

ANALYZING STRUCTURED DEFORMABLE SHAPES VIA MEAN FIELD MONTE CARLO

Gang Hua, Ying Wu, Ting Yu

ECE Department, Northwestern University
2145 Sheridan Road, Evanston, IL 60208, USA

{ganghua,yingwu,tingyu}@ece.nwu.edu

ABSTRACT

This paper describes a novel approach to analyzing and tracking the motion of structured deformable shapes that consist of multiple correlated deformable subparts. Due to the high dimensional nature of this problem, existing methods are plagued either by the inability of capturing detailed local deformation or the enormous complexity, induced by the curse of dimensionality. Taking advantage of the structure of the deformable shapes, the paper presents a new representation, i.e., dynamic Markov network model, to overcome the challenges induced by high dimensionality. Probabilistic variational analysis of this Markov network model reveals a set of fixed point equations, i.e., the mean field equations, which manifest the interactions among the posterior motions of these deformable subparts and suggest an efficient solution to such a high dimensional motion analysis problem. Combined with Monte Carlo strategies, the new algorithm, namely mean field Monte Carlo (MFMC), achieves very efficient Bayesian inference with close-to-linear complexity. Experiments on tracking human faces and human lips demonstrate the effectiveness of the proposed method.

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, 5], or as a set of control points [2, 6]. The second problem is how to recover the structured deformation from the video sequence. There are mainly two approaches: the top-down

approach takes a two step strategy, i.e., the hypothesis generation and image observation verification such as the particle filtering algorithm [5]; while the bottom-up approach estimates the best model parameters by optimizing a cost function just as the SNAKES [4].

The high dimensional nature of structured deformable shapes causes many challenges. Both approaches mentioned above are challenged by this situation: for the first approach, e.g., the particle filtering algorithm [5], 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., SNAKES [4], the cost function 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, detailed local deformation is hardly to be analyzed.

Our method employs a dynamic Markov network to represent the structured deformable shapes. The structural constraints are modeled in the Markov network at each time instant. The probabilistic mean field variational analysis of the 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 mean field Monte Carlo (MFMC) [7] algorithm is proposed as a nonparametric approximation. Different subparts can be tracked by different trackers while these trackers can exchange information with each other to reinforce the structural constraints. This way, we achieve very efficient Bayesian inference 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 of this paper will be discussed; then in Section 3, we propose the Markov network representation and a mean field Monte Carlo algorithm; in Section 4, by extending the model and the algorithm, we propose a dynamic Markov network to model the deformation of the structured deformable shapes and a sequential mean field Monte Carlo algorithm to implement the inference; experimental results will be presented and discussed in Section 5; and the conclusion and future work will be presented in Section 6.

2. RELATED WORK

Because of the high dimensional nature of structured deformable shapes, the analysis of them faces the curse-of-dimensionality. We use a dynamic Markov network representation to approach it. There are two problems we need to address, the first is how to formulate such a tracking algorithm, the second is how to implement the Bayesian inference for the dynamic Markov network. Therefore, related work in the literature can be categorized into two, the first category addresses the problem of deformable object tracking; the second category deals with the inference of complex statistical graphical models.

For deformable object tracking, there are two main approaches just as briefly mentioned in Section 1. The representatives of them are the particle filtering tracker [5] and its variants [8, 9] and the SNAKES [4]. To attack the high dimensionality problem, the active shape model [2] uses PCA to learn and find the best linear approximation of the original model. More recent work includes using ISOMAP algorithm for dimension reduction [10]. However, learning a lower dimensional description of the object needs a set of training data and sometimes it would be very difficult to obtain. Partitioned sampling [11] samples different parts one by one but the information flow is unidirectional.

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 can obtain good approximate results in many applications [12, 13]. Probabilistic variational analysis is a principled approximate inference technique. It uses a more tractable approximate form of the posterior probability [14, 9] and an approximate inference is achieved by minimizing the K-L divergence between the approximated posterior and the original one. Particle filtering [5] employs a set of weighted samples to represent the real distributions and probabilistic inference is implemented based on these samples. Most recent work using particle filtering to perform the inference of generalized graphical model includes the nonparametric belief propagation (NBP) algorithm [15] and the PAMPAS algorithm [16] where Monte Carlo method is used in the message passing process in belief propagation. In both algorithms, the messages are modeled as Gaussian mixtures and Markov chain Monte Carlo (MCMC) samplers are designed to draw samples from them.

In the proposed method, each subpart of the structured deformable shape is modeled by a parametric description of a spline curve in the motion space. And the constraint between two subparts are modeled 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 [14] with the idea of particle filtering [5]. Different from NBP [15] and PAMPAS [16], it is a fully nonparametric inference algorithm with close to linear complexity.

3. REPRESENTATION AND MFMC

Suppose the structured deformable object consists of \mathcal{K} subparts, then we denote 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 reveals the probability of two random variables being subject to a certain constraint. Also, 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.

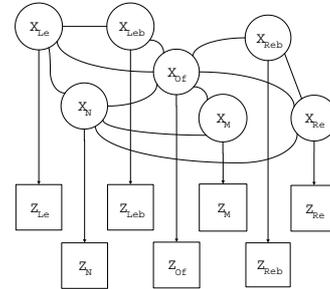


Fig. 1. Markov network for human face.

Figure 1 shows a Markov network of human face where the subscripts 'Of', 'Leb', 'Reb', 'Le', 'Re', 'N' and 'M' correspond to out 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 $\phi(\mathbf{z}_i|\mathbf{x}_i)$.

The joint probability corresponding to the graphical model in Figure 1 is:

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

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

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

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

where $Q_i(\mathbf{x}_i)$ is an independent approximate distribution of $P(\mathbf{x}_i|\mathbf{Z})$. Then we can construct a cost function

$$\begin{aligned} J(Q) &= \log P(\mathbf{Z}) - \int_{\mathbf{X}} \prod_i Q_i(\mathbf{x}_i) \log \frac{\prod_i Q_i(\mathbf{x}_i)}{\mathbf{P}(\mathbf{X}|\mathbf{Z})} \\ &= - \sum_i H_i(Q_i) + \int_{\mathbf{x}_i} Q_i(\mathbf{x}_i) \mathbf{E}_Q \{ \log \mathbf{P}(\mathbf{X}, \mathbf{Z}) | \mathbf{x}_i \} \end{aligned}$$

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

$$E_Q \{ \log P(\mathbf{X}, \mathbf{Z}) | \mathbf{x}_i \} = \int_{\{\mathbf{x}_j\} \setminus \mathbf{x}_i} \left[\prod_{\{j\} \setminus i} Q_j(\mathbf{x}_j) \right] \log \mathbf{P}(\mathbf{X}, \mathbf{Z}). \quad (3)$$

Minimizing $J(Q)$ will lead to $Q_i(\mathbf{x}_i) \rightarrow \mathbf{P}(\mathbf{x}_i|\mathbf{Z})$. Taking the derivative of $J(Q)$ w.r.t. $Q_i(\mathbf{x}_i)$, and constraining $\sum_{\mathbf{x}_i} Q_i(\mathbf{x}_i) = \mathbf{1}$, we can obtain a set of fixed point equations for $Q_i(\mathbf{x}_i)$ [14, 7]:

$$Q_i(\mathbf{x}_i) \leftarrow \frac{1}{Z_{\mathcal{E}}} e^{\mathbf{E}_Q \{ \log \mathbf{P}(\mathbf{X}, \mathbf{Z}) | \mathbf{x}_i \}} \quad (4)$$

where $Z_{\mathcal{E}}$ is a normalization constant to make sure $Q_i(\mathbf{x}_i)$ is a valid probabilistic distribution. Embedding equation 1 into equation 3 and equation 4, we can obtain a set of simplified mean field fixed point equations [14, 7]:

$$Q_i(\mathbf{x}_i) \leftarrow \frac{1}{Z'_{\mathcal{E}}} \phi(\mathbf{z}_i | \mathbf{x}_i) e^{\sum_{j \in \mathcal{N}(i)} \int_{\mathbf{x}_j} Q_j(\mathbf{x}_j) \log \psi(\mathbf{x}_i, \mathbf{x}_j)} \quad (5)$$

where $Z'_{\mathcal{E}}$ is again a normalization constant. 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., $Q_i(\mathbf{x}_i) \sim \{s_{\mathbf{x}_i}^{(n)}, \pi_{\mathbf{x}_i}^{(n)}\}$. The iteration of the fixed point equation set is then implemented based on these samples. This leads to the mean filed Monte Carlo (MFMC) algorithm, which is shown in Figure 2.

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

1. Importance Sampling: Sample $\{s_{x_i, k+1}^{(n)}, \frac{1}{n}\}$ from a suitable importance function $I_i(x_i)$.
2. Re-weight: For each sample $s_{x_i, k+1}^m$, set its weight to $\omega_{x_i, k+1}^m = \phi(z_i | s_{x_i, k+1}^m) G_{x_i}(s_{x_i, k+1}^m) / I_i(s_{x_i, k+1}^m)$ where $G_{x_i}(s_{x_i, k+1}^m) = e^{\sum_{j \in \mathcal{N}(i)} \sum_{n=1}^N \omega_{\mathbf{x}_j, k}^{(n)} \log \psi(s_{\mathbf{x}_i, k+1}^m, s_{\mathbf{x}_j, k}^{(n)})}$
3. Normalization: $\omega_{x_i, k+1}^m = \omega_{x_i, k+1}^m / \sum_m \omega_{x_i, k+1}^m$, then we get $\{s_{x_i, k+1}^{(n)}, \omega_{x_i, k+1}^{(n)}\}$
4. Iteration: $k \leftarrow k + 1$, iterate until convergence.

Fig. 2. Mean Filed Monte Carlo Algorithm

Compared with the NBP algorithm [15] and the PAMPAS algorithm [16], the MFMC algorithm employs weighted samples for all the distributions while both NBP and PAMPAS model these distributions as Gaussian mixtures. In this sense, our algorithm is fully nonparametric. Also, our algorithm employs importance sampling thus it avoids using MCMC samplers which have been used in both NBP and PAMPAS. The use of importance sampling has two advantages: firstly, it avoids sampling directly from the complex Gibbs distribution of equation 4 or equation 5; secondly, domain knowledge can be used to construct a suitable importance function to generate samples from more confident areas.

4. 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 Figure 3. It is the temporal extension of the Markov network in Figure 1. Each horizontal arrow represents the dynamic motion model of each corresponding subpart. Denote $\underline{\mathbf{Z}}_{1:t} =$

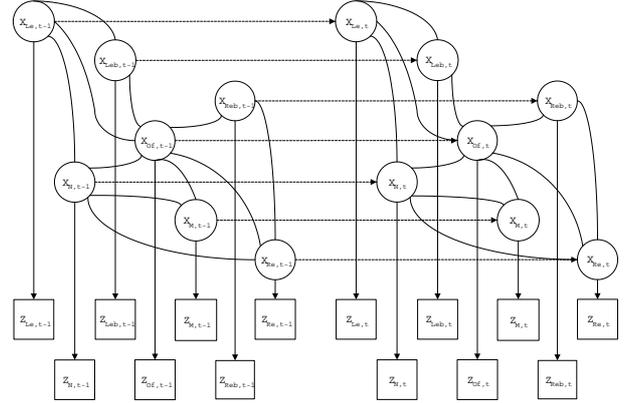


Fig. 3. DMN for Human Face deformation.

$\{\mathbf{Z}_1, \dots, \mathbf{Z}_t\}$, the inference problem becomes [5]:

$$P(\mathbf{X}_t | \underline{\mathbf{Z}}_{1:t}) \propto P(\mathbf{Z}_t | \mathbf{X}_t) \int_{\mathbf{X}_{t-1}} P(\mathbf{X}_t | \mathbf{X}_{t-1}) P(\mathbf{X}_{t-1} | \underline{\mathbf{Z}}_{1:t-1}) \quad (6)$$

Also, assuming independent dynamic model for each subpart, i.e.,

$$P(\mathbf{X}_t | \mathbf{X}_{t-1}) = \prod_i P(\mathbf{x}_{i,t} | \mathbf{x}_{i,t-1}) \quad (7)$$

the mean field approximation will result in the following fixed point equations [7]:

$$\begin{aligned}
Q_{i,t}(\mathbf{x}_{i,t}) &= \frac{1}{Z_C} \phi(\mathbf{z}_{i,t} | \mathbf{x}_{i,t}) \\
&\times \int_{x_{i,t-1}} P(x_{i,t} | x_{i,t-1}) Q_{i,t-1}(\mathbf{x}_{i,t-1}) \\
&\times \exp\left(\sum_{j \in \mathcal{N}(i)} Q_{j,t}(x_{j,t}) \log(\psi(x_{i,t}, x_{j,t}))\right)
\end{aligned} \tag{8}$$

where Z_C is a normalization constant. Comparing equation 8 with equation 5, we can see that there is one more item in equation 8, which exactly models the dynamic prediction prior. Actually, if we do know the priori probability of each node in Figure 1, we can also incorporate a local prior item in equation 4 and equation 5.

This set of fixed point equations are more complex because they involve multiple integrations of complex probabilistic distributions. Again, we adopt a Monte Carlo strategy to approach to this problem. We call it the sequential mean field Monte Carlo algorithm (SMFMC), as shown in Figure 4.

The SMFMC algorithm has two steps, the first step is the sequential Monte Carlo (SMC) process; then, the result is used as the initialization of the MFMC process. The MFMC process runs until convergence.

5. EXPERIMENTS

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

5.1. Tracking Human Lip

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 6-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. In our experiment, we use a Gaussian potential function to model the constraint, i.e., the average distance of the two pair of end points obey a zero mean Gaussian distribution. And the parameter of the potential function can be trained from a set of manually annotated images. The observation function is similar as that in [5].

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

Generate $\{s_{x_{i,t}}, \omega_{x_{i,t}}\}_{n=1}^N$ from $\{s_{x_{i,t-1}}, \omega_{x_{i,t-1}}\}_{n=1}^N$.

I. Initialization: Sequential Monte Carlo, $k \leftarrow 1$

I.1. Re-sampling: Re-sampling $\{s_{x_{i,t-1}}\}_{n=1}^N$ according to the weights $\omega_{x_{i,t-1}}^{(n)}$ to get $\{s_{x_{i,t-1}}, \frac{1}{N}\}_{n=1}^N$

I.2. Prediction: For each weighted sample in $\{s_{x_{i,t}}\}_{n=1}^N$, sampling from $P(\mathbf{x}_{i,t} | \mathbf{x}_{i,t-1})$ to get $\{s_{x_{i,t,k}}, \frac{1}{N}\}_{n=1}^N$

I.3. Re-weight: Assign weight $\omega_{x_{i,t,k}}^{(n)} = \phi(z_i | s_{x_{i,t,k}}^{(n)})$ to each corresponding $s_{x_{i,t,k}}^{(n)}$ and normalize the weight such that $\sum_{n=1}^N \omega_{x_{i,t,k}}^{(n)} = 1$

II. Iteration: Mean Field Monte Carlo

II.1. Importance Sampling: Sample $\{s_{x_{i,t,k+1}}, \frac{1}{N}\}_{n=1}^N$ from a suitable importance function $I_i(x_{i,t})$.

II.2. Re-weight: For each sample $s_{x_{i,t,k+1}}^m$, set its weight to
$$\omega_{x_{i,t,k+1}}^m = \frac{\phi(z_i | s_{x_{i,t,k+1}}^m) D_{x_{i,t}}(s_{x_{i,t,k+1}}^m) G_{x_{i,t}}(s_{x_{i,t,k+1}}^m)}{I_i(s_{x_{i,t,k+1}}^m)}$$
 where
$$D_{x_{i,t}}(s_{x_{i,t,k+1}}^m) = \sum_{n=1}^N P(s_{x_{i,t,k+1}}^m | s_{x_{i,t-1}}^n) \omega_{x_{i,t-1}}^{(n)}$$
 and
$$G_{x_{i,t}}(s_{x_{i,t,k+1}}^m) = e^{\sum_{j \in \mathcal{N}(i)} \sum_{n=1}^N \omega_{x_{j,t,k}}^{(n)} \log \psi(s_{x_{i,t,k+1}}^m, s_{x_{j,t,k}}^{(n)})}$$

II.3. Normalization: $\omega_{x_{i,t,k+1}}^m = \omega_{x_{i,t,k+1}}^m / \sum_m \omega_{x_{i,t,k+1}}^m$, then we get $\{s_{x_{i,t,k+1}}, \omega_{x_{i,t,k+1}}^{(n)}\}$

II.4. Iteration: $k \leftarrow k + 1$, iterate until convergence.

III. Result: $\{s_{x_{i,t}}, \omega_{x_{i,t}}^{(n)}\}_{n=1}^N \leftarrow \{s_{x_{i,t,k}}, \omega_{x_{i,t,k}}^{(n)}\}_{n=1}^N$

Fig. 4. Sequential Mean Filed Monte Carlo Algorithm

in Figure 5. For comparison, a single SMC tracker and a multiple independent SMC (MiSMC) tracker are also implemented. The results are shown in Figure 6 and Figure 7, respectively. From the results, we observe that the single SMC tracker can not 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 can not 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.

In fact, if a very tight potential function is used in the



Fig. 5. Tracking Human Lip Via SMFMC, more results in “lip.avi”.



Fig. 6. Tracking Human Lip Via a Single SMC Tracker.

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.

5.2. Face Alignment and Tracking

The SMFMC algorithm proposed in this paper can be used for automatic face alignment. 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.

The face model is exactly what we have proposed in Figure 1. The potential function is constructed based on the assumption that the state of two connected nodes be as close as possible in the parametric space. Again, we employ a Gaussian density to model the distribution of the distance of two constrained subparts in the parametric space. And it can be trained from a set of manually annotated images. The other setup is similar as those in Section 5.1.

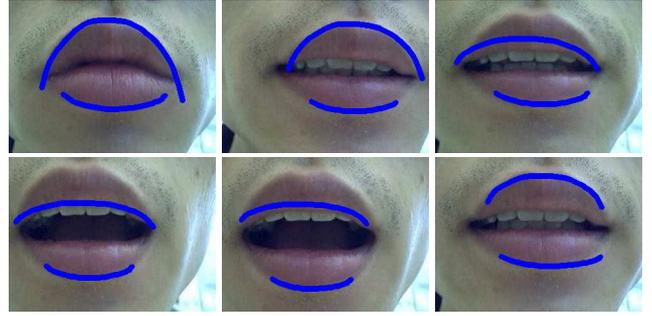


Fig. 7. Tracking Human Lip Via a MiSMC Tracker.

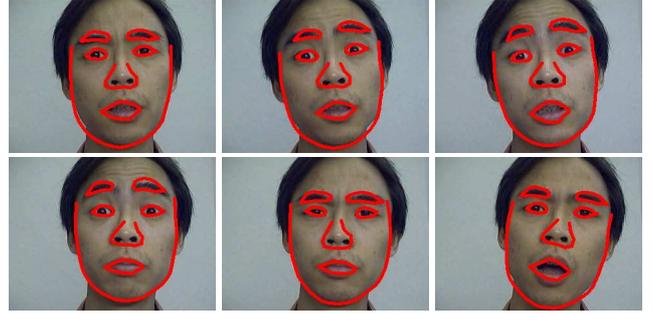


Fig. 8. Tracking Human Face Via Sequential MFMC, more results in “face.avi”.

The proposed SMFMC algorithm achieves good results for automatic face alignment and face tracking over a 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 Figure 8. Compared with the ASM [2], our face prior model is actually a Gibbs model while the ASM uses a Gaussian model. Thus our method is more general than the ASM model, and more capable of capturing detailed shape deformation.

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 w.r.t. the dimensionality. Suppose

SMCLip	MiTlip	SMFMCCLip	SMFMCFace
16.5 F/s	10.9 F/s	4.0 F/s	1.6 F/s

Table 1. Processing Frame Rate of different trackers

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 $O(M\mathcal{K}\mathcal{N})$ [7], 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 complexed 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. Noticing that the lip has 2 subparts, the face has 7 subparts and the number of mean field iteration is 3, it verifies that the SMFMC tracker do have linear complexity w.r.t. the dimensionality just as we have expected.

6. CONCLUSION AND DISCUSSION

In this paper, a novel SMFMC algorithm is proposed to analyze and track structured deformable shapes based on a dynamic Markov network representation. The SMFMC algorithm combines both analytical and nonparametric inference methods, i.e., the statistical 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 [7]. The idea of dealing with the local subparts while reinforcing the global constraint proved to be an effective way of overcoming the curse-of-dimensionality. Our future work includes applying SMFMC to other vision problems and further theoretical study of SMFMC such as the convergence rate of the algorithm.

7. ACKNOWLEDGMENT

This work was supported in part by National Science Foundation Grant IIS-0308222, Northwestern faculty startup funds for YW and Murphy Fellowships for GH and TY.

8. REFERENCES

- [1] Robert Kaucic, Barney Dalton, and Andrew Blake, "Real-time lip tracking for audio-visual speech recognition applications," in *Proc. European Conference on Computer Vision*, Cambridge, UK, 1996, vol. 2, pp. 376–387.
- [2] T.F. Cootes, D. Cooper, C.J. Taylor, and J. Graham, "Active shape models - their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, January 1995.
- [3] Dorin Comaniciu, "Nonparametric information fusion for motion estimation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Madison, Wisconsin, June 2003.
- [4] Michael Kass, Andrew Witkin, and Demetri Terzopoulos, "SNAKES: Active contour models," *International Journal of Computer Vision*, pp. 321–331, 1988.
- [5] Andrew Blake and Michael Isard, "Contour tracking by stochastic propagation of conditional density," in *Proc. European Conference on Computer Vision*, 1996, vol. 1, pp. 343–356.
- [6] Ce Liu, Heung-Yeung Shum, and Changshui Zhang, "Hierarchical shape modeling for automatic face localization," in *Proc. European Conference on Computer Vision*, May 2002, pp. 687–703.
- [7] Ying Wu, Gang Hua, and Ting Yu, "Tracking articulated body by dynamic markov network," in *Proc. International Conference on Computer Vision*, Nice, Côte d'Azur, France, October 2003, pp. 1094–1101.
- [8] Michael Isard and Andrew Blake, "Icondensation: Unifying low-level and high-level tracking in a stochastic framework," in *Proc. European Conference on Computer Vision*, 1998.
- [9] Ying Wu and Thomas S. Huang, "Robust visual tracking by co-inference learning," in *Proc. IEEE Int'l Conference on Computer Vision*, Vancouver, July 2001, vol. II, pp. 26–33.
- [10] Qiang Wang, Guangyou Xu, and Haizhou Ai, "Learning object intrinsic structure for robust visual tracking," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, June 2003, vol. 2, pp. 227–233.
- [11] John MacCormik and Michael Isard, "Partitioned sampling, articulated objects, and interface-quality hand tracking," in *Proc. European Conference on Computer Vision*, 2000.
- [12] Kevin Murphy, Yair Weiss, and Michael Jordan, "Loopy-belief propagation for approximate inference: An empirical study," in *Proc. Fifteenth Conference on Uncertainty in Artificial Intelligence*, Stockholm, Sweden, July 1999.
- [13] William T. Freeman and Egon C. Pasztor, "Learning low-level vision," in *Proc. International Conference on Computer Vision*, Kerkyra, Greece, September 1999.
- [14] Tommi S. Jaakkola, "Tutorial on variational approximation method," *Advanced mean field methods: theory and practice*. MIT Press, 2000.
- [15] Erik B. Sudderth, Alexander T. Ihler, William T. Freeman, and Alan S. Willsky, "Nonparametric belief propagation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Madison, Wisconsin, June 2003, pp. 605–612.
- [16] Michael Isard, "PAMPAS: Real-valued graphical models for computer vision," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Madison, Wisconsin, June 2003, pp. 613–620.