A Novel Bayesian Shape Model for Facial Feature Extraction

Zhong Xue[1], Stan Z. Li[2], Dinggang Shen[1], Eam Khwang Teoh[3]

[1] University of Pennsylvania, Suite 380, 3600 Market ST, Philadelphia PA 19104
[2] Microsoft Research Asia, Sigma Center, Zhichun Road, Haidian District, Beijing 100080
[3] School of EEE, Nanyang Technological University, Singapore 639798
E-mail: zxue@rad.upenn.edu, szli@microsoft.com, dgshen@rad.upenn.edu, eekteoh@ntu.edu.sg

Abstract

This paper presents a novel application of the Bayesian Shape Model (BSM) for facial feature extraction. First, a full-face model is designed to describe the shape of a face, and the PCA is used to estimate the shape variance of the face model. Then, the BSM is applied to match and extract the face patch from input face images. Finally, using the face model, the extracted face patches are easily warped or normalized to a standard view. Applications of this facial feature extraction algorithm include face recognition, face video coding and retrieval, face animation and multi-media.

1 Introduction

Deformable models have been studied intensively during the last ten years [1, 2, 3, 4]. They have been demonstrated to be more effective in object matching and able to adapt themselves to fit objects more closely than the traditional rigid models. Among different types of the deformable models, the parametric models [1, 2, 5, 6, 7] are more suitable for matching complex object shapes because some prior information of the geometrical shape is incorporated. However, most existing parametric models encode the shape information in a "hard" manner that the prototype contour is fixed during the matching process. As a result, only small and local deformations can be tolerated. To remedy this shortcoming, a deformable model with the name of "EigenSnake" is proposed in the Bayesian framework [8], where the prototype contour can be adaptively adjusted in the process of object matching. Comparative studies based on face extraction experiments verified that the EigenSnake performs better than its fixed counterpart G-Snake [7].

In 3D object matching using 2D images, the object shapes are subject to projections such as affine transformations. However, most deformable models are not affine invariant, and their performance deteriorates when the algorithms are applied to matching affine transformed object shapes. AI-Snake deals with the affine-invariant active contour matching [9], but it uses fixed prototypes of object shapes.

Recently a Bayesian Shape Model was proposed, which formulates the matching of a deformable model to the object in the Bayesian framework [10]. The BSM incorporates the advantages of both the EigenSnake model and AI-Snake model. In BSM, the deformable contour used to match an object is modeled as the affine-transformed and deformed version of the prototype contour, and this prototype contour is dynamically deformable to adapt the shape variations using the information gathered from the matching process. The prior distribution of the prototype contour estimated via PCA [2, 11] which reflects the major shape deformations, is used to constrain the admissible shapes. In this way, large shape deformation due to the shape variations of samples can be tolerated. Moreover, an affine-invariant internal energy is defined to describe the shape deformations between the prototype contour in the shape domain and the deformable contour in the image domain. Good performance was observed by comparing the BSM model with the well known ASM model and the AI-Snake model.

In this paper, the BSM is applied to facial feature extraction. The following specific requirements of the new algorithm are considered: (1)It should be able to distinguish different expressions, *i.e.* global variations of the whole face. (2) The face model should be efficient for warping and normalization.

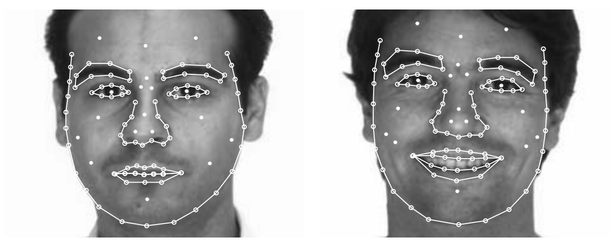

Figure 1: Two examples of the manually marked faces. 'o': 88 contour points, '*': 18 control points.

In our method, first a 2D full-face model is proposed to meet the requirements by referring to the *CANDIDE* face model [12] and the facial feature model in MPEG proposals. In this model, a number of points are recruited to represent the shape of a whole face. The full-face model consists of two sets: the contour points and the control points. The contour points are the landmark points of the face outline, eyebrows, eyes, nose and mouth; the control points represent the feature points that placed in the key position of the face, and every three of the control and contour points can be used to form a face unit triangular for facial image warping. Based on a number of manually marked training samples, the prior distribution of the full-face model is estimated by using the PCA. In addition, the individual shape model of each component contour is also built. Therefore, not only the separate shape variation of each component contour of the facial features, but also can the global shape variation be represented conveniently. We use 88 contour points and 18 control points to describe a face image, as shown in Fig.1.

Since the control points are not corresponding to some salient features such as edges of the image, their position is estimated from that of the contour points after the matching result is acquired. Therefore, given an input face image, the contour points of the face model are acquired by using the BSM matching framework, and the control points are estimated from the matching results of the BSM according to the prior distribution of the full-face model.

The face patch is then extracted from the input image, and further operations such as facial image warping and normalization, or facial expression cancellation, can be performed to the extracted face patch very easily by using the piece-wise affine algorithm.

2 Introduction of the Bayesian Shape Model

The matching of a deforming contour to the object in a given image can be formulated as Maximizing *A Posteriori* (MAP) estimation [8]. Denote the mean of the sample contours in the shape domain as $\bar{F}_0$ (called "mean contour"), the deformed version of $\bar{F}_0$ as $\bar{F}$ (called "prototype contour"), and the deforming contour in the image domain as F, where $\bar{F}_0 \in R^{2\times N}$, $\bar{F} \in R^{2\times N}$ and $F \in R^{2\times N}$ are the matrices representing the corresponding contours formed by the coordinates of N landmark/boundary points. For convenience, these contours are also denoted by expanding the $2 \times N$ matrices to the corresponding $2N \times 1$ vectors.

In this paper, both of the terms "contour F" and "contour $\mathbf{f}$" indicate the same contour, and the difference is that the former refers to a matrix, while the latter is a vector. Also, $\mathbf{f}_i$ represents the ith point of contour F.

According to the Bayesian estimation, the joint posterior distribution of $\mathbf{f}$ and $\bar{\mathbf{f}}$, $p(\mathbf{f}, \bar{\mathbf{f}}|d)$, is [8]

$$p(\mathbf{f}, \bar{\mathbf{f}}|d) = \frac{p(d|\mathbf{f})p(\mathbf{f}, \bar{\mathbf{f}})}{p(d)}, \quad (1)$$

where $p(d|\mathbf{f}) = p(d|\mathbf{f}, \bar{\mathbf{f}})$ is the likelihood distribution of input image data d.

$$p(\mathbf{f}, \bar{\mathbf{f}}) = p(\mathbf{f}|\bar{\mathbf{f}})p(\bar{\mathbf{f}}) \quad (2)$$

is the joint prior distribution of $\mathbf{f}$ and $\bar{\mathbf{f}}$. For a given image d, the MAP estimates, $\mathbf{f}_{MAP}$ and $\bar{\mathbf{f}}_{MAP}$, can be defined as

$$\begin{aligned} \{\mathbf{f}_{MAP}, \bar{\mathbf{f}}_{MAP}\} &= \arg\max_{\mathbf{f},\bar{\mathbf{f}}} \{p(\mathbf{f}, \bar{\mathbf{f}}|d)\} \\ &= \arg\max_{\mathbf{f},\bar{\mathbf{f}}} \{\frac{p(d|\mathbf{f})p(\mathbf{f}|\bar{\mathbf{f}})p(\bar{\mathbf{f}})}{p(d)}\} \quad (3) \end{aligned}$$

The Bayesian framework is a MAP estimation of the joint prior distribution of $\mathbf{f}$ and $\bar{\mathbf{f}}$. In Eq.(3), both $\mathbf{f}_{MAP}$ and $\bar{\mathbf{f}}_{MAP}$ are estimated, and the resulting contour $\mathbf{f}_{MAP}$ can be different in shape with its prototype $\bar{\mathbf{f}}_{MAP}$. Two kinds of shape variations are considered, $p(\bar{\mathbf{f}})$ models the prior distribution of the the shape variation and reflects major shape variations of the object shape, and $p(\mathbf{f}|\bar{\mathbf{f}})$ considers the local shape variations between $\mathbf{f}$ and $\bar{\mathbf{f}}$. Therefore, the joint MAP estimation can match object shapes more accurately than the classic MAP estimation.

When the densities can be modeled as Gibb's distribution, *i.e.*

$$\begin{aligned} p(\bar{\mathbf{f}}) &= Z_1^{-1} \exp\{-E_{con}(\bar{\mathbf{f}})\}, \\ p(\mathbf{f}|\bar{\mathbf{f}}) &= Z_2^{-1} \exp\{-E_{int}(\mathbf{f}|\bar{\mathbf{f}})\}, \quad (4) \\ p(d|\mathbf{f}) &= Z_3^{-1} \exp\{-E_{ext}(d|\mathbf{f})\}, \end{aligned}$$

where Z_1, Z_2 and Z_3 are the partition functions, maximizing the posterior distribution is equivalent to minimizing the corresponding energy function of the contour,

$$\{\mathbf{f}_{MAP}, \bar{\mathbf{f}}_{MAP}\} = \arg\min_{\mathbf{f},\bar{\mathbf{f}}} \{E_{BSM}\}, \quad (5)$$

where

$$E_{BSM} = \lambda_{con}E_{con} + \lambda_{int}E_{int} + \lambda_{ext}E_{ext}, \quad (6)$$

and $E_{con} = E_{con}(\bar{\mathbf{f}})$ is the constraint energy term of the adjustable prototype contour $\bar{\mathbf{f}}$, which limits the variations of $\bar{\mathbf{f}}$ and ensures that $\bar{\mathbf{f}}$ is similar with $\bar{\mathbf{f}}_0$ in shape. $E_{int} = E_{int}(\mathbf{f}|\bar{\mathbf{f}})$ is the internal energy term that describes the global and local shape deformation between $\mathbf{f}$ and $\bar{\mathbf{f}}$. The external energy term $E_{ext} = E_{ext}(d|\mathbf{f})$ defines the degree of matching between $\mathbf{f}$ and the salient image features. λ_{con}, λ_{int} and λ_{ext} are the manually set weighting parameters to regularize the energy terms. They are consistent for all the tests in an experiment or application. Please refer to [10] for more details of the BSM.

3 2D Full-Face Model and Facial Feature Extraction

3.1 Face Modeling and Matching

In this section the full-face model is proposed and the prior distribution of the whole face and the separate component models is estimated using PCA. As shown in Fig.1, a full-face model are formed by the boundary/landmark points of the face outline, mouth, nose, eyes and eyebrows, as well as other scattered or isolated points. These points of the full-face model can be divided into two sets:

1. Contour points: the contour points of the face outline, mouth, nose, eyebrows and eyes; (88 points)
2. Control points: the points that are very useful for face image warping, but not correspond to salient image features, *e.g.* edges. (18 points)

Fig.2 plots several examples of the full-face model, where the contour points and control points are linked to formulate a number of triangles or face units. It can be seen that this plot is similar to the *CANDIDE* face model. However, it is worth noting that the *CANDIDE* mode is a 3D face model for plotting, animating and synthesizing faces in various applications such as multimedia.

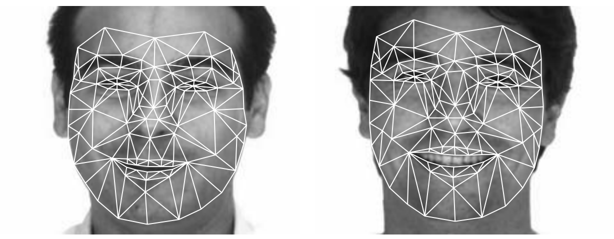

Figure 2: Link the model points to triangles.

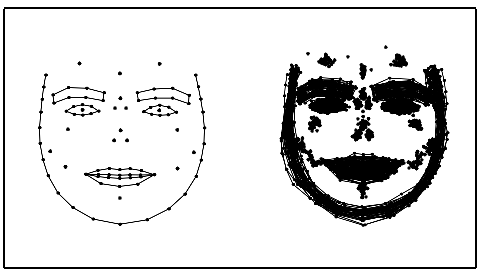

Figure 3: The mean face model and all of the 39 normalized face models.

Denoting the full-face model in the shape domain as $\bar{\mathbf{s}}$, and N refers to the number of points,

$$\bar{\mathbf{s}} = \begin{bmatrix} \bar{\mathbf{f}} \\ \bar{\mathbf{c}} \end{bmatrix}, \tag{7}$$

where $\bar{\mathbf{f}}$ $(N_f \times 1)$ represents the contour points and $\bar{\mathbf{c}}$ $(N_c \times 1)$ is the control points, and $N = N_f + N_c$. The contour points of the face consist of 7 component contours of separate facial features: a face outline contour, a mouth, a nose, two eyes and two eyebrows (see Fig.3). These component contours are denoted as $\bar{\mathbf{f}}_i, 1 \le i \le 7$, and

$$\bar{\mathbf{f}} = \left[\bar{\mathbf{f}}_1^T \ \bar{\mathbf{f}}_2^T \ \cdots \bar{\mathbf{f}}_7^T\right]^T. \tag{8}$$

The PCA is utilized to estimate the statistical model of the full-face model. First, all the sample face images are manually marked and normalized/aligned to a standard view by using least square errors method. *i.e.* the affine transformation is resolved to transfer the samples from the image domain onto the standard shape domain. Then, the distribution of the full-face model, the whole contour model as well as the separate component contour models of the face are established from these normalized sample data using PCA. These three kinds of shape models are summarized as follows,

1. Full-face model ($\bar{\mathbf{s}}$): a hybrid overall distribution model for all the contour points and control points of the full-face in the shape domain. Using PCA, given a shape parameter $\mathbf{w}_s$, the full-face model is reconstructed by,

   $$\bar{\mathbf{s}} = \bar{\mathbf{s}}_0 + \Phi_s \mathbf{w}_s, \tag{9}$$

   where $\bar{\mathbf{s}}_0$ is the mean vector of all the samples $\bar{\mathbf{s}}_i$, and Φ_s is the matrix composed of the eigenvectors corresponding to the largest M_s eigenvalues, which are computed from the covariance matrix of all of the samples. On the other hand, the shape parameter of an input sample, $\bar{\mathbf{s}}$, can be determined by

   $$\mathbf{w}_s = \Phi_s^T (\bar{\mathbf{s}} - \bar{\mathbf{s}}_0). \tag{10}$$

   The full-face model is very useful for estimating the control points.

2. Whole contour model ($\bar{\mathbf{f}}$): the distribution model that takes all the contour points as a whole. The shape parameters of the whole contour model is

   $$\mathbf{w}_f = \Phi_f^T (\bar{\mathbf{f}} - \bar{\mathbf{f}}_0), \tag{11}$$

   and a new contour is synthesized from a given shape parameter $\mathbf{w}_f$,

   $$\bar{\mathbf{f}} = \bar{\mathbf{f}}_0 + \Phi_f \mathbf{w}_f. \tag{12}$$

   The whole contour model is useful for matching the overall face object.

3. Component contour model ($\bar{\mathbf{f}}_i, \ 1 \le i \le 7$) : separate distribution model for each component contour. The construction of the component models is similar to 1 and 2.

To apply the BSM algorithm to facial feature extraction, first the BSM is used to match all the contour points $\mathbf{f}$ of the face model to the target face in the input image. During the matching procedure, both the whole and separate contour models are utilized to ensure that the facial contours are matched accurately. Finally, the control points are estimated from the matching result of the contour points based on the full-face model, to extract the face patch. Further processes like face patch warping can be performed very easily.

The method for estimating the control points from the contour points using the prior distribution of the full-face model is proposed in the following section.

3.2 Estimation of the Control Points

Suppose the current deforming contour $\mathbf{f}$ in the image domain and its prototype contour $\bar{\mathbf{f}}$ are known, the corresponding transformation parameters A and T (between $\mathbf{f}$ and $\bar{\mathbf{f}}$, and thus F and $\bar{F}$) can be estimated using alignment algorithms. Then, the contour points can be transformed into the shape domain using,

$$\bar{F}' = A^{-1}(F - T), \tag{13}$$

where A is 2x2 transformation matrix, $T = \mathbf{t}[1, 1, ..., 1]$ is a $2 \times N_f$ matrix, $\mathbf{t}$ is the translation vector. To estimate the control points corresponding to $\bar{\mathbf{f}}'$ using the full-face distribution model, in this section, a fast approximate algorithm is proposed. The idea is to estimate the shape parameter $\hat{\mathbf{w}}_s$, so that the contour points $\hat{\bar{\mathbf{f}}}$ of the reconstructed full-face model $\hat{\bar{\mathbf{s}}} = \begin{bmatrix} \hat{\bar{\mathbf{f}}} \\ \hat{\bar{\mathbf{c}}} \end{bmatrix}$ matches $\bar{\mathbf{f}}'$ closely, and hence the corresponding reconstructed control points $\hat{\bar{\mathbf{c}}}$ can be regarded as the estimate of the control points.

The detail procedure is described as follows,

1. The estimation problem is to find out $\hat{\mathbf{w}}_s$, so that the result $\hat{\bar{\mathbf{f}}}$ matches $\bar{\mathbf{f}}'$ in the sense of least square errors. From Eq.(9), $\hat{\bar{\mathbf{f}}}$ is calculated by

$$\begin{aligned} \hat{\bar{\mathbf{f}}} &= [I\ \mathbf{0}] \begin{bmatrix} \hat{\bar{\mathbf{f}}} \\ \hat{\bar{\mathbf{c}}} \end{bmatrix} \\ &= [I\ \mathbf{0}](\bar{\mathbf{s}}_0 + \Phi_s \hat{\mathbf{w}}_s) \\ &= [I\ \mathbf{0}]\bar{\mathbf{s}}_0 + [I\ \mathbf{0}]\Phi_s \hat{\mathbf{w}}_s, \end{aligned} \tag{14}$$

   where I is a unity matrix and $\mathbf{0}$ is a zero matrix. Noting $\bar{\mathbf{f}}_0 = [I\ \mathbf{0}]\bar{\mathbf{s}}_0$ and denoting $\Phi_f{}' = [I\ \mathbf{0}]\Phi_s$, from $\bar{\mathbf{f}}_0$, $\Phi_f{}'$, $\bar{\mathbf{f}}'$ and Eq.(10), the shape parameter $\hat{\mathbf{w}}_s$ can be estimated using

$$\hat{\mathbf{w}}_s = \Phi_f{}'^T(\bar{\mathbf{f}}' - \bar{\mathbf{f}}_0). \tag{15}$$

2. The control points $\hat{\bar{c}}$ can be calculated through $\hat{\mathbf{w}}_s$,

$$\hat{\bar{\mathbf{c}}} = [\mathbf{0}\ I](\bar{\mathbf{s}}_0 + \Phi \hat{\mathbf{w}}_s). \tag{16}$$

3. The final full-face contour in the shape domain is represented by

$$\hat{\bar{\mathbf{s}}} = \begin{bmatrix} \bar{\mathbf{f}}' \\ \hat{\bar{\mathbf{c}}}, \end{bmatrix} \tag{17}$$

   thus its affine transformed counterpart in the image domain $\mathbf{s} = \begin{bmatrix} \mathbf{f} \\ \mathbf{c} \end{bmatrix}$ can be calculated.

In this way, the extracted face not only consists of the contour points but also the control points, and now it is ready to be warped or normalized by linking these points to triangular face units and using the piece-wise affine warping algorithm.

Note that the proposed method tries to estimate one set of points from another set by using the joint prior distribution of them. Alternatively, please refer to [13] for solving this problem using optimization method.

The above estimation performs very fast, and its calculation begins from the face contour, $\mathbf{f}$, and ends at the control points, $\mathbf{c}$, as,

$$\mathbf{f} \to \bar{\mathbf{f}}' \to \hat{\mathbf{w}}_s \to \hat{\bar{\mathbf{c}}} \to \mathbf{c}. \tag{18}$$

The principle behind this approximate estimation is that the information provided by the contour and control points of a full-face model is redundant, *i.e.* using partial contour information, the whole shape of the full-face can be described approximately.

In order to evaluate this estimation quantitatively, we tested the algorithm using a number, NUM, of manually marked images. The average error or distance between the estimated control points (NUM_C is the number of the control points) and the marked points in the shape domain is used:

$$Error = \frac{1}{NUM \cdot NUM_C} \sum_{NUM} \sum_{i=1}^{NUM_C} ||\bar{\mathbf{c}}_i - \hat{\bar{\mathbf{c}}}_i||. \tag{19}$$

Table 1: Average error of the estimation (in pixel).

Test images	Number of images	Average error	Error %
Training samples	25	1.3	0.4%
Testing images	25	3.05	1.0%

In the experiment, 50 marked images with different face expressions are used, which are partitioned into two sets. The first set consists of 25 training sample images, while the other uses the marked images not used for training. Table 1 shows the average error of the estimation for both of the trained sample images and the testing images. It can be seen from Table 1 that although more accurate estimation can be obtained from the trained sample images, the average error is also small for the other image set, which are not used for training, the value is 1.3 *vs.* 3.05. Moreover, it can be seen that the relative errors, *i.e.* the error/scale ratio, are very small with respect to the scale of mean face model in the shape domain, *i.e.* 360×294.

(a) the original face patches.

(b) warped face patches.

Figure 4: Demonstration of the face patch warping.

Fig.4 shows several examples of the warping results, where the contour points are known and the control points are estimated by using the proposed algorithm, these full-face models are then utilized for face patch warping. In the figure, the upper row is the original face patches, while the bottom row corresponds to the warped images of the upper row, which changes the expressions of an extracted face patch. For example, the first column of the images illustrates how to change the expression to smile, and the second column is a reverse procedure. It can be seen from the figure that the proposed full-face model describes the face well, and through image warping, the expressions of face can be removed/changed, which is very useful for further face recognition, 2D face animation, content based face image coding and retrieval in multimedia applications.

4 Experimental Results

In the experiment, a simple facial feature extraction system is implemented to demonstrate the performance of BSM. The system uses the proposed full-face model to describe the face contour, and the distribution parameters of the model, *i.e.* the mean face contour and the shape parameters of the shape variation, are trained off-line, and the training samples are the images from the AR face database.

The procedure for the operation can be summarized as follows:

- Capturing an image from the USB video camera or loading an existing face image file from disk.
- Initializing the face contour manually on the input image, so that it is close to the target face.
- Matching the input face iteratively using the proposed BSM algorithm.
- Extracting the face patch and warping it to the mean face contour in the shape domain.

(a) Initial (b) Matching results

Figure 5: Examples of the matching results.

In this system, the initialization of the contour is manually set. In further projects, the modules of face detection, model initialization and face contour extraction will be combined together. The initialization is done by manually selecting three points in the input image (the three points that we select are the center points of the left, right eyes and the mouth). Then the corresponding transformation parameters of the initial contour can be found by aligning the selected three points with their corresponding three points in the shape domain. At last, the initial contour is set as the transformed version of the mean face contour.

Fig.5 and 6 plot some matching results of the face images, including the initial contours, the matching results and the warped versions of the extracted face patches. The resolution of these images is 320×240, and the size of the face is approximately half of the image size. It can be seen from the figures that good matching and extraction results are obtained. The good matching results using BSM on the face images captured from USB camera further indicated the potential applications of the BSM facial feature extraction algorithm. For example, in face recognition system, BSM can be used to accurately match and extract the face patches, then the classification based on the shape and the appearance information of the extracted face patch, such as EigenFace algorithm [14] can be performed. In face image indexing, coding and multimedia systems, BSM facial feature extraction algorithm can also be utilized to extract the patches of the face of interest accurately as long as the prior face model is trained and the face can be detected, located properly close to the target face position.

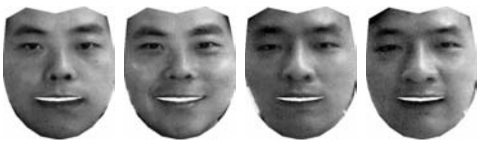

Figure 6: The normalized versions of the extracted face patches of Fig.5

5 Conclusion

The facial feature extraction using BSM is studied in this paper. A full-face model is developed to describe the shape of a face. Using this model, face patches can be easily extracted and warped to a normal position. The applications of the BSM facial feature extraction algorithm include face recognition, face video coding and retrieval, face animation and multi-media.

References

[1] A. Jain, Y. Zhong, and S. Lakshmanan, "Object matching using deformable templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 3, pp. 267–278, 1996.

[2] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active shape models - their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[3] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[4] D. Terzopoulos, A. Witkin, and M. Kass, "Constraints on deformable models: Recovering 3d shape and nonrigid motion," *Artificial Intelligence*, vol. 36, no. 1, pp. 91–123, 1988.

[5] A. Yuille, P. Hallinan, and D. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, vol. 8, no. 2, pp. 99–111, 1992.

[6] S. Lakshmanan and D. Grimmer, "A deformable template approach to detecting straight edges in radar images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 438–443, 1996.

[7] K. Lai and R. Chin, "Deformable contours: modeling and extraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 11, pp. 1084–1090, 1995.

[8] S. Z. Li and J. Lu, "Modeling Bayesian estimation for deformable contours," in *Proceedings of 7th IEEE International Conference on Computer Vision, Kerkyra, Greece.*, pp. 991–996, 1999.

[9] H. Ip and D. Shen, "An affine-invariant active contour model (AI-Snake) for model-based segmentation," *Image and Vision Computing*, vol. 16, no. 2, pp. 125–146, 1998.

[10] Z. Xue, S. Li, and E. Teoh, "Ai-eigensnake: An affine-invariant deformable contour model for object matching," *Image and Vision Computing*, vol. 20, no. 2, pp. 77–84, 2002.

[11] B. Moghaddam and A. Pentland, "Probabilistic visual learninig for object representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.

[12] M. Rydfalk, *CANDIDE, a parameterized face.* Report No. LiTH-ISY-I-866, Dept. of Electrical Engineering, Linkping University, Sweden, 1987.

[13] C. Davatzikos, D. Shen, A. Mohamed, and S. Kyriacou, "A framework for predictive modeling of anatomical deformation," *IEEE Trans. on Medical Imaging*, vol. 20, no. 8, pp. 836–843, 2001.

[14] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Congnitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.