotropic observation noise in the image space. $\eta \sim N(0,\ \rho^2 I_{2N})$. $\rho$ is set by $\rho^2 = c \| y^{old} - y \|^2$, where *c* is a manually chosen constant.

### 2.4. Posterior

Now we can compute the posterior of model parameters $(b, s, c, \theta)$ given the observed shape vector $y$. By applying Bayes rule we have derived the equation (5). Directly optimizing the posterior is difficult. Alternatively, if the tangent shape $x$ is known, the posterior of model parameters conditional on both $x$ and $y$ are much simpler. This leads us to implement the EM based parameters estimation algorithm.

$$p(b, c, s, \theta \mid y) \tag{5}$$
$$\propto \exp\{-\frac{1}{2}[(\sigma^2 + s^{-2}\rho^2)^{-1}(\| \Phi_r^T T_\theta^{-1}(y) - b \|^2 + \| \Phi_{-r}^T T_\theta^{-1}(y) \|^2)$$
$$+ s^2\rho^{-2} \| A^T T_\theta^{-1}(y) \|^2 + b^T \Lambda^{-1} b]\} \cdot \frac{const}{(\sigma^2 + s^{-2}\rho^2)^{(N-2)} s^{-4} \rho^4}$$

where the *const* do not vary with $(b, c, s, \theta)$ and $\Phi_{-r}$ is the sub-matrix of $\Phi$ by removing the first *r* columns. The derivation is left to the appendix A.

### 2.5. BTSM as A Hidden Variable Model

A graphical illustration of BTSM is shown in Figure 2. The tangent shape $x$ is the hidden variable and $y$ is observation. The prior shape model and the likelihood model are connected through tangent shape.

## 3. Parameter Estimation in BTSM Searching

In this section, we describe an EM algorithm for estimation the MAP parameters of BTSM model. Before

[2] $A_{m\times n} \otimes B_{p\times q} = (a_{ij}B)_{i,j} : mp \times nq$

immersing ourselves in the details of derivation, however, let us first present the results of EM parameter estimation and compare them with those of ASM.

### 3.1. Comparison between BTSM and ASM

The iterative updating procedure of ASM is shown in Figure 3. In ASM, tangent shape $x$ is directly constructed from shape parameter $b$, where $b$ is a truncation of $T_\theta^{-1}(y)$'s coordinates within the range of $(-3\sqrt{diag(\Lambda)}, 3\sqrt{diag(\Lambda)})$. In BTSM we derive the updating equations of $x$ and $b$ shown in Figure 4. (See Section 3.2 and 3.3 for the details of derivation.) The major difference of the two algorithms comes from their updating rules of the tangent shape $x$ and shape parameter $b$.

a) In BTSM, the tangent shape $x$ is updated by a weighted average of the shape reconstructed from the shape parameter $b$ and the tangent projection of the observed shape $y$. In this way, the estimation of $x$ encodes both prior shape knowledge and image evidence. It is interesting to note that the weight $p$ is automatically chosen by computing the ratio between the variance $\sigma$ of prior noise in tangent space and the variance $\rho$ of the observation noise. They are aligned to the same scale by multiplying the scale factor $s$ of similarity transform. When $\rho$ is large, which implies the image is noisy or the observation is not stable, shape parameters are more important for updating $x$. On the other hand, when $\rho$ is small, the shape estimation may be converged already, we need not to regularize it too strictly.

b) Regularization on shape parameters is required to generate valid shape instances. Using a continuous regularization function often is preferred to using a truncation function because numerically, discontinuous regularization on $b$ may result in a unstable estimation. That is, the result may shift back and forth instead of converging to a point. In BTSM, the shape parameter is constrained by multiplying a contrained factor $R = \Lambda(\Lambda + \sigma^2 I)^{-1}$. Remember that $\Lambda$ represents prior shape variance matrix and $\sigma^2$ represents the resudial variance. (See Section 2.2 for details). Specifially, along the *ith* principle axis, $b_i$ is updated by $b_i = \lambda_i / (\lambda_i + \sigma^2)(\Phi_{\cdot i}^T x)$, where $\Phi_{\cdot i}$ is the *ith* column of $\Phi$.

In short BTSM algorithm enjoys its merits in two aspects: weighted representation of tangent shape and continuous regularization of shape parameters. These results are derived from optimizing an explicit and continuous loss function using EM.

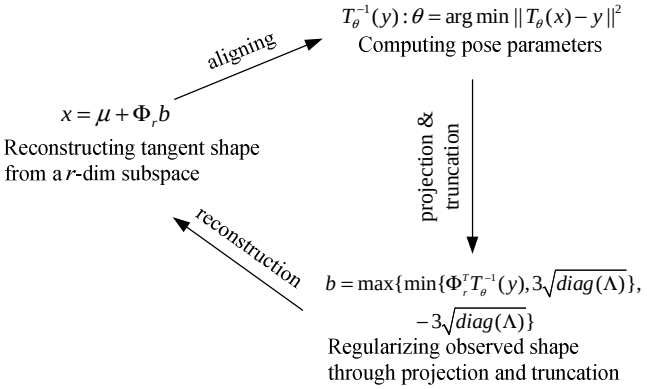


Figure 3. Updating rules of Active Shape Model

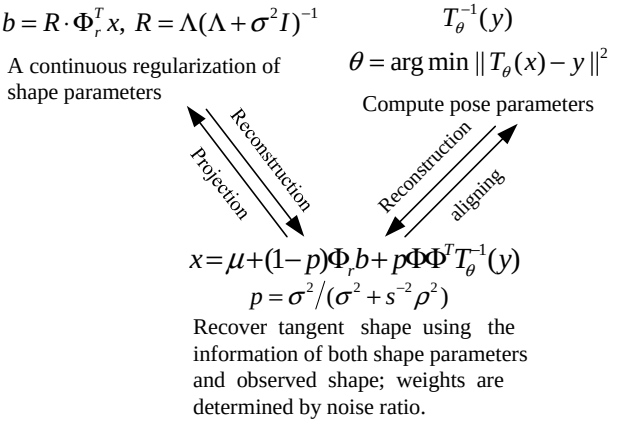


Figure 4. BTSM updating rules: Tangent shape $x$ is estimated by a weighted sum of the shape reconstructed from shape parameters $b$ and the transform of the observed shape $y$ to the tangent space.

### 3.2. Expectation Step

Given a set of complete data $\{x, y\}$, the complete posterior of model parameters is simply a product of the following two distributions,

$$p(b \mid x) \propto \exp\{-1/2[b^T \Lambda^{-1} b + \sigma^{-2} \| x - \mu - \Phi_r b \|^2]\} \quad (6)$$

$$p(\gamma \mid x, y) \propto \exp\{-1/2[\rho^{-2} \| y - X\gamma \|^2]\} \quad (7)$$

where $X = (x, x^*, e, e^*)$ and $\gamma = (s \cdot \cos\theta, s \cdot \sin\theta, c_1, c_2)^T$. Taking the logarithm and the conditional expectation, we obtain:

$$\begin{aligned} & Q(\gamma \mid \gamma_{old}) \quad (8) \\ &= \langle \log p(b, c, s, \theta \mid x, y) \rangle = \langle \log p(b \mid x) + \log p(\gamma \mid x, y) \rangle \\ &= -\frac{1}{2}\left[ b^T \Lambda^{-1} b + \sigma^{-2} \langle \| x - \mu - \Phi_r b \|^2 \rangle + \rho^{-2} \langle \| y - X\gamma \|^2 \rangle \right] \\ &\quad + const \end{aligned}$$

Computing the Q-function of (8) essentially equals to calculate two statistics, the conditional expectations of $x$ and $\| x \|^2$ with respect to $p(x \mid y, c, s, \theta)$

$$\langle x \rangle = \mu + (1-p)\Phi_r b + p\Phi\Phi^T T_\theta^{-1}(y) \quad (9)$$

$$\langle \| x \|^2 \rangle = \| \langle x \rangle \|^2 + (2N-4)\delta^2 \quad (10)$$

where $p = \sigma^2 / (\sigma^2 + s^{-2}\rho^2)$ and $\delta^2 = (\sigma^{-2} + s^2\rho^{-2})^{-1}$. The detailed derivation is left to the appendix B.

### 3.3. Maximization Step

The M step maximizes the Q-function over model parameters. Since the terms depending on $b$ and $\gamma$ are decoupled in (8), it is a much simpler expression to maximize than the logarithm of the posterior in (5). We use " ~ " to denote the updated parameters. By computing the derivative of the Q-function we have,

$$\tilde{b} = \Lambda(\Lambda + \sigma^2)^{-1}\Phi_r^T(\langle x \rangle - \mu) = \Lambda(\Lambda + \sigma^2)^{-1}\Phi_r^T \langle x \rangle \quad (11)$$

$$\tilde{\gamma} = \left(\frac{y^T \langle x \rangle}{\langle \| x \|^2 \rangle}, \frac{y^T \langle x \rangle^*}{\langle \| x \|^2 \rangle}, \frac{1}{N}\sum_{i=1}^{N} y_{1i}, \frac{1}{N}\sum_{i=1}^{N} y_{2i}\right) \quad (12)$$

Accordingly, the updating equations of each pose parameter are,

$$\tilde{s} = \sqrt{\tilde{\gamma}_1^2 + \tilde{\gamma}_2^2}\,, \tilde{\theta} = atan(\tilde{\gamma}_1 / \tilde{\gamma}_2), \text{ and } \tilde{c} = (\gamma_3, \gamma_4)^T \quad (13)$$

### 3.4. Inhomogeneous Observation Noise

In Section 2.3 we assume the observation noise is distributed as an isotropic Gaussian. This assumption may not always hold, because the noise of each feature landmark may be different due to partial occlusion, noisy background or other effects in the image. We can choose a diagonal variance matrix instead for the observation noise $\eta$ as,

$$\eta \sim N(0,\ \Sigma), \Sigma = diag(\rho_1^2, ..., \rho_N^2) \otimes I_2 \quad (14)$$

where $\rho_i^2 = c((y_{2i-1}^{old} - y_{2i-1})^2 + (y_{2i}^{old} - y_{2i})^2)$ EM algorithm can also be applied to this case with slight modification. Instead of computing $\langle x \rangle$ and $\langle x^2 \rangle$, the statistics we need to compute in the E-Step is $\langle x \rangle$ and $\langle x^T \Sigma^{-1} x \rangle$. The results of EM parameters estimation are given in the appendix C.

## 4. Experimental Results

In this section we compare BTSM with ASM and demonstrate BTSM searching improves both accuracy and stability.

Figure 5. An example of BTSM searching: (Left) Initial shape mask, we perturb its orientation and scale parameter to make the task more difficult. (Middle) Searching result after 10 iterations at the top layer. (Right) Final result by searching all three layers.

### 4.1. BTSM Searching

Similar to ASM the BTSM searching algorithm is decomposed into two major steps: local texture matching and EM inference for shape and transformation parameters. As usual, the searching is run in a multi-resolution framework. A three level Gaussian image pyramid is formed on a testing image by repeated sub-sampling. Model instance starts at the $1/4$ resolution of the image. Different dimensions $r$ of shape parameter vector are used for different pyramid layers. We choose $r = 5$ for the first layer, $r = 20$ for the second layer, and $r = 40$ for the third layer. Figure 5 shows a typical example of BTSM searching. More searching results of ASM and BTSM are shown in Figure 10.

### 4.2. Accuracy

To compare the accuracy of the two algorithms quantitatively we divide our database into two parts, one used for training and the other used for testing. Our database contains 870 grey-scale images in the FERET database [8], the AR database [9] and other collections. Each image contains a face with a size ranging from $150 \times 150$ to $220 \times 220$, and with different facial expressions and different illumination conditions. A total of 83 face landmarks are labeled manually on each image of the training set.

We train both the ASM model and our model on 599 faces, and use the else 271 images for testing. For each testing image, an initial guess of face location are provided by a Boosting based face detector and then, the mean face shape mask is transformed and putted on the detected region. We perturb the shape mask by randomly rotating (from 0° to 45°) and scaling (from 1 to 1.2). The perturbed shape is used as initial values and is fed into the two algorithms. The searching processes would not stop unless the results are converged or the number of iterations is over than 100.

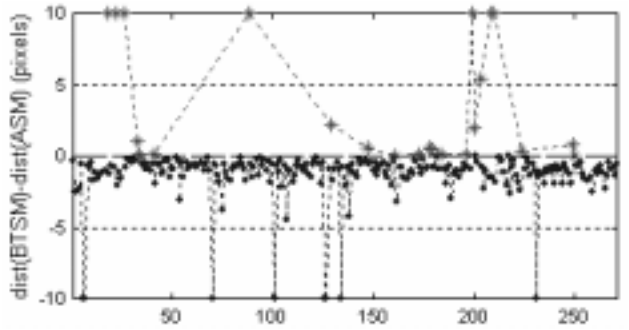


Figure 6. Comparison of the accuracy of BTSM and ASM: x-axis demotes the index of test images and y-axis denotes the difference of the estimation errors $dist(BTSM)_j - dist(ASM)_j$ between ASM and BTSM. Points below $y = 0$ (blue points) denote images with better performance by BTSM and red points are opposite. For a total of 271 testing images, 248 of them are marked blue and 23 of them are marked red, which means on 91.51% testing images the searching results of BTSM are better than that of ASM.

Figure 7. An ambiguous searching result: an unstable algorithm does not guarantee that the model converges to similar results while searching in similar images. The figure shows the searching results on three contiguous frames with a slightly change in view. Notice the inconsistent searching results on the nose and the chin of the boy.

To compare the accuracy of the two algorithms, we compute the estimation error by a difference measure defined by the sum of the distance between searched landmark and annotated landmark.

$$dist(A)_j = \sum_{i=1}^{N} \sqrt{(x_i^A - x_i)^2 + (y_i^A - y_i)^2}$$

$dist(A)_j$ denotes estimation error of algorithm A on the image $j$, where $(x_i, y_i)$ is annotated coordinates of the *ith* landmark and $(x_i^A, y_i^A)$ is the searched coordinates of the *ith* landmark by algorithm A. We have plotted $j \sim dist(BTSM)_j - dist(ASM)_j$ in Figure 6. It is shown that on 248 of 271 (91.51%) images, the search results of BTSM are better than that of ASM.

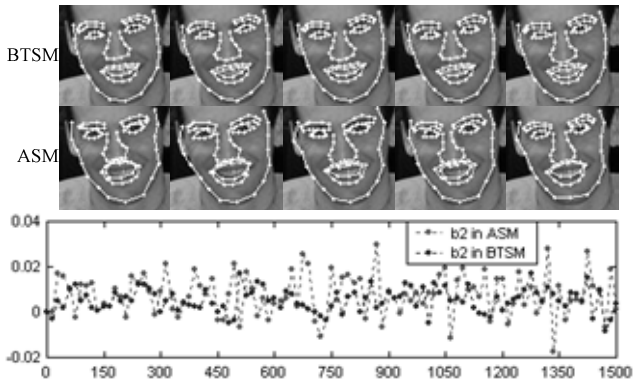


Figure 8. Comparison variation of estimation results in one individual dimension of shape parameter *b*: (Top) Five intermediate results of ASM searching and BTSM searching. (Bottom) The evolution of the shape parameter *b[2]* with the increasing of iterations number. Red points denote *b[2]* produced by the truncation procedure of ASM. Blue points are *b[2]* estimated by BTSM algorithm.

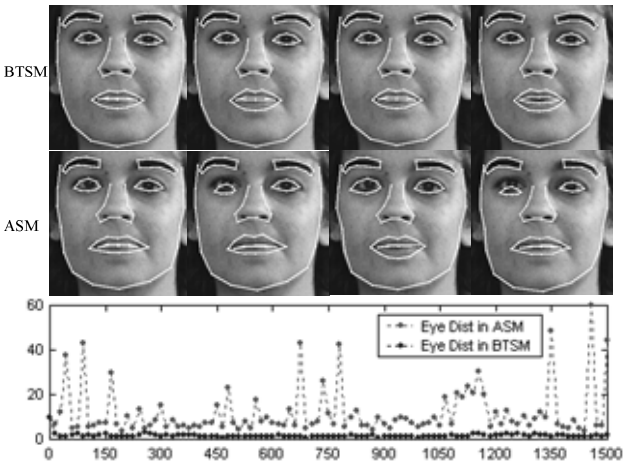


Figure 9. Comparison of variations of the alignment errors on eye points: (Top) Four intermediate results of ASM searching and BTSM searching. (Bottom) the evolution of eye errores of ASM and BTSM.

### 4.3. Stability

Another character of shape analysis algorithms we concern is numerical stability of estimation results. For a robust searching algorithm we expect that variation of estimation results decreases with the increasing of iteration number $i$. An unstable algorithm will produce ambiguous results. See Figure 7 for an example.

We explore the stability of ASM and our BTSM algorithm in two ways. The first is the variation of estimation results in one individual dimension of shape parameter $b$, and the next is the variation of some facial component. Figure 8 compares the variations in the estimation of the second shape parameter $b[2]$. Figure 9 compares the variations of the estimation errors on eyes.

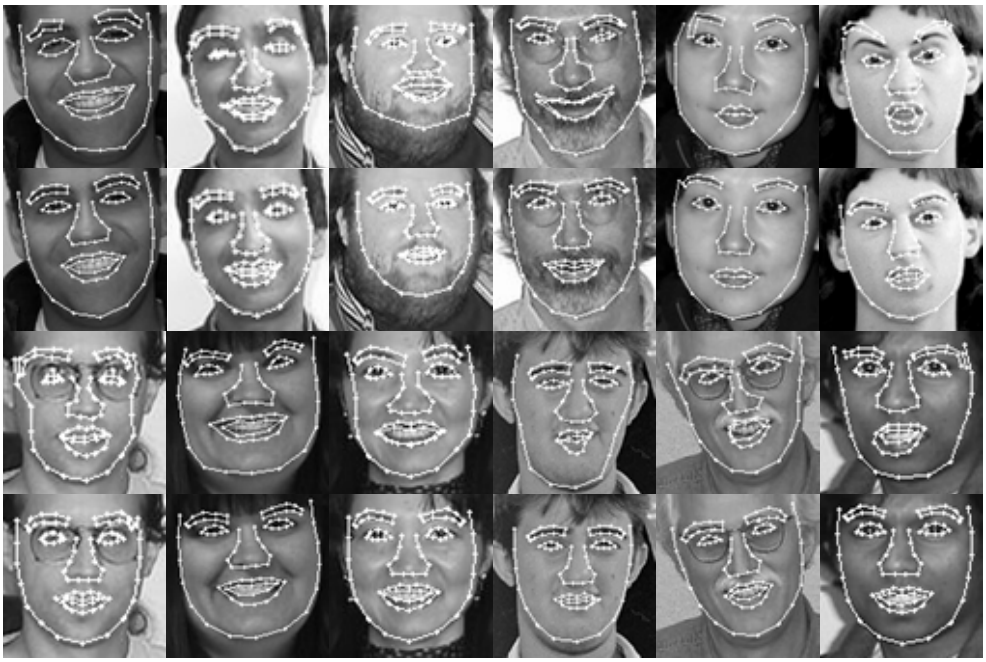

Figure 10. Comparison of BTSM and ASM searching results: (First and Third Rows) results of ASM searching; (Second and Fourth Rows) results of BTSM searching

We have plotted the value of $b[2]$ and $dist(eye)$ for every 15 steps of iterations. From the figures we can observe that the variations of the estimation results by BTSM algorithm are much smaller.

## 5. Discussion

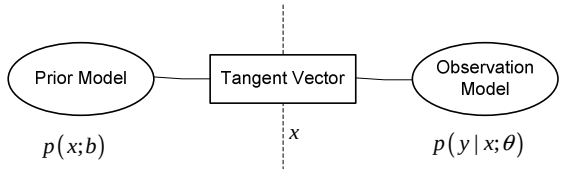


Figure 11. A generalization of BTSM model

BTSM can be extended to a more general form illustrated by the undirected graph in Figure 11. The prior model describes shape variation, the observation model incorporates image evidence and they are connected through the tangent shape. While the tangent shape is estimated, due to its local Markov property, the MAP estimation of pose parameters depends only on the right side of the graph, and it is degenerated to standard Procrustes Analysis with the assumption that the observation noise is an isotropic Gaussian. Note that the equation (25) equals to estimate pose parameters using weighted Procrustes analysis. Similarly the MAP estimation of shape parameters is completely determined by the left part of the graph given the tangent shape. Figure 11 provides a general framework for shape analysis problem. In contrast to directly optimizing a huge, heuristically defined loss function, the statistical treatment in BTSM provides the flexibility to deal with different problems in different sub-models. For example, if we are interested in modeling multimodal shape variations like exaggerated face expression, we may parameterize the left part as a Gaussian mixtures; if we are interested in handling partial occlusion or image noise, we may implement the right part using robust statistics methods. Approximate inference algorithm may need to be adopted in both cases.

BTSM shape registration runs very fast since we derive analytical solution in EM parameter estimation. In E-step, computing the expectation of the two statistics (please refer to (9) and (10)) includes only three matrix multiplication. In the M-step, parameters updating rules (equations (11) and (12)) involves only one matrix multiplication and some inner products of vectors. In our experiments, it takes about 200 ms in general for BTSM to converge on a 300x300 face on a Pentium3-800Hz machine; and for smaller faces it takes less time (from 60ms to 200ms, depends on the size of the face).

## 6. Conclusion

This paper presents a Bayesian approach for shape registration problem. By projecting shape to tangent shape, we have built the models describing the prior distribution of face shapes and their likelihood. We have developed the BTSM algorithm to uncover the shape parameters and transformation parameters of an arbitrary figure. We have

compared our algorithm with the classic ASM algorithm and demonstrated its accuracy and efficiency.

## 7. References


[1] Colin Goodall, "Procrustes Methods in the Statistical Analysis of Shape", *J. R. Statist.Soc. B*, 1991, pp. 285-339.

[2] J. T. Kent and K.V. Mardia, "Shape, Procrustes tangent projection and bilateral symmetry", *Biometrica*, 2001, pp. 469-485.

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

[4] M. Tipping and C. Bishop. "Probabilistic principal component analysis" Technical Report Technical Report NCRG/97/010, Neural Computing Research Group, Aston University, Birmingham, UK, September 1997.

[5] C. Small, 'The Statistical Theory of Shape', Springer, 1996.

[6] Kendall, D. G., 'The Diffusion of Shape', *Adv. Appl. Probab.*, 1977, pp. 428-430.

[7] Ian L Dryden and Kanti V. Mardia, ` Statistical Shape Analysis ', published by John Wiley and Sons, Chichester in July, 1998.

[8] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. The FERET evaluation methodology for face recognition algorithms. *IEEE Trans. on PAMI*, 22(10):1090--1104, 2000.

[9] A.M. Martinez and R. Benavente. The AR Face Database. *CVC Technical Report* #24, June 1998.

[10] A. Blake, M. Isard, Active Contours, Springer, Berlin, 1998.

[11] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321--331, 1987.

[12] Gower, J. C. Generalized Procrustes analysis. Psychometrika, 40, 33-5, 1975.


## Appendix

### A. Posterior Distribution of Parameters

By combining (3) with (4) and multiplying $\Phi_r^T$ on both sides of the equation we have,

$$\begin{aligned} T_\theta^{-1}(y) &= \mu + \Phi_r b + \Phi\varepsilon + s^{-1}U_\theta^{-1}\eta \qquad (\xi \triangleq s^{-1}U_\theta^{-1}\eta) \\ &= \mu + \Phi_r b + \Phi(\varepsilon + \Phi^T\xi) + AA^T\xi \\ \Rightarrow \Phi_r^T T_\theta^{-1}(y) - b &= (I_r, 0_{r\times(2N-4-r)})(\varepsilon + \Phi^T\xi) \end{aligned} \tag{15}$$

where $A = (e, e^*, \mu, \mu^*)$ . Since $\varepsilon$ and $\xi$ are independent, the distributions of $\varepsilon + \Phi^T\xi$ and $A^T\xi$ can be computed as,

$$\begin{aligned} \left(\varepsilon + \Phi^T\xi\right) &\sim N(0,\ (\sigma^2 + s^{-2}\rho^2)I_{2N-4}) \\ A^T\xi &\sim N(0,\ s^{-2}\rho^2 I_4) \end{aligned} \tag{16}$$

Combining (15) and (16) we obtain the likelihood of model parameters. The posterior of model parameters is computed by applying the Bayes rule as (5).

### B. Detailed Derivation of Expectation Step

The conditional probability of the tangent shape vector $x$ given the observed shape $y$ and model parameters is

$$p(x \mid y, c, s, \theta) \tag{17}$$

$$= \begin{cases} \propto \exp\{-\frac{1}{2}[\sigma^{-2} \parallel x - \mu - \Phi_r b \parallel^2 \\ \quad + s^2\rho^{-2} \parallel x - T_\theta^{-1}(y) \parallel^2]\} & when\ A^T(x-\mu) = 0 \\ 0, & otherwise \end{cases}$$

The tangent shape $x$ can be written as

$$x = (AA^T + \Phi\Phi^T)x = AA^T\mu + \Phi\Phi^T x = \mu + \Phi\Phi^T x \tag{18}$$

where $A = (e, e^*, \mu, \mu^*)$ . Since $x$ is an isotropic Gaussian, the elements of $x$ on the two orthogonal subspaces are independent, i.e. $A^T x \perp \Phi^T x$ . So

$$\begin{aligned} p(\Phi^T x \mid A^T(x-\mu) = 0) &= p(\Phi^T x) \\ &= N((1-p)\Phi^T\mu_1 + p\Phi^T\mu_2,\ \delta^2 I_{2N-4}) \end{aligned} \tag{19}$$

where $\mu_1 = \mu + \Phi_r b$, $\mu_2 = T_\theta^{-1}(y)$, $p = \sigma^2/(\sigma^2 + s^{-2}\rho^2)$ and $\delta^2 = (\sigma^{-2} + s^2\rho^{-2})^{-1}$ . Therefore the conditional expectation of $x$ is

$$\begin{aligned} \langle x \rangle &= AA^T\mu + \Phi E(\Phi^T x) \\ &= (1-p)\Phi\Phi^T\mu_1 + p\Phi\Phi^T\mu_2 + AA^T\mu \\ &= \mu + (1-p)\Phi_r b + p\Phi\Phi^T T_\theta^{-1}(y) \end{aligned} \tag{20}$$

and the conditional expectation of the norm of $x$ is

$$\langle x^2 \rangle = \parallel A^T\mu \parallel^2 + E \parallel \Phi^T x \parallel^2 = \left\| \langle x \rangle \right\|^2 + (2N-4)\delta^2 \tag{21}$$

### C. EM for Inhomogeneous Observation Noise

We ignore the details of the derivation and just present the results of E-step and M-Step. Let us denote

$$\Delta \triangleq (\sigma^{-2} + s^2\Sigma^{-1})^{-1}$$

$$P \triangleq (I + \sigma^2 s^2 \Sigma^{-1})^{-1}$$

$$\alpha \triangleq \Delta^{-1/2}[P(\mu + \Phi_r b) + (I - P)T_\theta^{-1}(y)]$$

Let $B \triangleq Orth(\Delta^{1/2}A)$ , whose column vectors form an orthogonal basis of the column space of $\Delta^{1/2}A$ .

The E-step:

$$\langle x \rangle = \Delta^{1/2}[\alpha + BB^T(\Delta^{-1/2}\mu - \alpha)] \tag{22}$$

$$\langle x^T\Sigma^{-1}x \rangle = \langle x \rangle^T \Sigma^{-1} \langle x \rangle + tr(\Delta\Sigma^{-1}) - tr(\Delta\Sigma^{-1}BB^T) \tag{23}$$

The M-step:

$$\tilde{b} = \Lambda(\Lambda + \sigma^2)^{-1}\Phi_r^T(\langle x \rangle - \mu) = \Lambda(\Lambda + \sigma^2)^{-1}\Phi_r^T\langle x \rangle \tag{24}$$

$$\tilde{\gamma} = \left( \frac{y^T\Sigma^{-1}\langle x \rangle}{\langle x^T\Sigma^{-1}x \rangle}, \frac{y^T\Sigma^{-1}\langle x \rangle^*}{\langle x^T\Sigma^{-1}x \rangle}, \frac{\sum_{i=1}^N \rho_i^{-2} y_{1i}}{\sum_{i=1}^N \rho_i^{-2}}, \frac{\sum_{i=1}^N \rho_i^{-2} y_{2i}}{\sum_{i=1}^N \rho_i^{-2}} \right)^T \tag{25}$$