

# Sequential mean field variational analysis of structured deformable shapes

Gang Hua<sup>\*</sup>, Ying Wu

*Department of Electrical and Computer Engineering, 2145 Sheridan Road, Northwestern University, Evanston, IL 60208, USA*

Received 15 March 2004; accepted 25 July 2005

Available online 3 October 2005

## Abstract

A novel approach is proposed to analyzing and tracking the motion of structured deformable shapes, which consist of multiple correlated deformable subparts. Since this problem is high dimensional in nature, existing methods are plagued either by the inability to capture the detailed local deformation or by the enormous complexity induced by the curse of dimensionality. Taking advantage of the structured constraints of the different deformable subparts, we propose a new statistical representation, i.e., the Markov network, to structured deformable shapes. Then, the deformation of the structured deformable shapes is modelled by a dynamic Markov network which is proven to be very efficient in overcoming the challenges induced by the high dimensionality. Probabilistic variational analysis of this dynamic Markov model reveals a set of fixed point equations, i.e., the sequential mean field equations, which manifest the interactions among the motion posteriors of different deformable subparts. Therefore, we achieve an efficient solution to such a high-dimensional motion analysis problem. Combined with a Monte Carlo strategy, the new algorithm, namely sequential mean field Monte Carlo, achieves very efficient Bayesian inference of the structured deformation with close-to-linear complexity. Extensive experiments on tracking human lips and human faces demonstrate the effectiveness and efficiency of the proposed method.

© 2005 Elsevier Inc. All rights reserved.

*Keywords:* Structured deformable shapes; Dynamic Markov network; Mean field Monte Carlo

## 1. Introduction

Structured deformable shapes consist of multiple correlated deformable subparts. For example, a human face is composed of outer face contour, eyebrows, eyes, nose, and mouth. Analyzing the motion of structured deformable shapes has many real applications such as tracking human lips for speech recognition [1], locating human faces for face recognition [2], and medical applications such as tracking the endocardial wall [3]. The structured deformation is different from articulated motion. In structured deformation, each subpart is a deformable shape while articulation consists of a linked structure of rigid subparts.

For structured deformation analysis, the first problem is how to represent the deformable shapes. Existing methods

either represent the deformable shapes as spline curves [4–7], or as a set of control points [2,8]. The second problem is how to recover the structured deformation from the video sequences. There are mainly two approaches: the top-down approach and the bottom-up approach. The top-down approach takes a two-step strategy, i.e., the hypothesis generation and image observation verification as in the particle filtering algorithm [5–7]. The bottom-up approach estimates the motion parameters by minimizing deterministic cost functions. The SNAKES [4] is the representative using such an approach.

The high-dimensional nature of the structured deformable shapes causes a lot of challenges. Both approaches mentioned above are confronted by the high dimensionality: for the first approach, e.g., the particle filtering algorithm [5–7], the number of particles needed to achieve a good result may increase exponentially with the dimensionality, so does the computation cost; for the second approach, e.g., the SNAKES [4], the cost function

<sup>\*</sup> Corresponding author. Fax: +1 847 491 4455.

E-mail addresses: [ganghua@ece.northwestern.edu](mailto:ganghua@ece.northwestern.edu) (G. Hua), [yingwu@ece.northwestern.edu](mailto:yingwu@ece.northwestern.edu) (Y. Wu).

needs to be optimized in a very high-dimensional space. It is confronted by the enormous local minima induced by the high dimensionality. In addition, since these methods treat the deformable shapes as a whole part, detailed local deformation of a structured deformable shape is hardly to be analyzed.

Our method employs a Markov network to represent the structured deformable shapes. The structural constraints are modelled in the Markov network as potential functions, which are in fact the measures of the probability of two constrained subparts being in certain states of deformation. The structured deformation is then modelled by a dynamic Markov network, which is a temporal extension of the Markov network that models the structured deformation at each time instant. The probabilistic mean field variational analysis of the dynamic Markov network results in a set of mean field fixed point equations. Since it is difficult to obtain the closed form solution to such a set of fixed point equations, a sequential mean field Monte Carlo (SMFMC) [9] algorithm is proposed as a non-parametric approximation. Under this formulation, different subparts are tracked by different trackers while these trackers exchange information with one another to reinforce the structural constraints at the mean time. By way of this, we achieve very efficient Bayesian inference of the posterior deformation with near-to-linear complexity and the local deformation of each subpart can be recovered very well.

The remainder of this paper is organized as follows: in Section 2, related work in the literature is discussed; then in Section 3, we propose the Markov network representation of the structured deformable shapes and the MFMC [9] algorithm which can perform efficient approximate Bayesian inference in such a Markov network; in Section 4, by extending the Markov network and the MFMC algorithm temporally, we propose a dynamic Markov network to model the structured deformation and a SMFMC algorithm to efficiently implement the Bayesian inference in the dynamic Markov network; experimental results are presented and discussed in Section 5; and we present the conclusion and the future work in Section 6.

## 2. Related work

Because of the high-dimensional nature of the motion of the structured deformable shapes, the analysis of them faces many problems. We use a dynamic Markov network representation to approach to the high-dimensional problem and the tracking algorithm is formulated as the Bayesian inference in this graphical model. Therefore, related work in the literature can be categorized into three: the first category addresses the formulation of visual tracking algorithm; the second category refers to the endeavors to conquer the curse-of-dimensionality; and the third category is the Bayesian inference algorithms in complex graphical models.

For visual tracking algorithms, there are two main approaches just as briefly mentioned in Section 1. For the top-down approach, there is a significant literature in the particle filtering tracker [5–7] and its numerous variants [10–14], which basically formulate the visual tracking algorithm as the propagation of the conditional probability of the target state at the current time instant given the image observations up to the current time instant. For the bottom-up approach, the mean-shift blob tracker [15,16], the SNAKES [4], and the efficient region tracker with the parametric model of geometry and illumination [17] are the representatives, to list a few.

To attack the problems caused by high dimensionality, there are mainly two methodologies. The first methodology is to learn the intrinsic lower-dimensional description of the manifold spanned by the valid states in the very high-dimensional state space. To mention some, the active shape model [2] uses PCA to find the best possible low-dimensional linear approximation to the valid shape model state space. The same approach has also been adopted to learn a union of linear manifolds spanned by a set of basis configurations for articulated natural hand motion [18]. While the Bayesian PCA [19,20] provides a more principled way to learn both the intrinsic dimensionality and the statistical mixture model for the valid states. More recent work includes using ISOMAP algorithm for dimension reduction [21]. However, learning a lower-dimensional description of the object needs a set of training data and sometimes it would be very difficult to obtain. On the other hand, the second methodology to attack the curse-of-dimensionality focuses on using more efficient means of computation. Partitioned sampling [22] samples different parts in a hierarchical way and thus achieves linear complexity w.r.t. the number of parts, but the information flow is unidirectional and it violates the symmetric constraints of the different subparts in a structured deformable shape. The MFMC algorithm actually has been used for tracking articulated human body motion [9], which keeps the symmetric constraints between two different articulated links. In this paper, we adopted the same MFMC algorithm to analyze structured deformable shapes, which can keep the symmetric constraints between two different deformable subparts as well as achieve very efficient Bayesian inference with near-to-linear complexity w.r.t. the number of subparts.

For Bayesian inference in graphical model, belief propagation can achieve exact results in the case of directed acyclic graphical model (DAG). When the graphical model has loops, loopy belief propagation [23] or generalized belief propagation (GBP) [24] can obtain good approximate results in many applications [25,26]. Probabilistic variational analysis is a principled approximate inference technique. It uses a more tractable approximate form of the true joint posterior probability [13,20,27–29] and an approximate inference is achieved by minimizing the Kullback–Leibler (KL) divergence between the approximated posterior and the real posterior distribution. Particle filtering or sequential Monte Carlo (SMC) [5–7,30] employs a set of weighted

samples to represent the real distributions and statistical inference is implemented based on these samples. More recent work using particle filtering to perform the inference of general graphical model includes the non-parametric belief propagation (NBP) algorithm [31] and the PAMPAS algorithm [32], where Monte Carlo method is used in the message passing process in belief propagation. In both algorithms, the messages are modelled as Gaussian mixtures and Markov chain Monte Carlo (MCMC) samplers are designed to draw the samples of the components of the Gaussian mixtures for the newly updated messages from the old message distributions. Applications of these probabilistic modelling and inference algorithms for computer vision problems include the part-based object recognition by mixture of trees [33], finding deformable shape by loopy BP [34], tracking 3-D human body motion through a loose-limbed model [35], and the multiple frame articulated motion inference using BP [36], to list a few.

In the proposed method, each subpart of the structured deformable shape is modelled by a parametric description of a spline curve in the motion space. The constraint between each two correlated subparts is modelled explicitly in the dynamic Markov network representation. As described in the following sections, our method of performing inference in such a dynamic Markov network actually combines the idea of variational inference methods [13,20,27–29] with the idea of particle filtering [5–7]. Different from NBP [31] and PAMPAS [32], which combines the BP algorithm with MCMC sampler and all the distributions are modelled as Gaussian mixtures, the proposed SMFMC is a fully non-parametric inference algorithm with close to linear complexity.

### 3. Markov network representation of structured deformable shapes and mean field Monte Carlo

Suppose the structured deformable object consists of  $\mathcal{K}$  subparts, then we denote the state of each subpart of the object as a random variable  $\mathbf{x}_i$ , which can be any parametric description of the motion such as the affine motion in our experiment. Then, we can construct a suitable potential function  $\psi(\mathbf{x}_i, \mathbf{x}_j)$  between two different subparts, which in fact models the probability of two random variables being subject to a certain constraint. In addition, suppose the image observation for each  $\mathbf{x}_i$  is  $\mathbf{z}_i$ , and the observation function is  $\phi(\mathbf{z}_i|\mathbf{x}_i)$ , a Markov network can thus be constructed to model the structured deformable object.

Fig. 1 shows a Markov network of human face where the subscripts ‘Of,’ ‘Leb,’ ‘Reb,’ ‘Le,’ ‘Re,’ ‘N,’ and ‘M’ correspond to outer face contour, left eyebrow, right eyebrow, left eye, right eye, nose, and mouth, respectively. Each undirected link in the graphical model represents a potential function  $\psi(\mathbf{x}_i, \mathbf{x}_j)$  and each directed link in the graphical model represents an observation function  $\phi(\mathbf{z}_i|\mathbf{x}_i)$ .

Denote  $\mathbf{X} = \{\mathbf{x}_i, i = 1 \dots \mathcal{K}\}$  and  $\mathbf{Z} = \{\mathbf{z}_i, i = 1 \dots \mathcal{K}\}$ , the joint probability corresponding to the graphical model in Fig. 1 is

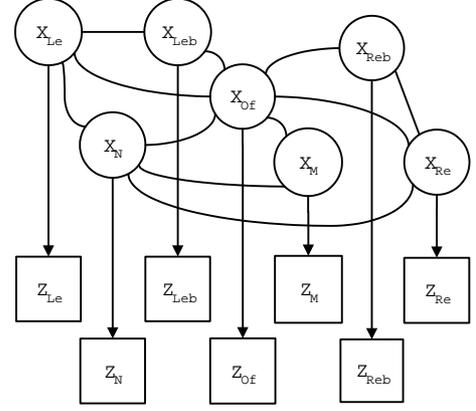


Fig. 1. Markov network for human face.

$$p(\mathbf{X}, \mathbf{Z}) = \frac{1}{Z_C} \prod_{(i,j) \in \mathcal{E}} \psi(\mathbf{x}_i, \mathbf{x}_j) \prod_{i \in \mathcal{V}} \phi(\mathbf{z}_i|\mathbf{x}_i), \quad (1)$$

where  $Z_C$  is a normalization constant,  $\mathcal{E}$  denotes the set of undirected link and  $\mathcal{V}$  denotes the set of directed links in the graphical model.

The inference problem of such a loopy graphical model is to calculate the posteriori probability  $p(\mathbf{x}_i|\mathbf{Z})$ . We employ a variational mean field inference algorithm to obtain an approximate solution to it, i.e., the joint posterior probability is approximated by

$$P(\mathbf{X}|\mathbf{Z}) \approx \prod_i Q_i(\mathbf{x}_i), \quad (2)$$

where  $Q_i(\mathbf{x}_i)$  is an independent approximate distribution of  $P(\mathbf{x}_i|\mathbf{Z})$ . Assuming such a fully factorized mean field approximation which has independent posterior marginal distribution does not mean that the correlation of the group of random variables in the graphical model is in any way lost or ignored, all it means is that any dependencies among this set of random variables  $\mathbf{X}$  remained given the observation  $\mathbf{Z}$  cannot be captured in the mean field approximation [20]. Then, we can construct a cost function

$$J(Q) = \log(P(\mathbf{Z})) - KL \left( \prod_i Q_i(\mathbf{x}_i) \| P(\mathbf{X}|\mathbf{Z}) \right) \quad (3)$$

$$= \log P(\mathbf{Z}) - \int_{\mathbf{X}} \prod_i Q_i(\mathbf{x}_i) \cdot \log \left( \frac{\prod_i Q_i(\mathbf{x}_i)}{P(\mathbf{X}|\mathbf{Z})} \right) d\mathbf{X} \quad (4)$$

$$= - \sum_i H_i(Q_i) + \int_{\mathbf{x}_i} Q_i(\mathbf{x}_i) E_Q \{ \log P(\mathbf{X}, \mathbf{Z}) | \mathbf{x}_i \} d\mathbf{x}_i, \quad (5)$$

where the  $KL(\cdot)$  is the KL divergence and

$$H_i(Q_i) = \sum_{\mathbf{x}_i} Q_i(\mathbf{x}_i) \log Q_i(\mathbf{x}_i) \quad (6)$$

is the entropy of the distribution  $Q_i(\mathbf{x}_i)$  and

$$E_Q \{ \log P(\mathbf{X}, \mathbf{Z}) | \mathbf{x}_i \} = \int_{\{\mathbf{x}_j\} \setminus \mathbf{x}_i} \prod_{\{j\} \setminus i} Q_j(\mathbf{x}_j) \cdot \log [P(\mathbf{X}, \mathbf{Z})] d\mathbf{X}. \quad (7)$$

Maximizing  $J(Q)$  with the constraint that  $\int_{\mathbf{x}_i} Q_i(\mathbf{x}_i) d\mathbf{x}_i = 1$  will lead to  $Q_i(\mathbf{x}_i) \rightarrow P(\mathbf{x}_i|\mathbf{Z})$ . Taking the derivative of  $J(Q)$  w.r.t.  $Q_i(\mathbf{x}_i)$ , and setting it to zero, we obtain that

$$Q_i(\mathbf{x}_i) = e^{E_Q\{\log P(\mathbf{X}, \mathbf{Z})|\mathbf{x}_i\}} \quad (8)$$

In conjunction with the constraint that  $\int_{\mathbf{x}_i} Q_i(\mathbf{x}_i) d\mathbf{x}_i = 1$ , we can solve the equation set to obtain a set of fixed point equations for  $Q_i(\mathbf{x}_i)$  [27,9]:

$$Q_i(\mathbf{x}_i) \leftarrow \frac{1}{Z_\mathcal{E}} e^{E_Q\{\log P(\mathbf{X}, \mathbf{Z})|\mathbf{x}_i\}}, \quad (9)$$

where

$$Z_\mathcal{E} = \int_{\mathbf{x}_i} \exp\{E_Q\{\log P(\mathbf{X}, \mathbf{Z})|\mathbf{x}_i\}\} d\mathbf{x}_i \quad (10)$$

is a normalization constant to constrain that  $Q_i(\mathbf{x}_i)$  is a valid probabilistic distribution. Embedding the factorized joint distribution of the graphical model in Eq. (1) into Eqs. (7) and (9), we can obtain a set of factorized mean field fixed point equations [9]:

$$Q_i(\mathbf{x}_i) \leftarrow \frac{1}{Z'_\mathcal{E}} \phi(\mathbf{z}_i|\mathbf{x}_i) \times \exp \left\{ \sum_{j \in \mathcal{N}(i)} \int_{\mathbf{x}_j} Q_j(\mathbf{x}_j) \log \psi(\mathbf{x}_i, \mathbf{x}_j) d\mathbf{x}_j \right\}, \quad (11)$$

where

$$Z'_\mathcal{E} = \int_{\mathbf{x}_i} \exp \left\{ \sum_{j \in \mathcal{N}(i)} \int_{\mathbf{x}_j} Q_j(\mathbf{x}_j) \log \psi(\mathbf{x}_i, \mathbf{x}_j) \right\} d\mathbf{x}_i \quad (12)$$

is again a normalization constant and  $\mathcal{N}(i)$  denotes the set of subscripts of the neighboring nodes of  $\mathbf{x}_i$ . Since this set of fixed point equations involves integration of complex probabilistic distributions, closed form solution would be difficult. We simulate it by Monte Carlo techniques where several sets of weighted samples are used to represent the probabilistic distributions, i.e.,  $Q_i(\mathbf{x}_i) \sim \{\mathbf{s}_i^{(n)}, \pi_i^{(n)}\}$ . The iteration of the fixed point equation set is then implemented based on these samples. This leads to the MFMC algorithm, which is shown in Fig. 2.

Compared with the NBP algorithm [31] and the PAMPAS algorithm [32], the MFMC algorithm employs weighted samples for all the distributions while both NBP and PAMPAS model all the distributions as Gaussian mixtures. In this sense, our algorithm is fully non-parametric while theirs are semi-parametric. Also, our algorithm employs importance sampling thus it avoids using slow MCMC samplers which have been used in both NBP and PAMPAS. Although how to determine the best importance function is still an art more than science, the use of importance sampling has two advantages: first, it avoids sampling directly from the complex Gibbs distribution of Eq. (9) or Eq. (11); second, we can easily integrate both our domain knowledge and heuristics into the importance function to generate samples from more confident areas. A natural choice of the importance function would be the potential function  $\psi(\mathbf{x}_i, \mathbf{x}_j)$  since it is like a conditional probability if one of the  $\mathbf{x}_i$  and  $\mathbf{x}_j$  is fixed. In our experiments in Section 5, we indeed adopted the potential function as the importance function,

Generate  $\{s_{i,k+1}^{(n)}, \omega_{i,k+1}^{(n)}\}_{n=1}^N$  from  $\{s_{i,k}^{(n)}, \omega_{i,k}^{(n)}\}_{n=1}^N$ .

(1) **Importance Sampling:** Sample  $\{s_{i,k+1}^{(n)}, \frac{1}{n}\}$  from a suitable importance function  $I_i(x_i)$ .

(2) **Re-weight:** For each sample  $s_{i,k+1}^m$ , set its weight to

$$\omega_{i,k+1}^m = \phi(z_i|s_{i,k+1}^m) G_{x_i}(s_{i,k+1}^m) / I_i(s_{i,k+1}^m)$$

where

$$G_{x_i}(s_{i,k+1}^m) = e^{\sum_{j \in \mathcal{N}(i)} \sum_{n=1}^N \omega_{j,k}^{(n)} \log \psi(s_{i,k+1}^m, s_{j,k}^{(n)})}$$

(3) **Normalization:**

$$\omega_{i,k+1}^m = \omega_{i,k+1}^m / \sum_m \omega_{i,k+1}^m,$$

then we get  $\{s_{i,k+1}^{(n)}, \omega_{i,k+1}^{(n)}\}$

(4) **Iteration:**  $k \leftarrow k + 1$ , iterate until convergence.

Fig. 2. Mean field Monte Carlo algorithm.

i.e., if one subpart is connected with several other subparts, we mix the samples from each potential function together.

#### 4. Modelling structured deformation by dynamic Markov network and sequential MFMC

Assuming each subpart of the structured deformable shape has an independent dynamic motion model  $p(\mathbf{x}_{i,t}|\mathbf{x}_{i,t-1})$ , we can describe the motion of a structured deformable object through a dynamic Markov network as in Fig. 3. It is the temporal extension of the Markov network in Fig. 1. Each horizontal arrow represents the dynamic motion model of the corresponding subpart. Denote  $\mathbf{X}_t = \{\mathbf{x}_{i,t}, i = 1 \dots \mathcal{K}\}$ ,  $\mathbf{Z}_t = \{\mathbf{z}_{i,t}, i = 1 \dots \mathcal{K}\}$ , and  $\mathbf{Z}_{1:t} = \{\mathbf{Z}_1, \dots, \mathbf{Z}_t\}$ , we can easily figure out that:

$$P(\mathbf{X}_t, \mathbf{Z}_{1:t}) \propto \prod_{(i,j) \in \mathcal{E}_t} \psi(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) \prod_{i \in \mathcal{V}_t} \phi(\mathbf{z}_{i,t}|\mathbf{x}_{i,t}) \times \int_{\mathbf{X}_{t-1}} P(\mathbf{X}_t|\mathbf{X}_{t-1})P(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) d\mathbf{X}_{t-1}. \quad (13)$$

Here, the inference problem is to recover  $P(\mathbf{x}_{i,t}|\mathbf{Z}_{1:t})$ ,  $i = 1 \dots \mathcal{K}$  at time instant  $t$  based on the inference result  $P(\mathbf{x}_{i,t-1}|\mathbf{Z}_{1:t})$ ,  $i = 1 \dots \mathcal{K}$  at time instant  $t-1$ . Under the context of mean field variational analysis, suppose we already have

$$P(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) \approx \prod_i Q_{i,t-1}(\mathbf{x}_{i,t-1}) \quad (14)$$

and then we need to find the best  $\prod_i Q_{i,t}(\mathbf{x}_{i,t})$  to approximate the real  $P(\mathbf{X}_t|\mathbf{Z}_{1:t})$ . Assuming independent dynamic model for each subpart, i.e.,

$$P(\mathbf{X}_t|\mathbf{X}_{t-1}) = \prod_i P(\mathbf{x}_{i,t}|\mathbf{x}_{i,t-1}). \quad (15)$$

Embedding Eqs. (14) and (15) into Eq. (13), we have

$$P(\mathbf{X}_t, \mathbf{Z}_{1:t}) \propto \prod_{(i,j) \in \mathcal{E}_t} \psi(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) \prod_{i \in \mathcal{V}_t} \phi(\mathbf{z}_{i,t}|\mathbf{x}_{i,t}) \times \int_{\mathbf{X}_{t-1}} \prod_i P(\mathbf{x}_{i,t}|\mathbf{x}_{i,t-1}) \prod_i Q_{i,t-1}(\mathbf{x}_{i,t-1}) d\mathbf{X}_{t-1} \quad (16)$$

$$\propto \prod_{(i,j) \in \mathcal{E}_t} \psi(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) \times \prod_{i \in \mathcal{V}_t} \left[ \phi(\mathbf{z}_{i,t}|\mathbf{x}_{i,t}) \int_{\mathbf{x}_{i,t-1}} P(\mathbf{x}_{i,t}|\mathbf{x}_{i,t-1}) Q_{i,t-1}(\mathbf{x}_{i,t-1}) d\mathbf{x}_{i,t-1} \right]. \quad (17)$$

The mean field approximation of  $P(\mathbf{X}_t|\mathbf{Z}_{1:t})$  is

$$P(\mathbf{X}_t|\mathbf{Z}_{1:t}) \approx \prod_i Q_{i,t}(\mathbf{x}_{i,t}). \quad (18)$$

Then, following almost the same steps in Section 3, we first construct the cost function  $J_t(Q)$  as in Eq. (21):

$$J_t(Q) = \log(P(\mathbf{Z}_{1:t})) - KL\left(\prod_i Q_{i,t}(\mathbf{x}_{i,t}) \| P(\mathbf{X}_t|\mathbf{Z}_{1:t})\right) \quad (19)$$

$$= \log P(\mathbf{Z}_{1:t}) - \int_{\mathbf{X}_t} \prod_i Q_{i,t}(\mathbf{x}_{i,t}) \cdot \log\left(\frac{\prod_i Q_{i,t}(\mathbf{x}_{i,t})}{P(\mathbf{X}_t|\mathbf{Z}_{1:t})}\right) d\mathbf{X}_t \quad (20)$$

$$= - \sum_i H_{i,t}(Q_{i,t}) + \int_{\mathbf{x}_{i,t}} Q_{i,t}(\mathbf{x}_{i,t}) E_Q\{\log P(\mathbf{X}_t, \mathbf{Z}_{1:t})|\mathbf{x}_{i,t}\} d\mathbf{x}_{i,t}, \quad (21)$$

where

$$H_{i,t}(Q_{i,t}) = \sum_{\mathbf{x}_{i,t}} Q_{i,t}(\mathbf{x}_{i,t}) \log [Q_{i,t}(\mathbf{x}_{i,t})] \quad (22)$$

is the entropy of the distribution  $Q_{i,t}(\mathbf{x}_{i,t})$  and

$$E_Q\{\log P(\mathbf{X}_t, \mathbf{Z}_{1:t})|\mathbf{x}_{i,t}\} = \int_{\{\mathbf{x}_{j,t}\}|\mathbf{x}_{i,t}} \prod_{(j) \neq i} Q_{j,t}(\mathbf{x}_{j,t}) \cdot \log P(\mathbf{X}_t, \mathbf{Z}_{1:t}) d\mathbf{X}_t. \quad (23)$$

Taking the derivative of  $J_t(Q)$  w.r.t.  $Q_{i,t}(\mathbf{x}_{i,t})$  and setting it to zero, with the constraint that  $\int_{\mathbf{x}_{i,t}} Q_{i,t}(\mathbf{x}_{i,t}) d\mathbf{x}_{i,t} = 1$ , we can solve the equation set to get the following fixed point equation:

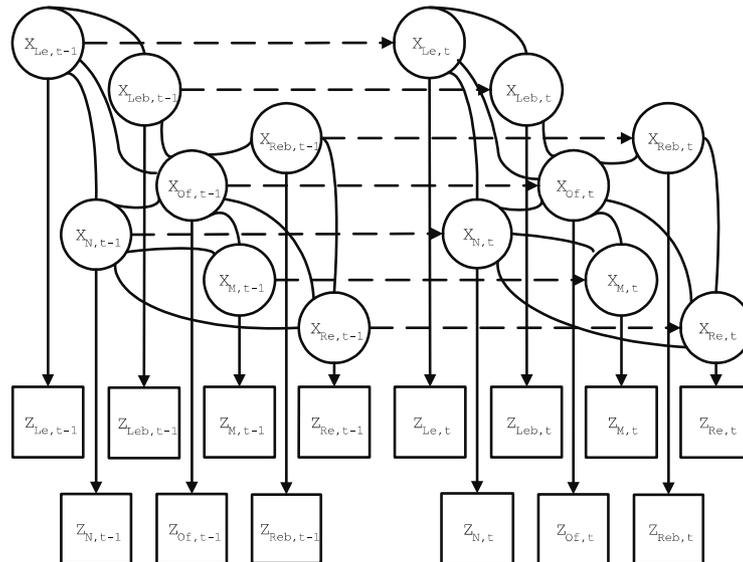


Fig. 3. DMN for human face deformation.

$$Q_{i,t}(\mathbf{x}_{i,t}) \leftarrow \frac{1}{Z_{\mathcal{E},t}} \mathbf{e}^{E_Q\{\log P(\mathbf{X}_i, \mathbf{Z}_{1:t} | \mathbf{x}_{i,t})\}}, \quad (24)$$

where  $Z_{\mathcal{E},t}$  is a normalization constant. By embedding Eq. (13) into Eq. (24), we obtain the following fixed point equation:

$$Q_{i,t}(\mathbf{x}_{i,t}) \leftarrow \frac{1}{Z_C} \phi(\mathbf{z}_{i,t} | \mathbf{x}_{i,t}) \times \int_{\mathbf{x}_{i,t-1}} P(\mathbf{x}_{i,t} | \mathbf{x}_{i,t-1}) Q_{i,t-1}(\mathbf{x}_{i,t-1}) d\mathbf{x}_{i,t-1} \times \exp \left( \sum_{j \in \mathcal{N}(i)} Q_{j,t}(\mathbf{x}_{j,t}) \log(\psi(\mathbf{x}_{i,t}, \mathbf{x}_{j,t})) \right), \quad (25)$$

where  $Z_C$  is a normalization constant which can make sure that  $Q_{i,t}(\mathbf{x}_{i,t})$  is a valid probabilistic distribution. Therefore, by using the mean field variational analysis, we actually approximate the propagation of the conditional posterior probability of  $P(\mathbf{X}_i | \mathbf{Z}_{1:t})$  [5–7] to the propagation of the variational mean field distribution  $\prod_i Q_{i,t}(\mathbf{x}_{i,t})$ . This is the sequential version of the mean field variational method, which we call sequential mean field variational analysis.

Comparing Eq. (25) with Eq. (11), we can see that there is one more item in Eq. (25), which exactly models the dynamic prediction prior. If we do know the priori probability of each node in Fig. 1, we can also incorporate a local prior item in Eqs. (9) and (11).

This set of fixed point equations is more complex because they involve more multiple integrations of complex probabilistic distributions. Again, we adopt a Monte Carlo strategy to implement it. We call it the SMFMC algorithm, as shown in Fig. 4.

The SMFMC algorithm has two steps, the first step is the SMC process; then, the result is used as the initialization of the MFMC process. The MFMC process runs until convergence. One question here would be the choice of the importance function, here we can construct a suitable importance function through our domain knowledge and valid heuristics, e.g., a more accurate dynamic model of the structured deformation, or a rough detection result of the different subparts of the structured deformable shape, or a learned conditional probability of one subpart given one of the connected subparts, which in fact could be from the learned potential function  $\psi(\mathbf{x}_i, \mathbf{x}_j)$ , etc.

## 5. Experiments and discussion

To demonstrate both the effectiveness and the efficiency of the SMFMC algorithm, it has been implemented to track both human lips and human faces. All the experiments run on a PC with 2.4 GHz CPU. The code is programmed with C++ and no code optimization is performed.

### 5.1. Tracking human lips

#### 5.1.1. Potential function and observation likelihood for lip tracking

A human lip can be decomposed into upper-lip and lower-lip, each of them is represented by an affine deformation of a spline curve. Thus, a two-node Markov network can be constructed and each node represents a six-dimensional random variable of the affine deformation. The constraint between the upper-lip and lower-lip is that the two pairs of the end points should be as close as possible, thus a potential function can be constructed based on this. Denote  $\mathbf{x}_u$  and  $\mathbf{x}_l$  as the affine motion of the upper-lip and lower-lip,  $\mathbf{Pt}_1^u$  and  $\mathbf{Pt}_2^u$  as the two end points of the upper-lip curve, and  $\mathbf{Pt}_1^l$  and  $\mathbf{Pt}_2^l$  as the two end points of the lower-lip curve, we adopt a Gaussian potential function to model the constraint, i.e.,

$$\psi(\mathbf{x}_u, \mathbf{x}_l) = \frac{1}{\sqrt{2\pi\sigma}} \times \exp \left( -\frac{\|\mathbf{Pt}_1^u - \mathbf{Pt}_1^l\|^2 + \|\mathbf{Pt}_2^u - \mathbf{Pt}_2^l\|^2}{2\sigma^2} \right). \quad (26)$$

Since it is a zero mean Gaussian distribution, the only parameter which needs to be estimated is  $\sigma^2$ . It can be easily trained from a set of manually labelled lip images by the maximum likelihood (ML) estimation. Nevertheless, our experiments show that this parameter is not that sensitive, it can be set in a large range around the trained one. Therefore, another choice to save the tedious manually labelling work is to manually set the parameter to see whether it can achieve good empirical results or not.

The observation function  $\phi(\mathbf{z}_s | \mathbf{x}_s)$ ,  $s \in \{u, l\}$  is similar as that in [14]. We follow the idea of using a set of  $n$  normal measurement lines of length  $L$  to collect image features [7,5]. For the  $i$ th measurement line,  $i = 1, \dots, n$ , we denote the contour point position by  $x_i$ . A 1-D edge detection is performed along the measurement line to find all the  $m_i$  locations of the edge points  $\{z_i^1, \dots, z_i^{m_i}\}$ . It is easy to observe that only one of the  $\{z_i^k\}$  could be produced by the object contour and all the other edge points should be produced by the cluttered background.

Therefore, we assume a Gaussian distribution of the location of the edge point produced by the object boundary, which could only be one of the  $m_i(W)$  edge points inside a window  $W < L$  around  $x_i$ , and a poisson distribution with spatial arrival rate  $\lambda$  on the number of edge points produced by the cluttered background. Then, the likelihood  $p(\mathbf{z}_{s,i} | x_i)$  is calculated as

$$p(\mathbf{z}_{s,i} | x_i) = \alpha \mathcal{N} \left( \frac{W}{2} | 0, \sigma_0^2 \right) \frac{\lambda L^{m_i}}{m_i!} e^{-\lambda L} + (1 - \alpha) \times \frac{\sum_{z_i^k \in W} \mathcal{N}(z_i^k | x_i, \sigma_0^2)}{m(W)} \frac{\lambda L^{m_i-1}}{(m_i-1)!} e^{-\lambda L}, \quad (27)$$

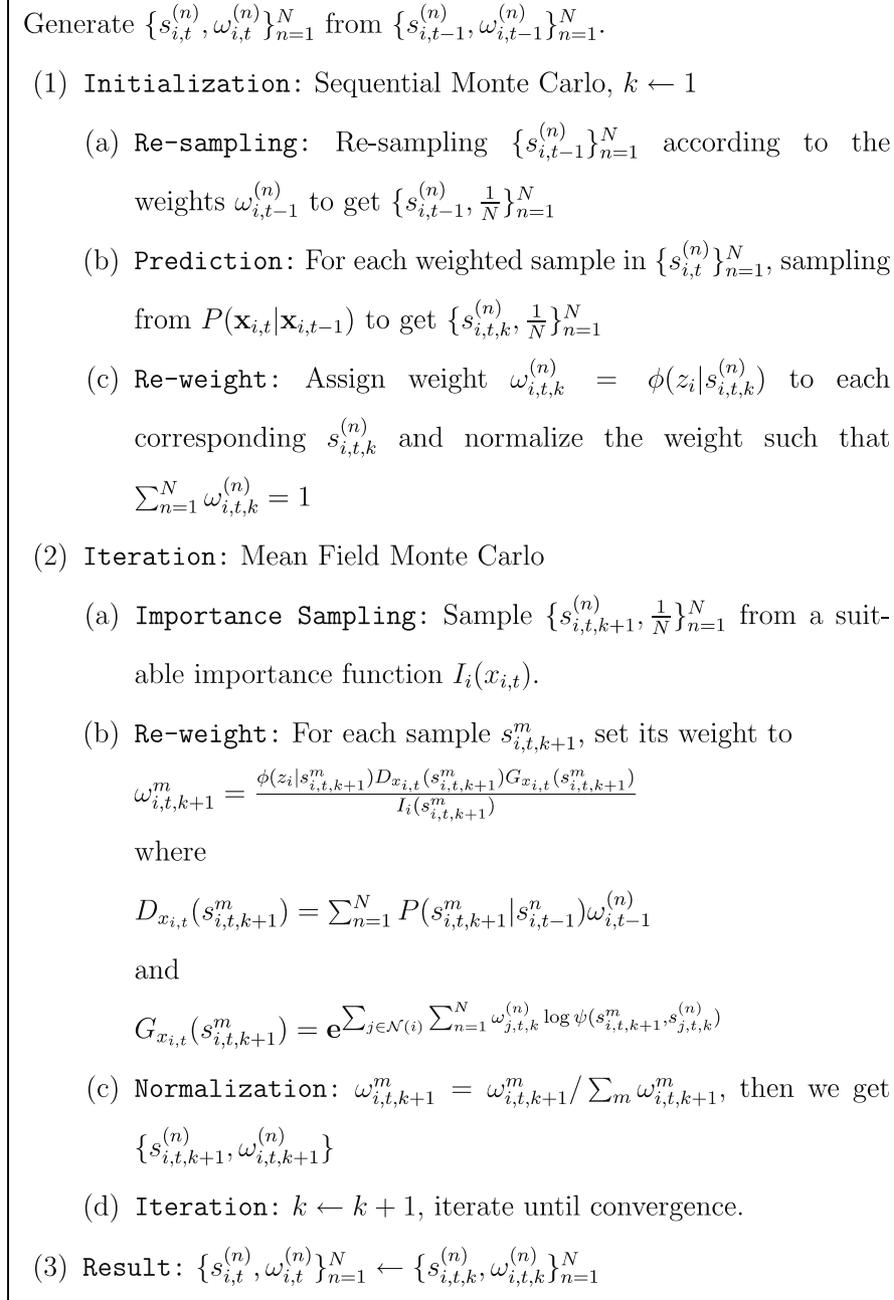


Fig. 4. Sequential mean field Monte Carlo algorithm.

where  $\alpha$  is the probability of missing detection of the edge point produced by the object boundary. Introducing such a probability may increase the robustness of the observation likelihood function [14,7,5].

Then, the full observation likelihood is simply the product of the likelihood of all the measurement lines, i.e.,

$$\phi(\mathbf{z}_s|\mathbf{x}_s) = \prod_{i=1}^n p(\mathbf{z}_{s,i}|x_i), s \in \{u, l\}. \quad (28)$$

We refer the reader to [14] for a more detailed discussion of the observation likelihood function used here. The same

observation likelihood function is also adopted in the face tracking experiment in Section 5.2.

### 5.1.2. Convergence of the MFMC iterations

To demonstrate the convergence of the proposed MFMC algorithm, we manually initialize the MFMC algorithm from different initial points on the first frame of the lip video. Our experimental results show that the MFMC algorithm usually converges into the correct results after five iterations, even with initializations which are significantly deviated from the real result, as shown in Fig. 5.



Fig. 5. Convergence of the MFMC iterations: A.(a)–A.(g), the convergence of the iteration under initialization A; B.(a)–B.(g), the convergence of the iteration under initialization B.

Fig. 5 presents the convergence of the MFMC iterations under two initializations. The first two rows and the last two rows show the convergence of the MFMC iterations under initialization A and initialization B, respectively. As we can observe in the figure, both initializations are largely deviated from the real locations of the lip but the MFMC iteration does converge to the correct result in five iterations. We extensively run the experiments many times and obtained similar results as shown in Fig. 5.

However, the convergence of the MFMC algorithm does depend on the initialization, i.e., with a better initialization in the sense that it is nearer to the real result, the convergence of the MFMC iteration is faster, and vice versa. Too wild initialization may result in the failure of the MFMC iteration to converge in a reasonable times of iteration, e.g., for the initialization shown in Fig. 6. Since dur-

ing tracking, we usually try to manually initialize the tracker as accurate as possible, the MFMC iteration usually needs less iterations to converge. We generally set the number of iterations to three during the tracking process using the SMFMC algorithm.



Fig. 6. The MFMC algorithm failed to converge under this kind of wild initialization.

### 5.1.3. Lip tracking by SMFMC

The lip video sequence has more than 150 frames. Sample frames of the results of the SMFMC algorithm are shown in Fig. 7.

For comparison, a single SMC tracker and a multiple independent SMC (MiSMC) tracker are also implemented. The results are shown in Figs. 8 and 9, respectively. From the results, we observe that the single SMC tracker cannot catch the detailed local deformation well. The reason is that the upper-lip and lower-lip are actually subject to different correlated affine motion instead of the same motion, but the single SMC algorithm treats them as the same. We also observe that the MiSMC cannot keep the structure of the upper- and lower-lip, and it easily loses track because of the lack of constraints between the two independent trackers. On the contrary, the proposed SMFMC algorithm achieves much better results. It tracks both the upper-lip and lower-lip deformation very well, and the structure of the combined upper-lip and lower-lip has also been kept because it has been reinforced by the dynamic Markov network.

### 5.1.4. A unified view of SMFMC, MiSMC, and single SMC

In fact, if a very tight potential function is used in the Markov network, i.e., the state of one node can fully decide the state of the other, then the SMFMC tracker degenerates to a single SMC tracker. On the contrary, if the potential function is very loose, i.e., each node is almost independent with the others, then the SMFMC tracker degenerates to a MiSMC tracker. Thus, we can treat SMC and MiSMC as two extreme cases of the SMFMC tracker, as showing in Fig. 10.

## 5.2. Face alignment and tracking

### 5.2.1. Potential function and observation likelihood for face

Just as we have shown in Fig. 1, the face contour are modelled by a seven-node Markov network. It is obvious that two subparts on the face are constrained by the rigidity of the human face, e.g., the relative location of the eye and the nose can only be changed in certain range. Denote the centroid point of the subpart curve  $i$  as  $\mathbf{Pt}_i$  and the centroid point of the subpart curve  $j$  as  $\mathbf{Pt}_j$ , then the potential



Fig. 7. Tracking human lip via SMFMC.



Fig. 8. Tracking human lip via a single SMC tracker.



Fig. 9. Tracking human lip via a MiSMC tracker.

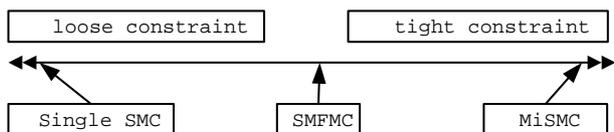


Fig. 10. The relationship of SMFMC, SMC, and MiSMC.

function between these two is defined by a Gaussian distribution as

$$\psi(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2\pi|\Sigma_{ij}|} \times \exp \left\{ \left( -\frac{1}{2}(\mathbf{P}\mathbf{t}_i - \mathbf{P}\mathbf{t}_j - \bar{\mu}_{ij})^\top \Sigma_{ij}^{-1} (\mathbf{P}\mathbf{t}_i - \mathbf{P}\mathbf{t}_j - \bar{\mu}_{ij}) \right) \right\}, \quad (29)$$

where  $\bar{\mu}_{ij}$  and  $\Sigma_{ij}$  are the two-dimensional mean vector and 2 by 2 covariance matrix of the relative translation between two sub-parts. We easily obtained the ML estimation of the  $\bar{\mu}_{ij}$  and  $\Sigma_{ij}$  from a set of manually annotated face images. The trained potential functions are then adopted to perform the face tracking. Just as we have stated, the observation likelihood function is the same as that in Section 5.1.1.

### 5.2.2. Face tracking by SMFMC

The proposed SMFMC algorithm achieves good results for automatic face alignment and face tracking over the video sequence of more than 200 frames. After manual initialization in the first frame, the algorithm automatically aligns and tracks the face over the sequence. Sample frames of the experimental result are shown in Fig. 11. Compared with the ASM model [2], our face prior model is actually a Gibbs model while the ASM uses a Gaussian model. Thus, our distributed model is more general than the ASM model, and more capable of capturing detailed shape deformation.

Just as we have mentioned in Section 4, the first step of the SMFMC algorithm is the SMC step, then the result is used as the initialization of the MFMC step. In most cases, the result of the SMC step is unsatisfactory, then the MFMC step will use the structure information of the face to guide an iterative alignment to a better result.

### 5.2.3. Quantitative analysis

By manual labelling the tracked face video sequence, quantitative evaluation of the proposed SMFMC algorithm was performed. Since during the SMFMC tracking process, the affine parameters are estimated with respect to the manually initialized B-spline curve in the first frame, we are facing the problem of recovering the ground-truth affine motion from the manually labelled video sequence. Denote the labelled curve at the first frame as  $\mathbf{x}_1$  and the labelled curve at time instant  $t$  as  $\mathbf{x}_t$ , this could be achieved to solve a least square fitting problem, i.e.,

$$\mathbf{A}_t^* = \min_{\mathbf{A}_t} \|\mathbf{A}_t(\mathbf{x}_1) - \mathbf{x}_t\|^2, \quad (30)$$

where  $\mathbf{A}_t$  denotes the affine motion,  $\mathbf{A}_t^*$  denotes the estimated “ground-truth” affine motion, and  $\|\cdot\|$  denotes the Euclidean distance.

However, solving the above least square fitting problem involves the evaluation of the squared residue error on certain selected points on the two curves. And using the fitted affine motion as the ground-truth may introduce more numerical errors. Therefore, instead we take a direct evaluation approach by using the recovered affine motion from the motion tracking algorithm to transform the reference curve in the first frame to the current time instant. We can then evaluate the root mean square error (RMSE) between the transformed curve and ground-truth labelled curve on certain selected points. The selected points are sampled evenly from the spline curve. For example, denote the evenly sampled  $n$  points on the reference curve as  $\{s_1, \dots, s_n\}$ , the  $n$  points on the ground-truth curve at the current time instant as  $\{s'_1, \dots, s'_n\}$ , and the recovered affine motion as  $\mathbf{A}_t$ , then the RMSE is evaluated as

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_i \|\mathbf{A}_t(s_i) - s'_i\|^2}, \quad (31)$$

where  $\|\cdot\|$  is again the Euclidean distance. Our face model contains sever subparts. Each of them has its own curve

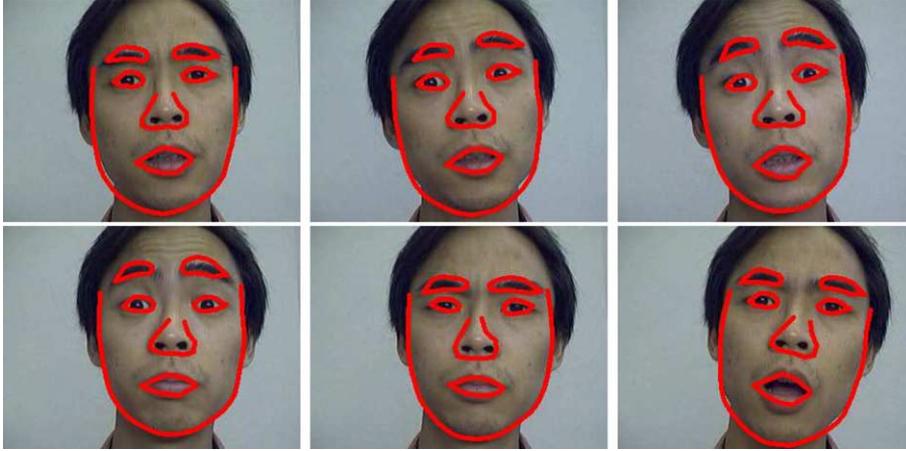


Fig. 11. Tracking human face via sequential MFMC.

representation and individual but correlated affine motion. We evenly sampled  $n_i$  points from the curve of different subparts. Then, the overall RMSE is calculated as

$$\text{RMSE} = \sqrt{\frac{1}{7} \sum_{j=1}^7 \frac{1}{n_i} \sum_i^{n_i} \|\mathbf{A}_i^j(s_i^j) - s_i^{j,t}\|^2}. \quad (32)$$

The number of points evenly sampled from each subpart curve is proportional to the length of the curve, e.g., 30 points are sampled for face outer contour, 20 points are sampled for mouth contour, and 12 points are sampled for the other subpart. We show the overall RMSE of the SMFMC tracker, the MiSMC tracker, and the single SMC tracker on the tested face video sequence on Fig. 12. We can observe that the proposed SMFMC tracking algorithm generally achieved better accuracy than the

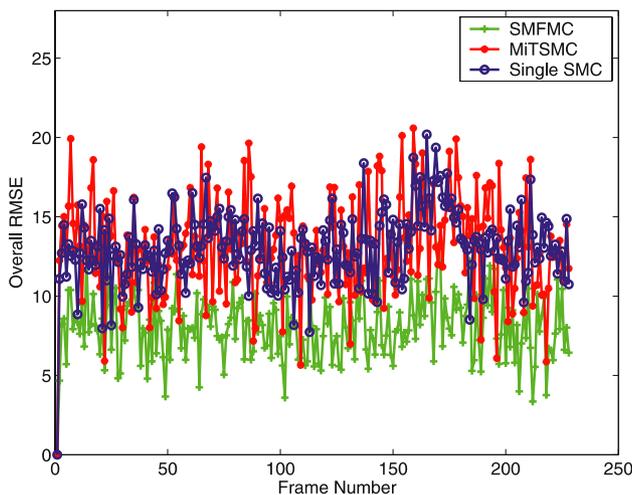


Fig. 12. The overall RMSE of the SMFMC, the SMC, and the MiSMC on the test face video sequence. The green line, the red line, and the blue line represent the overall RMSEs of the SMFMC algorithm, the MiTSMC algorithm, and the single SMC algorithm, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

MiSMC tracking algorithm and the single SMC tracking algorithm on the tested video sequence.

### 5.3. Computation efficiency

In the SMC algorithm, the major computation comes from the evaluation of the observation likelihood. Since in each time step, the SMC algorithm needs to evaluate the observation likelihood once for each sample, the computation complexity is almost proportional to the number of samples. For conventional particle filtering methods, the needed number of samples is exponential increasing w.r.t. the dimensionality. Suppose for a single object, the number of samples needed to make the tracker work is  $\mathcal{N}$ . Then, for a structured deformable shape with  $\mathcal{K}$  subparts, if mean field iteration converges in  $M$  steps, the computation complexity is  $O(M\mathcal{K}\mathcal{N})$  [9], which is clearly linear w.r.t. the number of subparts, thus to the dimensionality.

Moreover, our algorithm can let us allocate the computation resources to different subparts according to their needs, i.e., some of the more complex subparts may need more samples while some of the others may need less samples. In this way, the computation resources can be used more efficiently as we have done in the experiments. Table 1 shows the rough processing frame rates of different methods in our experiments. Notice that the lip has two subparts, the face has seven subparts and the number of mean field iteration is three, it verifies that the SMFMC tracker does have linear complexity w.r.t. the dimensionality.

## 6. Conclusion and discussion

In this paper, a novel SMFMC algorithm is proposed to analyze and track structured deformable shapes based on a

Table 1  
Processing frame rate of different trackers

Algorithms:	SMCLip	MiSMCLip	SMFMCCLip	SMFMCFace
Frame rate (F/s):	16.5	10.9	4.0	1.6

dynamic Markov network representation. It approximates the propagation of the posterior motion of the structured deformable shapes to the propagation of the variational mean field distributions. The SMFMC algorithm combines both analytical and non-parametric inference methods, i.e., the probabilistic mean field variational analysis and Monte Carlo methods. More interestingly, it has linear complexity w.r.t. the dimensionality. In addition, it shows that the single SMC tracker and the MiSMC tracker can be regarded as two extreme cases of our algorithm. Experiments have demonstrated the effectiveness and efficiency of the proposed SMFMC algorithm.

Although the SMFMC algorithm is employed in this paper for analyzing structured deformable shapes, it is actually generalizable for many other vision problems such as articulated motion analysis [9]. The idea of dealing with the local subparts while reinforcing the global constraint proved to be an effective way to overcome the curse-of-dimensionality. Our future work includes applying SMFMC to other vision problems and more thorough theoretical study of the SMFMC such as the theoretic analysis of the convergence rate.

## Acknowledgments

This work was supported in part by National Science Foundation Grant IIS-0347877, IIS-0308222, Northwestern faculty startup funds for Ying Wu and Walter P. Murphy Fellowship for Gang Hua.

## References

- [1] R. Kaucic, B. Dalton, A. Blake, Real-time lip tracking for audio-visual speech recognition applications, in: Proc. Eur. Conf. on Computer Vision, vol. 2, 1996, pp. 376–387.
- [2] T. Cootes, D. Cooper, C. Taylor, J. Graham, Active shape models—their training and application, *Comput. Vis. Image Understand.* 61 (1) (1995) 38–59.
- [3] D. Comaniciu, Nonparametric information fusion for motion estimation, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2003.
- [4] M. Kass, A. Witkin, D. Terzopoulos, SNAKES: active contour models, *Int. J. Comput. Vis.* (1988) 321–331.
- [5] M. Isard, A. Blake, Contour tracking by stochastic propagation of conditional density, in: Proc. Eur. Conf. on Computer Vision, vol. 1, 1996, pp. 343–356.
- [6] M. Isard, A. Blake, CONDENSATION—conditional density propagation for visual tracking, *Int. J. Comput. Vis.* 29 (1) (1998) 5–28.
- [7] A. Blake, M. Isard, *Active Contours*, Springer-Verlag, Berlin, 1998.
- [8] C. Liu, H.-Y. Shum, C. Zhang, Hierarchical shape modeling for automatic face localization, in: Proc. Eur. Conf. on Computer Vision, 2002, pp. 687–703.
- [9] Y. Wu, G. Hua, T. Yu, Tracking articulated body by dynamic Markov network, in: Proc. IEEE Int. Conf. on Computer Vision, 2003, pp. 1094–1101.
- [10] M. Isard, A. Blake, Icondensation: unifying low-level and high-level tracking in a stochastic framework, in: Proc. Eur. Conf. on Computer Vision, 1998.
- [11] M. Isard, A. Blake, A smoothing filter for condensation, in: Proc. Eur. Conf. on Computer Vision, 1998.
- [12] M. Isard, A. Blake, A mixed-state condensation tracker with automatic model-switching, in: Proc. 6th Internat. Conf. on Computer Vision, 1998, pp. 107–112.
- [13] Y. Wu, T.S. Huang, Robust visual tracking by co-inference learning, in: Proc. IEEE Internat. Conf. on Computer Vision, vol. II, 2001, pp. 26–33.
- [14] Y. Wu, G. Hua, T. Yu, Switching observation models for contour tracking in clutter, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 1, 2003, pp. 295–302.
- [15] D. Comaniciu, V. Ramesh, P. Meer, Real-time tracking of non-rigid objects using mean shift, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, Hilton Head Island, South Carolina, 2000, pp. 142–149.
- [16] R.T. Collins, Mean shift blob tracking through scale space, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, Madison, WI, 2003, pp. 234–240.
- [17] G. Hager, P. Belhumeur, Efficient region tracking with parametric models of geometry and illumination, *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (10) (1998) 1025–1039.
- [18] Y. Wu, J. Lin, T.S. Huang, Capturing natural hand articulation, in: Proc. IEEE Internat. Conf. on Computer Vision, vol. II, 2001, pp. 426–432.
- [19] C.M. Bishop, J.M. Winn, Non-linear bayesian image modeling, Proc. Sixth Eur. Conf. on Computer Vision, vol. 1, Springer, Berlin, 2000, pp. 3–17.
- [20] J.M. Winn, Variational message passing and its application, Ph.d. thesis, Department of Physics, University of Cambridge, 2003.
- [21] Q. Wang, G. Xu, H. Ai, Learning object intrinsic structure for robust visual tracking, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, 2003, pp. 227–233.
- [22] J. MacCormik, M. Isard, Partitioned sampling, articulated objects, and interface-quality hand tracking, in: Proc. Eur. Conf. on Computer Vision, 2000.
- [23] K. Murphy, Y. Weiss, M. Jordan, Loopy-belief propagation for approximate inference: an empirical study, in: Proc. Fifteenth Conf. on Uncertainty in Artificial Intelligence, 1999.
- [24] J. Yedidia, W. Freeman, Y. Weiss, Understanding belief propagation and its generalization, in: *Exploring Artificial Intelligence in the New Millennium*, Elsevier Science and Technology Books, Amsterdam, 2003, pp. 239–286 (Chapter 8).
- [25] W.T. Freeman, E.C. Pasztor, Learning low-level vision, in: Proc. IEEE Internat. Conf. on Computer Vision, 1999, pp. 1182–1189.
- [26] W.T. Freeman, H. Zhang, Shape-time photography, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2003.
- [27] T.S. Jaakkola, Tutorial on variational approximation method, *Advanced Mean Field Methods: Theory and Practice*, MIT Press, Cambridge, MA, 2000.
- [28] J. Vermaak, N.D. Lawrence, P. Pérez, Variational inference for visual tracking, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 1, 2003, pp. 773–780.
- [29] M.J. Beal, Variational algorithms for approximate bayesian inference, Ph.d. thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.
- [30] J.S. Liu, R. Chen, Sequential Monte Carlo methods for dynamic systems, *J. Am. Stat. Assoc.* 93 (443) (1998) 1032–1044.
- [31] E.B. Sudderth, A. Ihler, W. Freeman, A. Willsky, Nonparametric belief propagation, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2003, pp. 605–612.
- [32] M. Isard, PAMPAS: real-valued graphical models for computer vision, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2003, pp. 613–620.
- [33] S. Ioffe, D.A. Forsyth, Mixtures of trees for object recognition, in: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, 2001, pp. 180–185.
- [34] J.M. Coughlan, S.J. Ferreira, Finding deformable shapes using loopy belief propagation, in: Proc. Eur. Conf. on Computer Vision, Lecture Notes in Computer Science, vol. 2352, 2002, pp. 453–468.

- [35] L. Sigal, M. Isard, B. Sigelman, M. Black, Attractive people: assembling loose-limbed models using non-parametric belief propagation *Advances in Neural Information Processing System*, vol. 16, MIT Press, Cambridge, MA, 2004.
- [36] J. Gao, J. Shi, Multiple frame motion inference using belief propagation, in: *Proc. IEEE Inter. Conf. on Automatic Face and Gesture Recognition*, 2004.