Accurate Face Alignment using Shape Constrained Markov Network

Lin Liang, Fang Wen, Ying-Qing Xu, Xiaoou Tang and Heung-Yeung Shum
Visual Computing Group
Microsoft Research Asia
Beijing 100080, China
{lliang, fangwen, yqxu, xitang and hshum}@microsoft.com

Abstract

In this paper, we present a shape constrained Markov network for accurate face alignment. The global face shape is defined as a set of weighted shape samples which are integrated into the Markov network optimization. These weighted samples provide structural constraints to make the Markov network more robust to local image noise. We propose a hierarchical Condensation algorithm to draw the shape samples efficiently. Specifically, a proposal density incorporating the local face shape is designed to generate more samples close to the image features for accurate alignment, based on a local Markov network search. A constrained regularization algorithm is also developed to weigh favorably those points that are already accurately aligned. Extensive experiments demonstrate the accuracy and effectiveness of our proposed approach.

1. Introduction

Shape alignment is an actively studied problem in computer vision. Applications of shape alignment range from medical image processing [12], object tracking [15], face recognition [14] and modeling [4], to face cartoon animation [8]. Accurate alignment of face shapes or contours depends on parameter estimation of an optimal deformable shape model that matches the image evidence collected from a single image or a video sequence.

A number of different shape models have been proposed for face alignment. One approach is to postulate the deformation parameters by reducing the shape deformation correlations. The shape prior is then modeled by the distribution of the deformation parameters. After the pioneering work on active shape model (ASM) [2], many interesting models have been developed. For example, a Bayesian tangent shape model is proposed in [16] to make the parameter estimation more accurate and robust by using an EM based searching algorithm. To alleviate the local minima problem, a hierarchical shape model and DDMCMC inference algorithm are designed in [7]. The condensation algorithm has also been adopted in the ASM framework in [13]. To handle the nonlinear shape variance, a mixture of Gaussians [1] and kernel PCA [11] are used to model the distribution of deformation parameters. These methods usually generate an observed shape by sampling each feature point independently from a local likelihood and then regularize it using the shape prior model. The common problem for all these methods is that each feature point is sampled without considering its relationship with neighbor points. Since no local shape constraints are applied to the neighbor points, the observed location of each individual point is very sensitive to noise. Although the global regularization based on a shape prior may help to assure an overall shape reasonably similar to a face, it is difficult to obtain the accurate shape locations.

Recently, Markov Random Field model has been proposed for face alignment [3]. In this method, each feature point is considered as a node in a graph, and a link is set between each pair of feature points with the interaction energy designed to impose the local structure constraints between them. The benefit of such a model is that the shape prior is distributed in a Markov network of components and the image observation is still distributed by modeling the image likelihood of each individual component. The close interaction between the local image observation and structure constraints leads to more accurate local shape estimation. The shortcoming of this approach is that it models the shape only in a local neighborhood. Such a low level model cannot capture high level semantics in the shape. The lack of a global shape prior often leads such methods to nonstable results.

In our recent work [6], we adopted the Markov network to find an optimal shape, then regularize the shape by the PCA based shape prior though a constrained regularization algorithm. Although such integrated model can improve the alignment preciseness to some extend, but the regularization step does not consider the image information anymore, therefore the alignment's precision is still not good enough for some cases.

Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)
0-7695-2597-0/06 $20.00 © 2006 IEEE

In this paper, we develop an approach to incorporate the global shape prior directly into the Markov network for accurate face alignment. We are inspired by recent work of Object Cut [5], where robust object segmentation is obtained by integrating the global shape prior of an object into a Markov random field.

In our approach, we decompose the face shape into a number of small line segments to construct the Markov network. The global shape prior is defined as a set of weighted shape samples. For accurate face alignment, these shape samples should be mostly drawn from places close to the image features. We obtain such samples by designing a proposal density to incorporate the local geometry constraints between line segments. Moreover, we develop a new constrained regularization algorithm to keep the positions of those already aligned sampling points. Our algorithm is accurate and robust for face alignment, as demonstrated by extensive experiments.

2. Global Shape Constrained Markov Network for Face Alignment

2.1 Markov Network Shape Model

Assuming that a shape $\mathbf{S}$ is described by N feature points $\mathbf{s}_i = (x_i, y_i)$ in the image, we can represent it by a $2N$-dimensional vector $\mathbf{S} = \{(x_i, y_i), i = 1, ..., N\}$. We break the shape $\mathbf{S}$ into a set of line segments by the feature points. The parameters of each line segment $\mathbf{q}_i$ are the coordinates of its two endpoints $\mathbf{q}_i = [\mathbf{w}_i^s, \mathbf{w}_i^e]$. As shown in Figure 1, these line segments are the nodes in the hidden layer of the graph. If two nodes are correlated, there will be an undirected link between them. For a deformable shape, we assign a link between any pair of connected line segments.

Assuming the Markovian property among the nodes, the shape prior can be modeled as $p(\mathbf{Q})$, $\mathbf{Q} = \{\mathbf{q}_0, \mathbf{q}_1, ..., \mathbf{q}_K\}$, which is a Gibbs distribution and can be factorized as a product of all the potential functions over the cliques in the graph:

$$p(\mathbf{Q}) = \frac{1}{Z} \prod_{c \in C} \psi_c(\mathbf{Q}_c) \qquad (1)$$

where C is a collection of cliques in the graph, $\mathbf{Q}_c$ is the set of variables corresponding to the nodes in clique c, and Z is the normalization constant or the partition function.

In the context of deformable shapes, we adopt a pairwise potential function $\psi_{ij}(\mathbf{q}_i, \mathbf{q}_j)$ to present the constraint between two connected line segments. Thus we write the shape prior $p(\mathbf{Q})$ as:

$$p(\mathbf{Q}) = \frac{1}{Z} \prod_{(i,j) \in C^2} \psi_{ij}(\mathbf{q}_i, \mathbf{q}_j) \qquad (2)$$

The pairwise potential function is defined by the constraints of the distance of two endpoints ($\mathbf{w}_i^e$ and $\mathbf{w}_j^s$) and the angle γ_{ij} between the two line segments, as illustrated in Figure 2:

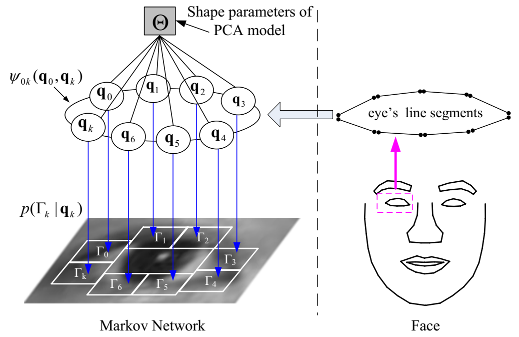


Figure 1. An illustration of the global shape constrained Markov network for a face. Using eye shape as an example, each node q_i in the graph is a line segment and q_i is associated with its image observation Γ_i. All the nodes are constrained by the global shape parameter $\mathbf{\Theta}$. The face shape is to show that for the graph of face, the links are added within each black line.

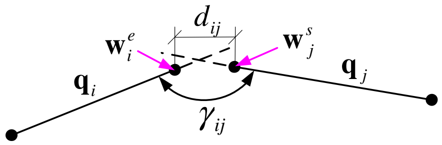


Figure 2. The constraints of two connected line segments: the distance d_{ij} between two end points, and the angle γ_{ij} between two connecting line segments.

$$\psi_{ij}(\mathbf{q}_i, \mathbf{q}_j) = G(d_{ij}; 0, \sigma_{ij}^d) \cdot G(A_{ij}; \mu_{ij}^A, \sigma_{ij}^A) \qquad (3)$$

where $d_{ij} = |\mathbf{w}_i^e - \mathbf{w}_j^s|$ is the distance between $\mathbf{w}_i^e$ and $\mathbf{w}_j^s$, $A_{ij} = \sin(\gamma_{ij})$, and σ_{ij}^d and σ_{ij}^A are variance parameters. σ_{ij}^d that control the tightness of the connectivity constraint.

Given the image observation I, as shown in Figure 1, each segment $\mathbf{q}_i$ is also associated with its image observation, denoted as Γ_i. Assuming the local observation to be independent of other nodes given $\mathbf{q}_i$, the likelihood is factorized as:

$$p(I|\mathbf{Q}) = \prod_i p_i(\Gamma_i|\mathbf{q}_i) \qquad (4)$$

Then the posterior can be factorized as:

$$p(\mathbf{Q}|I) \propto \frac{1}{Z} \prod_i p_i(\Gamma_i|\mathbf{q}_i) \prod_{(i,j) \in C^2} \psi_{ij}(\mathbf{q}_i, \mathbf{q}_j) \qquad (5)$$

The posterior can be maximized efficiently by Belief Propagation [10]. During the inference, the movement of each line segment is affected by the local image observation together with the local geometry constraints coming from its neighboring nodes. Compared with the centralized algorithms [2, 7] where the shape is regularized after each feature point is moved independently, the movements of feature points are more consistent and a more accurate result can be achieved.

2.2 Global Shape Constrained Markov Network

Although the Markov network can achieve a more accurate alignment result. However, since the currently designed graph only models the local geometry constraints between the neighboring feature points by the pairwise potential, the shape prior is often too local to guarantee a globally reasonable shape, especially when a strong edge appears near the ground truth.

Many algorithms like [2] adopt a generative shape prior model: the shape is generated by the combination of a set of deformable modes. A common way to do this is first align the shape to the tangent space, then adopt PCA to find the deformable modes. The aligned shape $\mathbf{x}$ is then generated by

$$\mathbf{x} = \mu + \phi_r \mathbf{b} + \varepsilon \tag{6}$$

where μ is the mean shape, and ϕ_r consists of the first r columns of the projection matrix. Each column of ϕ_r corresponds to a deformable mode. ε is an isotropic noise in the tangent space. $\mathbf{b}$ defines the shape intrinsic deformation. Then the global shape prior is modeled as $p(\mathbf{S}) \sim p(\mathbf{b})$.

This implies that all the nodes in the graph (Figure 1) are correlated by an underlying shape parameter. The question is how to add such constraint into the optimization of Markov network Equation (5). Further considering the the extrinsic parameters, e.g., scaling c, rotation R_θ, and translation $\mathbf{t}$, we denote the global shape parameter as $\mathbf{\Theta} = \{c, R_\theta, \mathbf{t}, \mathbf{b}\}$. As in [5], we treat $\mathbf{\Theta}$ as the missing data, based on the EM algorithm, the optimal position of the line segments $\mathbf{Q}$ can be obtained by iteratively maximizing the lower bound of $p(\mathbf{Q}|I)$:

$$\mathbf{Q}^* = \arg\max_{\mathbf{Q}} \int_{\mathbf{\Theta}} \log p(\mathbf{Q}|\mathbf{\Theta}, I) p(\mathbf{\Theta}|I, \mathbf{Q}_{t-1}) d\mathbf{\Theta} \tag{7}$$

In [5], random samples $\mathbf{\Theta}^{(1)}$, ... ,$\mathbf{\Theta}^{(N)}$ are drawn from the distribution $p(\mathbf{\Theta}|I, \mathbf{Q}_{t-1})$, then Equation (7) can be approximated as:

$$\mathbf{Q}^* = \arg\max_{\mathbf{Q}} \sum_k w_k \cdot \log p(\mathbf{Q}|\mathbf{\Theta}^{(k)}, I) \tag{8}$$

where $w_k = p(\mathbf{\Theta}^{(k)}|I, \mathbf{Q}_{t-1})$. Note that:

$$p(\mathbf{Q}|\mathbf{\Theta}^{(k)}, I) \propto p(\mathbf{\Theta}^{(k)}|\mathbf{Q}) p(\mathbf{Q}|I) \tag{9}$$

Given an image I, set the initial shape as the mean shape of the training data.

1. Adopt hierarchical condensation to obtain the samples $\mathbf{S}(\mathbf{\Theta}^{(1)}) \dots \mathbf{S}(\mathbf{\Theta}^{(N)})$
2. Obtain the mean shape of the weighted samples, generate the line segment candidates along the profile, as in Figure 4.
3. Compute $\log p(\mathbf{Q}|\mathbf{\Theta}^{(k)}, I)$ and set $w_k \approx p(\mathbf{\Theta}^{(k)}|I)$
4. The object function Equation 8 is maximized by belief propagation algorithm to obtain the optimal shape.

Table 1. The overview of our algorithm.

Given a global parameter $\mathbf{\Theta}^{(k)}$, a shape $\mathbf{S}^{(k)}$ can be generated. We expect the final shape inferred by the Markov network to be close to $\mathbf{S}^{(k)}$. Unlike the segmentation work in [5], we define the probability $p(\mathbf{\Theta}^{(k)}|\mathbf{Q})$ to make not only the feature point position close to the given shape $\mathbf{S}^{(k)}$, but also the relative position between two linked line segments close to the relation appearing in $\mathbf{S}^{(k)}$:

$$p(\mathbf{\Theta}^{(k)}|\mathbf{Q}) \propto \prod_i p_i(\mathbf{S}^{(k)}|(\mathbf{q}_i)) \prod_{(i,j)\in C^2} \psi_{ij}(\mathbf{q}_i, \mathbf{q}_j|\mathbf{S}^{(k)}) \tag{10}$$

where

$$p_i(\mathbf{S}^{(k)}|\mathbf{q}_i)) = \frac{1}{1 + \exp(d(\mathbf{q}_i, \mathbf{S}^{(k)}))} \tag{11}$$

$d(\mathbf{q}_i, \mathbf{S}^{(k)})$ is the distance of a line segment from the given shape.

$$\psi_{ij}(\mathbf{q}_i, \mathbf{q}_j|\mathbf{S}^{(k)}) = G(A_{ij}; \tilde{A}_{ij}, \nu_A) \tag{12}$$

$\tilde{A}_{ij}$ is the sine value of the angle between two given line segments, and ν_A is set as a small value. From Equation (5) and Equation (10):

$$\log p(\mathbf{Q}|\mathbf{\Theta}^{(k)}, I) = \sum_i (\log p_i(\Gamma_i|\mathbf{q}_i) + \log p_i(\mathbf{S}^{(k)}|\mathbf{q}_i)$$
$$+ \sum_j \log \psi_{ij}(\mathbf{q}_i, \mathbf{q}_j) + \log \psi_{ij}(\mathbf{q}_i, \mathbf{q}_j|\mathbf{S}^{(k)})) + const \tag{13}$$

Notice that based on the above definition, Equation (8) can still be maximized by the belief propagation algorithm. Then the key problem is how to draw samples from $p(\mathbf{\Theta}|I, \mathbf{Q}_{t-1})$. Similar to [5], we use $p(\mathbf{\Theta}|I)$ to approximate the initial distribution of $p(\mathbf{\Theta}|I, \mathbf{Q}_{t-1})$. In the following section, we will explain how we adopt the condensation algorithm [9] hierarchically to sample $p(\mathbf{\Theta}|I)$.

The overview of our accurate face alignment algorithm is summarized in Table 1.

3. Local Shape Constrained Sampling of $p(\mathbf{\Theta}|I)$

It is not easy to sample the distribution $p(\mathbf{\Theta}|I)$, because it is non-analytical and high dimensional. The condensation algorithm [9] provides an efficient way to approximate a complex distribution based on factored sampling. Recently it has been used for the face alignment by [13]. We adopt the hierarchical condensation [13] to sample $p(\mathbf{\Theta}|I)$, but design a more efficient proposal density $p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1})$ based on the Markov network model. We also develop a constrained regularization to improve the accuracy of the locations of sampled shapes.

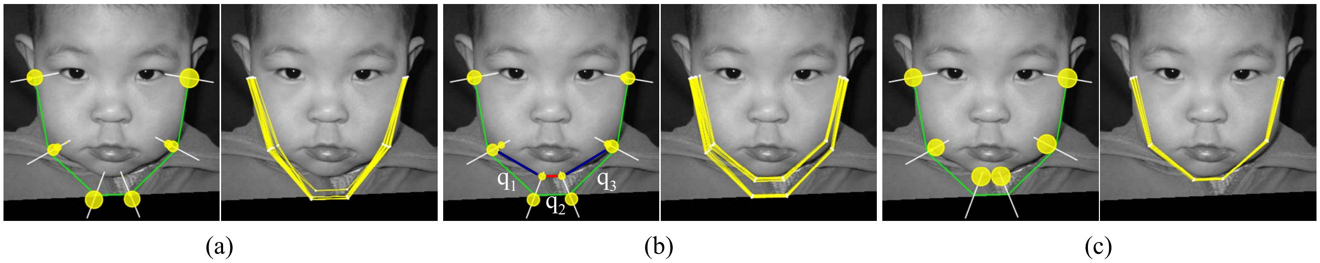


Figure 3. The comparison of different sampling strategies. (a) Sampling each feature point independently as in [13]. (b) and (c) are the results of our algorithm but using different likelihood models for the line segment: (b) models the likelihood only using the features of two endpoints, (c) uses some edge-based features. The results are from one condensation iteration at the lowest resolution level with the image size as 64x64 and the local search length as 5. The left image in (a), (b) or (c) shows the sampling probability of the feature point along the profile. The green line is the initial shape. The probability is proportional to the size of the yellow point. The right image in (a), (b) or (c) shows 100 shapes sampled by each method.

3.1 Hierarchical Condensation

We denote $\mathbf{\Theta}_i$ as the global shape parameters of iteration i and $\mathcal{I}_i = \{I_1, \dots, I_i\}$ as the entire image stream up to that time. In condensation [9], the rule to propagate the posterior $p(\mathbf{\Theta}_i|\mathcal{I}_i)$ is:

$$p(\mathbf{\Theta}_i|\mathcal{I}_i) \propto p(I_i|\mathbf{\Theta}_i)p(\mathbf{\Theta}_i|\mathcal{I}_{i-1}) \tag{14}$$

where

$$p(\mathbf{\Theta}_i|\mathcal{I}_{i-1}) = \int_{\mathbf{\Theta}_{i-1}} p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1})p(\mathbf{\Theta}_{i-1}|\mathcal{I}_{i-1})d\mathbf{\Theta}_{i-1} \tag{15}$$

To approximate $p(\mathbf{\Theta}_i|\mathcal{I}_i)$, samples $\{\mathbf{S}_i^{(n)}\}$ are drawn from $p(\mathbf{\Theta}_{i-1}|\mathcal{I}_{i-1})$, then each sample is evolved by sampling the proposal density $p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1})$. Finally, the samples are weighted by the image likelihood $\pi_i^{(n)} = p(I_i|\mathbf{\Theta}_i = \mathbf{S}_i^{(n)})$. The posterior $p(\mathbf{\Theta}_i|\mathcal{I}_i)$ is represented by the weighted samples $\{\mathbf{S}_i^{(n)}, \pi_i^{(n)}\}_{n=1}^N$.

To make the sampling more efficient, we adopt a hierarchical structure similar to [7, 13]: the shape resolution changes from coarse to fine corresponding to a Gaussian pyramid of the image. The weighted samples obtained at the resolution level l are propagated to the higher resolution level $l+1$ by factored sampling. We model the proposal density between adjacent resolution layers $p(\mathbf{\Theta}_{l+1}|\mathbf{\Theta}_l)$ as a Gaussian, as in [7].

During the condensation of one resolution layer, we design the proposal density $p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1})$ to make the shape parameter move towards regions of higher image likelihood, as explained in the following section.

3.2 Sampling proposal density $p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1})$

3.2.1 Incorporate Local Shape Prior into Sampling

We design $p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1})$ to make the shape parameter move based on the current local image observation $\Gamma(\mathbf{\Theta}_{i-1})$: $p(\mathbf{\Theta}_i|\mathbf{\Theta}_{i-1}) \propto p(\mathbf{\Theta}_i|\Gamma(\mathbf{\Theta}_{i-1}))$. In the work in [13], each feature point $\mathbf{s}_j$ is sampled based on its local image likelihood independently: $p(\mathbf{\Theta}_i|\Gamma(\mathbf{\Theta}_{i-1})) \propto \prod_j p(\Gamma_{j,i-1}|\mathbf{s}_{j,i-1})$. Because no geometry constraints between feature points are considered during the sampling, the sampled shape is usually a zigzag. As a result, the regularized shape cannot attach to the image features closely, so a precise alignment result cannot be achieved. Also the sampling is sensitive to the local noise. As shown in Figure 3 (a), because of the strong edge, for the two points below the jaw, the probability to sample them is very small, so the sampled shapes are stuck into the wrong region.

Instead, we define the proposal density as the Markov network based posterior of the line segments $\mathbf{Q}_i$:

$$p(\mathbf{\Theta}_i|\Gamma(\mathbf{\Theta}_{i-1})) \propto p(\mathbf{Q}_i|\Gamma(\mathbf{\Theta}_{i-1})) \tag{16}$$

with the formulation as in Equation (5). In this way, the local geometry constraints are incorporated into the sampling stage.

To sample $p(\mathbf{Q}_i|\Gamma(\mathbf{\Theta}_{i-1}))$, we first discretize the state space of the Markov network in the neighbor of the current shape $\mathbf{S}_{i-1}$ generated from $\mathbf{\Theta}_{i-1}$, as shown in Figure 4. The BP algorithm is adopted to calculate the belief (the marginal posterior) $p(\mathbf{q}_k|\Gamma(\mathbf{\Theta}_{i-1}))$ of each line segment $\mathbf{q}_k$. Then the poses of line segments are sampled sequentially, one line segment at a time. To guarantee that the pose of the line segment sampled satisfies the geometry constraint with its neighbors previously sampled, each line segment is sampled from:

$$p(\mathbf{q}_k|\Gamma(\mathbf{\Theta}_{i-1}), N(\mathbf{q}_k)) \propto p(\mathbf{q}_k|\Gamma(\mathbf{\Theta}_{i-1})) \prod_{j \in N(\mathbf{q}_k)} \psi_{kj}(\mathbf{q}_k, \mathbf{q}_j) \tag{17}$$

where $N(\mathbf{q}_k)$ denotes the neighbors of $\mathbf{q}_k$ previously sampled and $\psi_{kj}(\mathbf{q}_k, \mathbf{q}_j)$ is the potential as in Equation (3).

Since the belief of each line segment is the product of the local image likelihood and the messages from its neighbors, a message from a good neighbor will help to alleviate the effect of local noise. As shown in Figure 3 (b), in this case, to

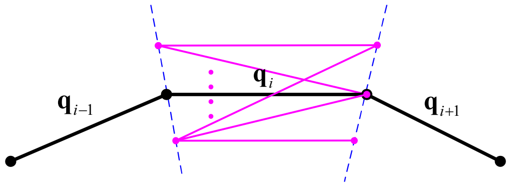


Figure 4. Generate Markov network states. The black line is the current shape $\mathbf{S}_l^t$. $\mathbf{q}_i, \mathbf{q}_{i-1}, \mathbf{q}_{i+1}$ are linked Markov network nodes. The candidate states of $\mathbf{q}_i$ are generated along the shape's profile.

use a similar point likelihood model as in [13], we only use the features of two endpoints to model the likelihood of a line segment. Under such model, the likelihoods of the blue lines (see the left image of Figure 3 (b)) of the node $\mathbf{q}_1$ and $\mathbf{q}_3$ are big. Because of the geometry constraint between the node $\mathbf{q}_2$ with $\mathbf{q}_1$ and $\mathbf{q}_3$, the belief of $\mathbf{q}_2$'s red line is increased. To compare with [13] clearly, we show the corresponding marginal distribution of the feature points in the left image of Figure 3 (b). It is obvious that compared with [13], the sampling probabilities of the jaw's feature points are increased. Furthermore, by using some edge-based features to model the likelihood, the ambiguity caused by the noisy edge is eliminated, as shown in Figure 3 (c).

3.2.2 Constrained Regularization

Based on the proposed line segments, the shape parameter $\mathbf{\Theta}$ can be obtained by the BTSM algorithm [16] to minimize the distance Δ between the generated shape $\mathbf{S}(\mathbf{\Theta})$ and the given shape under the constraint of the deformation parameters $\mathbf{b}$'s prior:

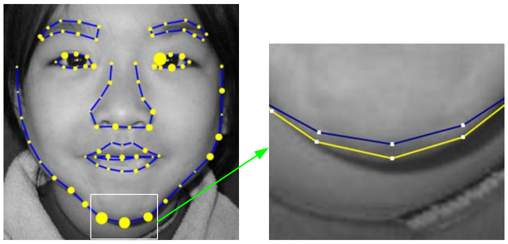
$$E_p = \Delta^T \Sigma_l^{-1} \Delta + \mathbf{b}^T \Sigma_b^{-1} \mathbf{b} \tag{18}$$

where $\Sigma_l = diag(\sigma_1^2, \sigma_2^2, ..., \sigma_N^2)$ is the likelihood variance matrix and Σ_b is the variance matrix of $\mathbf{b}$. Usually Σ_l is set as an identity matrix, and the regularization algorithm will try to find a solution that minimizes the distance to each point equally including bad points. As a result, some sampled points that are already at good positions may be dragged away, as shown in Figure 5. Integrating such global shapes as the constraints in the final Markov network optimization Equation (8) will decrease the alignment precision. If we can constrain those good points during the regularization, the problem can be alleviated.

Actually, from the beliefs of the line segments obtained by the local markov network search, we can approximate the marginal histogram $\{\mathbf{s}_k^n, w_k^n\}_{n=1}^M$ of the kth feature point's position $\mathbf{s}_k = (x_k, y_k)$. We parameterize the position distribution $p(\mathbf{s}_k)$ as a 1D Gaussian:

$$p(\mathbf{s}_k) = G(d(\mathbf{s}_k, \bar{\mathbf{s}}_k); 0, \eta_k) \tag{19}$$

where $\bar{\mathbf{s}}_k = \sum_n w_k^n \cdot \mathbf{s}_k^n$ is the mean position, and $d(\mathbf{s}_k, \bar{\mathbf{s}}_k)$ is the distance between the point and the mean $\bar{\mathbf{s}}_k$. From the statistical viewpoint, a peaked position distribution implies that there are no ambiguous regions nearby; therefore it's better to keep the point near the mean; while a smoother distribution means that the point's position is not clearly determined and its effect in Equation (18) should be decreased. Thus we set the weight of each point's error σ_k in Equation (18) in an inverse ratio to its position's probability:



(a) (b)

Figure 5. Constrained Regularization. (a) The constraint weights of the feature points. The blue (darker) line is the sampled shape. The weight is in a direct ratio to the size of the yellow point. (b) The results of regularization with (yellow line) and without (blue line) constraints.

$$\sigma_k = \frac{1}{1 + p(\mathbf{s}_k)} \tag{20}$$

As shown in Figure 5, the weights represent each point's confidence properly and the constrained regularization keeps the positions of the good points better than without constraint. We adopt such constrained regularization for the condensation at the high resolution layers to improve the final alignment precision.

4. Experiments

In our experiment, we use a set of face images of size 512 × 512. A total of 87 feature points are manually labeled on each image for both the training and testing data sets. This data set contains the photos of children from 2 to 15 years old with different expressions. Thus the face shape variance is large. Consequently, although the images have good quality, the data set is still difficult for precise alignment.

In the hierarchical condensation stage, a four-level Gaussian pyramid is built by repeated sub-sampling. For each image layer from coarse to fine, the corresponding face shape contains 18, 37, 57 and 87 feature points respectively.

4.1 Evaluation of the Shape Constrained Markov Network

To demonstrate the benefit of the shape constrained Markov network, we compare our algorithm with the Condensation and Condensation + Markov network (running the Markov network search after the Condensation). The result of the condensation is set as the mean of the weighted

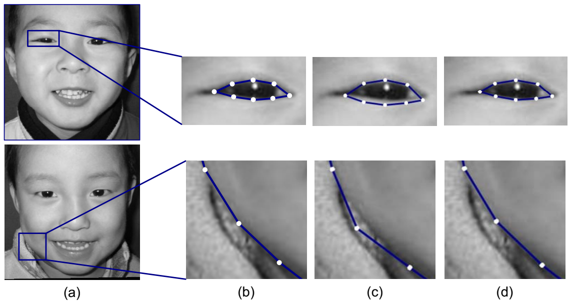


Figure 6. The comparison of our algorithm with Condensation and Condensation + Markov network. (a) is the source image and (b) is the labeled shape. In (c), the first row is Condensation's result, the second row is the result of Condensation + Markov network. (d) is our result.

samples and this mean shape is used as the initial shape for our algorithm and Condensation + Markov network.

As shown in the first row of Figure 6, the problem of the condensation is that after regularizing the shape by the PCA model, some good points that have already positioned at the boundary may be moved away and the resulting shape is not very accurate. And from the second row of Figure 6, we can see, although the Markov network has the ability to align to the boundary more accurately, it is sensitive to the local noise, and sometimes the searched shape seems unreasonable. For our algorithm, the integrated global shape's constraints will provide good guide for the Markov network search in the ambiguous regions caused by the local noises, thus a more reasonable and accurate shape can be obtained.

4.2 Comparison with BTSM algorithm

We compare our algorithm (succinctly named as SCMN in this section) with BTSM [16]. The reason for comparing with BTSM is that it is an improvement of the classic ASM algorithm and extensive experiments have demonstrated its good performance. To the best of our knowledge, BTSM is one of the most accurate face alignment systems to-date.

We first compare our algorithm with BTSM statistically. We divide the data set into 428 images for training and 350 images for testing. For each test image, an initial shape is generated by randomly rotating (from $-20°$ to $20°$) and scaling (from 0.9 to 1.1) the mean shape of the training set, and it is fed into the two algorithms.

To quantitatively evaluate the accuracy of the algorithm, we calculate the estimation error by a curve difference measurement. Defining D_k as the distance of one point P_k of the searched shape to its ground true curve as explained in Figure 7(a), the estimation error is calculated as:

$$dist(A)_j = \sum_{k=1}^{N} D_k^A \qquad (21)$$

where $dist(A)_j$ denotes the estimation error of algorithm A on the image j, and N is the number of feature points. Such a curve measurement is more reasonable for the comparison of alignment accuracy, because in many cases the curves are almost the same although the positions of two sets of control points are different.

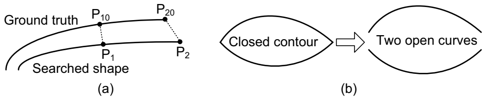


Figure 7. Illustration for shape distance. (a) For the point P_1, D_1 is defined as the minimum distance $|P_1P_{10}|$. For the endpoint P_2, D_2 is defined as the distance between two endpoints $|P_2P_{20}|$. (b) The closed contour is broken into two open curves to calculate the distance.

	< 3pixel	< 5pixel	> 10pixel
SCMN	252(72.0%)	318(90.0%)	5(1.4%)
BTSM	131(37.4%)	239(68.3%)	32(9.1%)

Table 2. The accuracy comparison for the contour. We show the number of samples with maximum alignment errors smaller (or larger) than a threshold.

	< 3pixel	< 5pixel	> 10pixel
SCMN	181(51.7%)	297(84.9%)	2(0.6%)
BTSM	103(29.4%)	243(69.4%)	25(7.1%)

Table 3. The accuracy comparison for the mouth.

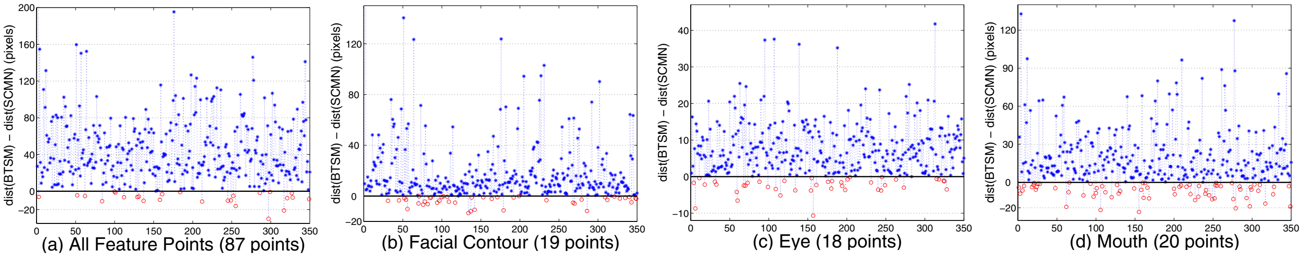
	< 2pixel	< 3pixel	> 5pixel
SCMN	321(91.7%)	346(98.9%)	0(0%)
BTSM	160(45.7%)	304(86.9%)	3(0.9%)

Table 4. The accuracy comparison for the eye.

For the whole face alignment, we have plotted $j \sim dist(BTSM)_j - dist(SCMN)_j$ in Figure 8(a). It is shown that on 324 of 350 (92.6%) images, the search results of our algorithm are better than that of BTSM. Since a human is more sensitive to the alignment accuracy for facial contour, eyes, and mouth, we also compare the accuracy of these three parts respectively. For the facial contour, the eyes, and the mouth, 307(87.7%), 309(88.3%), 279(79.7%) of 350 results of our algorithm are better than that of BTSM. As shown in Figure 8(b), 8(c) and 8(d), the improvement is distinct. Furthermore, for each algorithm, we calculate how many samples' errors (the max distance between two shapes) are within an accuracy threshold or larger than a error tolerance (failure). As shown in Table 2, 3 and 4, our algorithm can improve the alignment accuracy and robustness greatly.

Figure 9 shows a set of searching results of our algorithm and BTSM. In the case that the facial contour or other facial sub-parts is largely variant from the average shape or there are wrinkles and shadings on the face, while by condensation, our algorithm can recover from these local minimas.

While BTSM can localize the whole face well, the results are often not accurate enough especially for the mouth, eyes, and contour, as shown in Figure 10, 11 and 12. Our



(a) All Feature Points (87 points) (b) Facial Contour (19 points) (c) Eye (18 points) (d) Mouth (20 points)

Figure 8. Comparison of the accuracy of our algorithm and BTSM for the whole facial shape (a), facial contour (b), the eyes (c) and the mouth (d), respectively. The x-axis denotes the index j of test images and the y-axis denotes the difference of the estimation errors $dist(BTSM)_j - dist(SCMN)_j$. Points above $y = 0$ (blue stars) denote images with better accuracy by our algorithm and points below $y = 0$ (red circles) are the opposite.

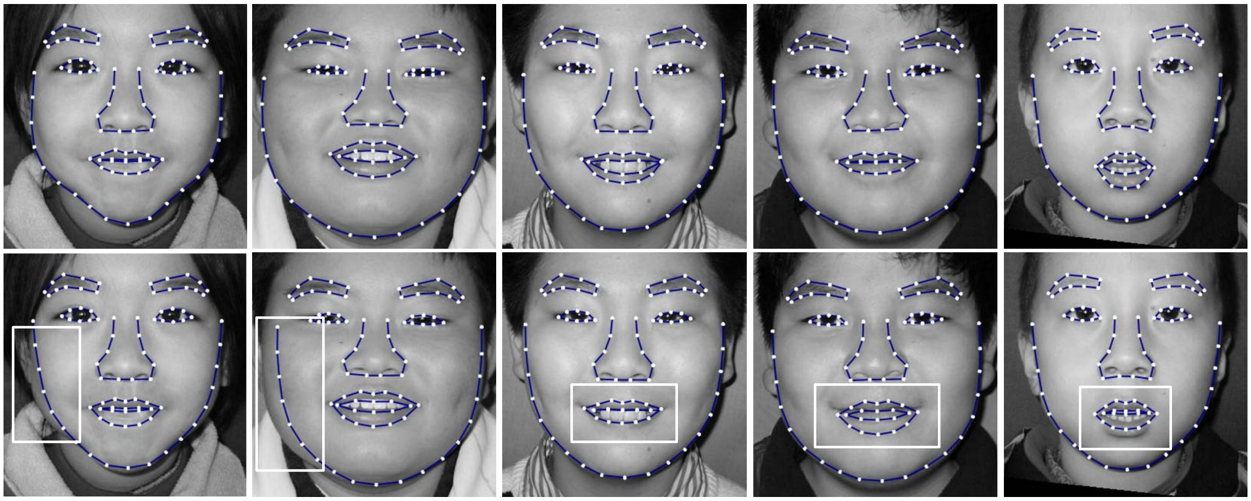


Figure 9. Comparison of our algorithm and BTSM results. The first row is our results, the second row is BTSM results. The white rectangles highlight the regions to be compared.

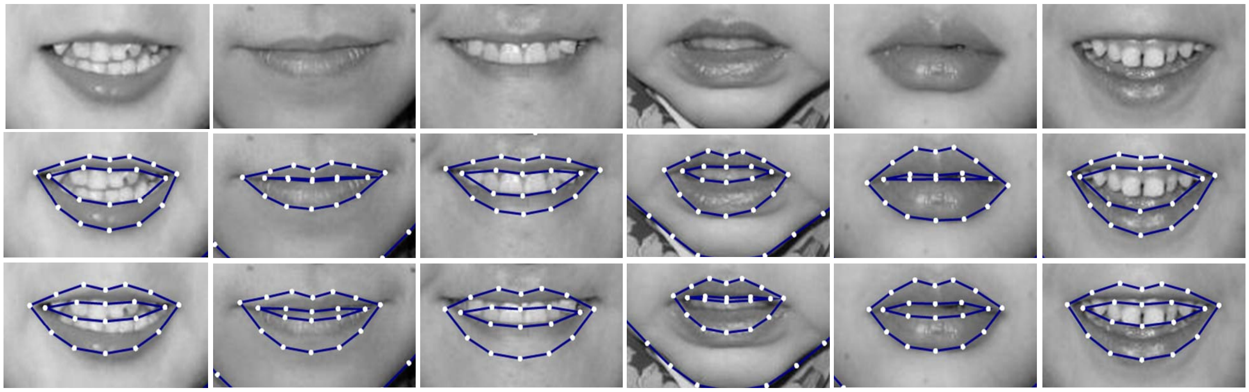


Figure 10. Comparison of our algorithm and BTSM searching results for the mouth part. The first row is the mouth part cut from the test image, the second row is our results, and the third row is BTSM's results.

algorithm can obtain much more accurate results.

Currently, it takes about a few seconds to complete face alignment with our un-optimized research code.

5. Conclusion

In this paper, we have presented a shape constrained Markov network for accurate face alignment.

Some weighted global shapes sampled by the condensation algorithm are added as additional shape structure constraints into the Markov network optimization. This makes the algorithm more robust to local noise. During the condensation stage, by using the Markov network posterior as the proposal density and adopting the constrained regularization, the sampled shapes are more close to the image

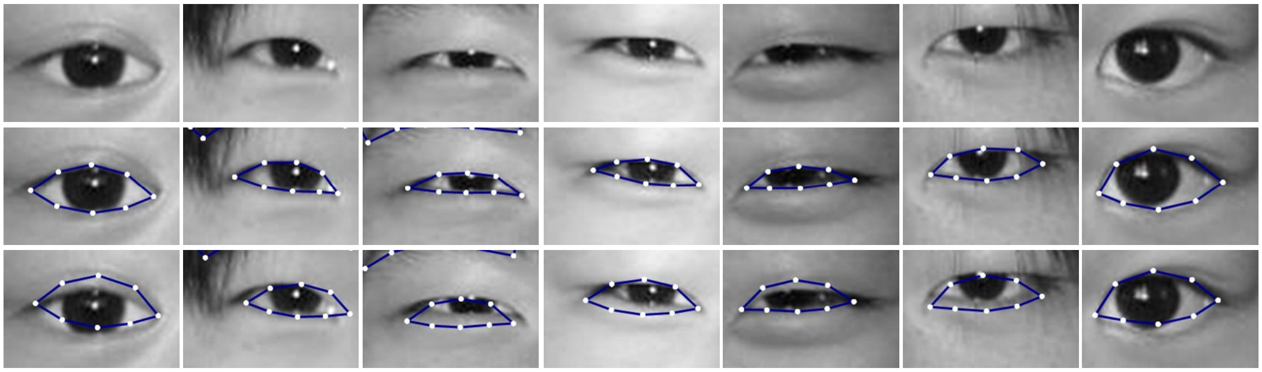


Figure 11. Comparison of our algorithm and BTSM searching results for the eye part. First row is the eye part cut from the test image, the second row is our results, and the third row is BTSM's result.

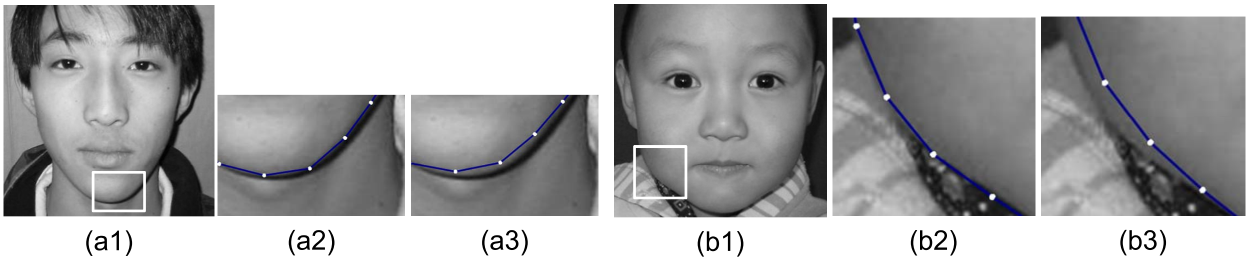


Figure 12. Comparison of our algorithm and BTSM searching results for the facial contour part. (a2) and (b2) are our results, (a3) and (b3) are BTSM's results.

features, thus the added global shapes constraints are more meaningful. We have compared our algorithm with BTSM and demonstrated greatly improved accuracy.

References

[1] T. F. Cootes and C. Taylor. A mixture model for representing shape variation. *Image and Vision Computing*, 17(8):567–574, 1999.

[2] T. F. Cootes, C. J. Taylor, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, 61:38–59, 1995.

[3] J. Coughlan and S. Ferreira. Finding deformable shapes using loopy belief propagation. In *The Seventh European Conference on Computer Vision*, Copenhagen, Denmark, May 2002.

[4] Y. Hu, D. Jiang, S. Yan, L. Zhang, and H. J. Zhang. Automatic 3d reconstruction for face recognition. In *FGR*, Seoul, Korea, May 2004.

[5] M. Kumar, P. Torr, and A.Zisserman. Obj cut. In *CVPR*, San Diego, CA, USA., June 2005.

[6] L. Liang, F. Wen, X. Tang, and Y. Xu. An integrated model for accurate shape alignment. In *ECCV*, volume IV, pages 333–346, Graz, Austria, May. 2006.

[7] C. Liu, H.-Y. Shum, and C. Zhang. Hierarchical shape modeling for automatic face localization. In *Enropean Conf. on Computer Vision*, pages 687–703, 2002.

[8] C. Liu, S.-C. Zhu, and H.-Y. Shum. Learning inhomogeneous gibbs model of faces by minimax entropy. In *IEEE Int'l Conf. on Computer Vision*, Vancouver, Canada, July 2001.

[9] M.Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):5–28, 1998.

[10] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Mateo, California, 1988.

[11] S. Romdhani, S. Cong, and A. Psarrou. A multi-view nonlinear active shape model using kernel pca. In *10th British Machine Vision Conference*, Nottingham, UK, Sept. 1999.

[12] H. H. Thodberg and A. Rosholm. Application of the active shape model in a commercial medical device for bone densitometry. In *BMVC*, Manchester, UK, Sept. 2001.

[13] J. Tu, Z. Zhang, Z. Zeng, and T. Huang. Face localization via hierarchical condensation with fisher boosting feature selection. In *IEEE Conf. on Computer Vision and Pattern Recognition*, Washington, DC, USA, 2004.

[14] L. Wiskott, J.M.Fellous, N.Krüger, and C. der Malsburg. Face recognition by elastic bunch graph matching. In *ICIP*, Santa Barbara, Oct. 1997.

[15] T. Zhang and D. Freedman. Tracking objects using density matching and shape priors. In *ICCV*, Nice, France, Oct. 2003.

[16] Y. Zhou, L. Gu, and H.-J. Zhang. Bayesian tangent shape model: Estimating shape and pose parameters via bayesian inference. In *IEEE Conf. on Computer Vision and Pattern Recognition*, Madison, WI, June 2003.

0-7695-2597-0/06 $20.00 © 2006 IEEE