Groupwise Shape Registration on Raw Edge Sequence via A Spatio-Temporal Generative Model

Huijun Di, Rao Naveed Iqbal, Guangyou Xu, Linmi Tao
State Key Laboratory of Pervasive Computing
Department of Computer Science and Technology, Tsinghua University, Beijing 100084, PR China.
{dhj98,naveed03}@mails.tsinghua.edu.cn,{xgy-dcs,linmi}@tsinghua.edu.cn

Abstract

Groupwise shape registration of raw edge sequence is addressed. Automatically extracted edge maps are treated as noised input shape of the deformable object and their registration are considered, results can be used to build statistical shape models without laborious manual labeling process. Dealing with raw edges poses several challenges, to fight against them a novel spatio-temporal generative model is proposed which joints shape registration and trajectory tracking. Mean shape, consistent correspondences among edge sequence and associated non-rigid transformations are jointly inferred under EM framework. Our algorithm is tested on real video sequences of a dancing ballerina, talking face, and walking person. Results achieved are interesting, promising, and prove the robustness of our method. Potential applications can be found in statistical shape analysis, action recognition, object tracking, etc.

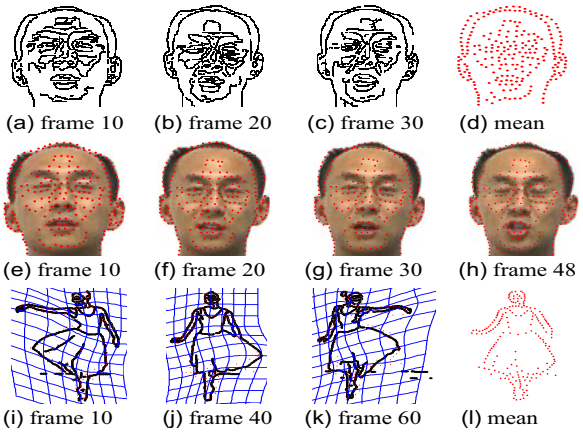

Figure 1. Input examples and achieved results. (a)-(c): raw edge maps of a talking face, (d): learned mean shape, (e)-(h): face images superimposed with obtained correspondence points; (i)-(k): raw edge maps of a ballet sequence along with estimated deformation field, (l): learned mean shape.

1. Introduction

Statistical models of shape have proved powerful tools for various tasks [1][2], such as segmentation, recognition, and tracking, where they provide *a priori* shape information of the deformable object. To construct such models, correspondences among all shapes over a set of training images are required. In order to meet this requirement two typical ways are investigated in the literature: use a set of manually defined landmarks or perform shape registration. Since manual or semiautomatic landmark labeling is time consuming, automatic ways were explored, e.g. [3] placed landmarks via locating and tracking salient features from image sequences. Nevertheless for objects like human body or human hand, it is difficult to find sufficient landmarks automatically. The second way deals with the unknown correspondence of training shapes and considers their registration in pairwise/groupwise [7]-[17] (details are covered in section 2). However pre-segmented shapes are still required for practical model building and usually it is again a time-consuming manual process.

Our interest is to design an automatic way to construct statistical shape models. We consider the case that builds the model from a training video sequence. To avoid manual labeling process, an edge sequence is first extracted by applying edge detection at each frame, and then treated directly as the noised input shape of the deformable object. The problem addressed in this paper therefore is the groupwise shape registration of this raw edge sequence. Figure 1 demonstrates input examples and achieved results. In our work mean shape (represented by a point set), correspondences and associated non-rigid transformations are jointly estimated from the raw edge maps. Subsequently, these results can be used to build the statistical shape model of the deformable object, and they are also useful in action recognition, object tracking, etc.

Based on our results, a shape model (at edge level) in-between sparse (feature) level and dense (pixel) level can

1-4244-1180-7/07/$25.00 ©2007 IEEE

be built. Either the estimated transformation parameters can be applied to model the deformation fields, or the obtained correspondence points can be used to learn a Point Distribution Model (PDM). Our outcome can also give a sufficient initialization for image based registration. Moreover when there is a significant non-rigid deformation (like in ballet sequence), we can still automatically learn a meaningful mean shape shared by the cluttered edges, thus enriching related applications.

For action recognition, the estimated transformation parameters or correspondence points can be good features to characterize the action, e.g. both can be used to analyze the function space of activities [6] and correspondence points can help in building the action sketch in [5].

For object tracking, we can automatically learn a shape representation of the non-rigid object from raw edge maps and this representation further can be used to learn the shape space of the object [2]. Moreover, our method can even serve as an offline shape tracker without need of *a priori* shape model. For instance, in CAVIAR database [4], by applying edge detection inside the labeled boxes of moving people in the recorded video clips, our method can be used to track the shape of human body and the database can be re-annotated with detailed shape information automatically.

However, the problem addressed is quite challenging due to the following tangled difficulties: unknown correspondence, non-rigid deformation, massive loss of data due to occlusions and edge drop-outs, ambiguity and outliers in cluttered edges (e.g. stray edges), etc. Under these severe conditions, it is a tough ask to register raw edge maps and to learn a meaningful mean shape from them. Fighting against these challenges, we introduce the view point of trajectory tracking into shape registration and joint them via a novel spatio-temporal generative model: mixture of transformed continuous-state HMMs (MoTcHMMs). Spatial and temporal constraints are considered in MoTcHMMs to regularize the registration process of raw edge sequence.

To encode the temporal information, the whole input edge sequence is viewed as a spatio-temporal volume and characterized by a mixture of trajectories, thus leading to trajectory tracking sub-problem. Each continuous-state HMM (c-HMM) in MoTcHMMs corresponds to a trajectory. To ensure consistent spatial relationships of the trajectories among frames, a mean shape (i.e. a common spatial structure) is associated with them and mapped to the states of c-HMMs at each frame through a smooth transformation function; this leads to groupwise shape registration sub-problem. Joint inference for these two sub-problems is carried out under EM framework and analytic Viterbi algorithm is derived for trajectory optimization. Resultantly, mean shape, consistent correspondences among complete edge sequence and associated non-rigid transformations are jointly estimated.

The paper is organized as follows; related work is discussed in section 2. Problem formulation is given in section 3; section 4 explains inference of the model along with complete algorithm. Section 5 discusses outlier and missing data handling. Test results on different data sets are presented in section 6. Finally, conclusion and future work is elaborated.

2. Related Work

Shape registration works under unknown correspondence are briefly reviewed here with special emphasis on the aspects of raw edge data handling.

Parameterization is employed in shape modeling works to deal with unknown correspondence e.g. in [7], correspondence problem is posed as finding the set of parameterizations for each shape that builds the best model, and an objective function is defined based on minimum description length criterion in a rigorous theoretical way. However, it is hard for these kinds of methods to deal with raw edge data since they assume single contour as input to enable the parameterization.

A graph is used in [10]-[12] to regularize the resulting correspondence between a template and target shape by preserving neighborhood structure among shape points. Loopy belief propagation is applied for efficient inference in [10][11], and relaxation labeling is used in [12]. Moreover, [11] assumes the order in shape points to penalize incorrect correspondences. However, when applying these graph based approaches in our problem domain, it is difficult to define a noiseless template automatically and it is not easy for them to learn a mean shape from cluttered edges. Besides, shape context [8] and integral invariants [9] assign a useful discriminative attribute to shape point and can be embedded in other registration methods [11][12].

A non-rigid registration algorithm is presented in [13][14] for a pair (set) of unlabeled point sets. The main strength of their work is the ability to solve chicken-egg problem by jointly estimating the correspondences as well as non-rigid transformation using EM algorithm and deterministic annealing. The work of [14] extends pairwise shape registration [13] to groupwise way and mean shape learning is posed as a density estimation problem. However, the registration result in [13][14] is not stable under point sets with outliers, as mentioned in [16][17].

Recently, works in [15]-[17] provide an interesting way to register shapes without explicitly establishing the correspondence. They define a probability distribution on the transformed version of each input point set and optimize an information-theoretic measure between them, yielding the desired transformations. The probability distribution is calculated based on kernel density estimation in [15][17], and is characterized explicitly in [16] via mixtures of Gaussians (MoG). For optimization, a pairwise similarity measure is

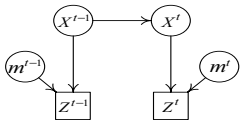

Figure 2. MocHMMs: Trajectory tracking

defined as kernel correlation in [15] and L_2 distance between two MoGs in [16]. It is a nontrivial task to extend work in [15][16] to groupwise shape registration under raw edge data since selection of proper reference shape is not easy. A groupwise way is presented in [17] where input data sets are registered and pooled at the end of the optimization of the CDF-based Jensen-Shannon divergence. However groupwise shape registration in the presence of outliers isn't discussed in the paper.

Our work is inspired by [13] and [14], although they are sensitive to outliers, we have achieved robustness against missing data and outliers by joint modeling the spatial and temporal information presented in the edge sequence.

3. Problem Formulation

The input edge sequence is generated via edge detection at each frame inside a region of interest (ROI) from a given video sequence (more specific details will be covered in section 6). Let input edge sequence consists of N frames, and the edge points in frame t are denoted by $\{Z^t : Z_i^t, t = 1, 2, ..., N, i = 1, 2, ..., N_t\}$, where N_t denotes the number of the edge points, and Z_i^t is a d dimensional coordinate vector of the ith edge point with size of dx1 (d always equal to 2 in our case). Hereinafter, when omitting superscript and/or subscript, expression implies use of whole set instead of a specific term.

In the following subsections, trajectory tracking is first considered to impose temporal dynamic constraints, later it is combined with shape registration by a spatio-temporal generative model, and our registration problem of raw edge sequence is expressed as the inference of this model.

3.1. Trajectory Tracking: MocHMMs

In order to fully explain our idea we use the concept of a spatio-temporal volume (STV) [5] which is formed by stacking all input edge frames. Intuitively, this STV can be characterized by a number of trajectories, say L trajectories without loss of generalization (it can be viewed as trajectory-set representation of a 3D volume, bearing an analogy to point-set representation of 2D shape). Formally, in the view point of generative model, this STV can be explained (or generated) by a mixture of continuous-state HMMs (MocHMMs), where each c-HMM with single Gaussian observation corresponds to a trajectory (model is shown in figure 2). When each c-HMM is inferred, subsequently its corresponding trajectory is also tracked.

Based on the MocHMMs shown in figure 2, a sample Z_i^t can be generated by sampling from it. At frame t, discrete random variable (RV) m_i^t denote the index of c-HMM which generates the current sample Z_i^t, and $m_i^t \in \{1, 2, ..., L\}$; continuous RV X_j^t is the state of jth c-HMM at current frame, $j = 1, 2, ..., L$.

To achieve tractable inference of c-HMM, idea of Kalman filter [18] is borrowed here to model the dynamics and observation, i.e. linear dynamic model with additive Gaussian noise is considered in-between X_j^t and X_j^{t-1}; linear observation model with additive Gaussian noise is considered in-between X_j^t and Z_i^t when given $(m_i^t = j)$:

$$p(X_j^t|X_j^{t-1}) = N(X_j^t, X_j^{t-1}A, \Psi_j^t), \tag{1}$$

$$p(Z_i^t|m_i^t = j, X^t) = N(Z_i^t, X_j^t B, \Phi_j^t), \tag{2}$$

where $N(X, \mu, \Phi)$ denote the normal density function on X with mean μ and variance Φ. The size of X_j^t, A, Ψ_j^t, B and Φ_j^t are $d * D, D * D, D * D, D * 1$ and $d * d$, respectively. D is the order of the linear dynamic model. The state X_j^t of c-HMM is augmented with derivative of shape w.r.t. time. For example, in constant velocity model used here, X_j^t is formed by stacking a position vector p and a velocity vector v into one vector ($D = 2$, see [18] for more examples), i.e.

$$X_j^t = \lfloor p_j^t \quad v_j^t \rfloor, \tag{3}$$

its associated dynamic matrix A and measurement matrix B are $A = [1\ 0; 1\ 1]$, $B = [1 \quad 0]^T$.

So far, temporal information is modeled in MocHMMs, in next section, we will explain why and how to model the spatial information as well, and give our ultimate problem formulation.

3.2. Joint Shape Registration and Trajectory Tracking: MoTcHMMs

MocHMMs proposed in section 3.1 doesn't model the relationship among c-HMMs, therefore it can't preserve local neighborhood structures of trajectories $\{X_j^t, j = 1, 2, ..., L\}$, i.e. two c-HMMs (trajectories) might tangle each other. Consequently, spatial constraint needs to be enforced to prevent this effect. Inspired by [14], a mean shape is introduced into MocHMMs as a common spatial structure, and spatial constraints associated to c-HMMs are imposed by registering the mean shape to them in each frame.

Mean shape Y is represented by a point set with the same number of c-HMMs, i.e.$\{Y : Y_j, j = 1, 2, ..., L\}$ where Y_j is the $2D$ coordinate vector of the jth point in mean shape, and L is the number of c-HMMs. In order to enforce the spatial constraint, a transform function $f^t : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is introduced at frame t to associate Y with the current states of c-HMMs by defining a new state vector:

$$X_j^t = [f^t(Y_j) \quad \tilde{X}_j^t]. \tag{4}$$

Comparing eqn. 4 with eqn. 3, p_j^t is replaced by $f^t(Y_j)$, and $\tilde{X}_j^t$ denotes the rest parts in eqn. 3 ($\tilde{X}_j^t \equiv v_j^t$ here). It is worth noting that X_j^t isn't an actual variable here, it consists of $\tilde{X}_j^t$, f^t and Y_j based on eqn. 4, and just used for easy explanation. Now Y is a common factor connected with all c-HMMs under f^t and smoothness constraint of f^t is used to help preserve local neighborhood structures of the trajectories $\{X_j^t, j = 1, 2, ..., L\}$, i.e. with a smooth transformation f^t , two trajectories are hard to be tangled.

This new model is a transformed version of MocHMMs, i.e. mixture of transformed continuous-state HMMs (MoTcHMMs), and it is a generative model explaining the spatio-temporal information presented in edge sequence. It is hard to express current model via Bayesian network, as the conditional probabilities among f^t, Y and $\tilde{X}_j^t$ are not easy to describe. A factor graph representation for MoTcHMMs is presented and shown in figure 3. Factor graph is "a bipartite graph that expresses which variables are arguments of which local functions" [20], where a global objective function factors into a product of these local functions. Our objective is the joint distribution of $\tilde{X}$, Y, f, m and Z, given by the product of all local functions in figure 3:

$$p(\tilde{X}, Y, f, m, Z) = \frac{1}{Z_{partition}} \prod_{j=1}^{L} g_y(Y_j) \prod_{t=1}^{N} g_f(f^t) \prod_{t=1}^{N} \Big(\prod_{j=1}^{L} g_x(X_j^t, X_j^{t-1}) \prod_{i=1}^{N} g_m(m_i^t) g_z(Z_i^t, m_i^t, X^t) \Big) \tag{5}$$

where constant $Z_{partition}$ (w.r.t. $\tilde{X}, Y, f$, and m) ensures the distribution is normalized. Local functions are

$$g_y(Y_j) = p(Y_j), g_m(m_i^t) = p(m_i^t), \tag{6}$$

$$g_x(X_j^t, X_j^{t-1}) = N(X_j^t, X_j^{t-1} A, \Psi_j^t), \tag{7}$$

$$g_z(Z_i^t, m_i^t = j, X^t) = N(Z_i^t, X_j^t B, \Phi_j^t), \tag{8}$$

$$g_f(f^t) = \exp^{-\frac{\lambda}{2}\|Lf^t\|^2}. \tag{9}$$

$p(Y_j)$ and $p(m_i^t)$ are priors (uniform priors are used), and $g_x(X_j^t, X_j^{t-1})$ is dynamics, $g_z(Z_i^t, m_i^t = j, X^t)$ is data measurement term. It should be pointed that, by substituting eqn. 4 into g_x and g_z, g_x will have 5 arguments, and g_z will have 4 arguments, matched with figure 3. For simplicity and easy explanation, we will assume a simple form of Φ_j^t,Ψ_j^t : $\Phi_j^t = (\sigma_j^t)^2 I_d$, and $\Psi_j^t = (\tau^t)^2 \Sigma$. λ in eqn. 9 is the regularization parameter [19] and controls the smoothness of transform function f, and differential operator in eqn. 9 is defined as

$$\|Lf\|^2 = \iint \left[\left(\frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 \right] \tag{10}$$

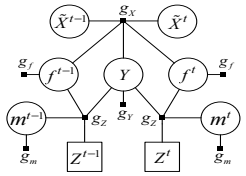

Figure 3. Factor Graph (FG) for MoTcHMMs

Actually, when optimizing the functional related to eqn. 10, the optimal f will be the well known thin-plate spline (TPS) [19]; more details will be given in section 4.

Finally our ultimate registration problem of raw edge sequence is formulated as the inference of MoTcHMMs, i.e. given the input edges Z, inferring $\tilde{X}$, Y, f and m based on eqn. 5, resulting Y will be the learned mean shape, f will be the associated non-rigid transformations, and m imply the correspondences. Details are be given in next section.

4. Inference and Optimization

Using the notions in [21], denote the hidden RVs by h and the visible RVs by v, where in MoTcHMMs they are defined as

$$h = \{(Y_j, \tilde{X}_j^t, f^t, m_i^t), j = 1, 2, ..., L; t = 1, 2, ..., N; i = 1, 2, ..., N_t\}, \tag{11}$$

$$v = \{(Z_i^t), t = 1, 2, ..., N; i = 1, 2, ..., N_t\}. \tag{12}$$

The inference of MoTcHMMs is to find a setting of h that generated the input edges v.

4.1. Inference under EM Framework

For a generative model, there might be many settings of hidden RVs h which can explain the input v well, so if possible, it is better to compute a distribution over h rather than a point estimation (maximum a posteriori) [21].

In MoTcHMMs, for the discrete RVs m, posteriori distribution is considered, however for the continuous RVs $\tilde{X}$, Y and f, it is hard to parameterize and compute its posteriori distribution (they are coupled together by eqn. 7-8), so we use point estimation as per the spirit of EM [21]. We follow the free energy formulation of EM and borrow the notations in [21] to derive the EM updating equations of MoTcHMMs.

To approximate the true posterior distribution $p(h|v)$ of MoTcHMMs, a distribution $Q(h)$ for EM is defined as

$$Q(h) = \prod_{j=1}^{L} \delta(Y_j - \hat{Y}_j) \prod_{t=1}^{N} \Big(\delta(f^t - \hat{f}^t) \prod_{j=1}^{L} \delta(\tilde{X}_j^t - \hat{\tilde{X}}_j^t | X_j^{t-1}) \prod_{i=1}^{N_t} Q(m_i^t) \Big) \tag{13}$$

where δ is Dirac delta function and related to point estimation, $\hat{Y}_j, \hat{f}^t$ and $\hat{\hat{X}}_j^t$ are corresponding optimal values, $Q(m_i^t)$ is a distribution, i.e.

$$\sum_{j=1}^{L} Q(m_i^t = j) = 1. \tag{14}$$

In order to measure the accuracy of $Q(h)$, a "free energy" with annealing is written as

$$F(Q, p, T_a) = \int_h Q(h) \ln \frac{Q(h)^{T_a}}{p(h, v)} \tag{15}$$

where $p(h, v)$ is calculated based on eqn.5, and T_a is a temperature adjusted from high to low to avoid local minima. By substituting eqn. 5 and eqn. 13 into eqn. 15 and omit constant terms, we get the finally free energy

$$\begin{aligned} F(Q, p, T_a) \propto & \sum_{t=1}^{N} \sum_{i=1}^{N_t} \sum_{j=1}^{L} Q(m_i^t = j) \left\| Z_i^t - \hat{X}_j^t B \right\|^2 / (\sigma_j^t)^2 \\ & + \sum_{t=1}^{N} \sum_{j=1}^{L} \frac{1}{(\tau^t)^2} tr\left((\hat{X}_j^t - \hat{X}_j^{t-1} A) \Sigma^{-1} (\hat{X}_j^t - \hat{X}_j^{t-1} A)^T \right) \\ & + \lambda \sum_{t=1}^{N} \left\| L \hat{f}^t \right\|^2 + T_a \sum_{t=1}^{N} \sum_{i=1}^{N_t} \sum_{j=1}^{L} Q(m_i^t = j) \ln Q(m_i^t = j) \end{aligned} \tag{16}$$

Subject to the constraint in eqn. 14, minimize $F(Q, p, T_a)$ w.r.t. $Q(m_i^t)$, we obtain the updating of $Q(m_i^t)$

$$Q(m_i^t = j) \leftarrow q_{ij}^t / \sum_{j=1}^{L} q_{ij}^t, \tag{17}$$

where $q_{ij}^t = \exp\left(- \left\| Z_i^t - \hat{f}^t(\hat{Y}_j) \right\|^2 / T_D^t \right)$ (T_a and σ_j^t are merged into one compact temperature parameter T_D^t which controls the softness of $Q(m_i^t)$). Actually the updating of $Q(m_i^t)$ in eqn. 17 is the true posterior probability $p(m_i^t = j | \hat{f}^t, \hat{Y}_j, Z_i^t)$ of m_i^t under current temperature T_D^t, which gives a soft assignment from edge point Z_i^t to the transformed mean shape.

Directly minimizing $F(Q, p, T_a)$ w.r.t. Y or f is intractable, as many terms are coupled together. Using the same divide-and-conquer fashion in [13], the remaining optimization problems can be split into two slightly simpler sub-problems: trajectory tracking SP_1 and shape registration SP_2 (Solutions are presented in section 4.2 and 4.3 respectively), i.e.

Initialize f, Y and $Q(m)$
Initialize T_D^t and λ
Begin: Deterministic Annealing
  Update $Q(m)$: Local Softassign
  Estimate σ_j^t: Enable computing eqn. 18
  $\sigma_j^t = \sqrt{\frac{\sum_{i=1}^{N_t} Q_i^t(m=j) \| Z_i^t - \hat{f}^t(\hat{Y}_j) \|^2}{\sum_{i=1}^{N_t} Q_i^t(m=j)}}$
  Solve SP_1: Trajectory Tracking
  Solve SP_2: Groupwise Shape Registration
    Update Y: Estimate mean shape
    Update f: Determine associated transformations
  Decrease T_D^t and λ
End

Table 1. Algorithm

$$\begin{aligned} SP_1 : \min_{\hat{X}} \sum_{j=1}^{L} \sum_{t=1}^{N} \Bigg(& \sum_{i=1}^{N_t} Q(m_i^t = j) \left\| Z_i^t - \hat{X}_j^t B \right\|^2 / (\sigma_j^t)^2 \\ & + tr\big((\hat{X}_j^t - \hat{X}_j^{t-1} A) \Sigma^{-1} (\hat{X}_j^t - \hat{X}_j^{t-1} A)^T \big) / (\tau^t)^2 \Bigg), \end{aligned} \tag{18}$$

$$SP_2 : \min_{f, Y} \sum_{t=1}^{N} \left(\sum_{i=1}^{L} \left\| f^t(Y_j) - \hat{X}_j^t B \right\|^2 + \lambda \left\| L \hat{f}^t \right\|^2 \right). \tag{19}$$

The two sub-problems SP_1, SP_2 and the updating scheme of $Q(m_i^t)$ (eqn. 17) can be explained intuitively as: at current EM iteration, previously registered shape result (i.e. previous transformed mean shape $f^t(Y_j)$) is used to locally update the current soft assignment $Q(m_i^t)$. This assignment may be wrong due to outliers and miss data. Therefore temporal information over all frames is subsequently explored to correct them, by solving trajectory tracking problem SP_1, hence their smooth temporal trajectories are obtained; however these trajectories may still be entangled. Accordingly, mean shape Y and spatial smooth transformation f^t is used to comb the trajectories in each frame via solving shape registration problem SP_2, thus completing the current iteration. The complete EM algorithm for MoTcHMMs is shown in Table 1.

4.2. Analytic Viterbi for Trajectory Tracking, SP_1

In fact, problem SP_1 corresponds to the inference of c-HMM. As Viterbi algorithm (dynamic programming) is used commonly to infer the best state sequence for discrete-state HMM (d-HMM) [22], we consider its continuous version of c-HMM for trajectory optimization.

Since our objective function in eqn. 18 is a quadratic form of target variable $\tilde{X}$, thus for each trajectory j ($j =$

$1, 2, ..., L$), minimizing energy E w.r.t. $\hat{X}_j$, we can get

$$\frac{\partial E}{\partial \hat{X}_j^t} = a_j^t \hat{X}_j^{t-1} + b_j^t \hat{X}_j^t + c_j^t \hat{X}_j^{t+1} - d_j^t = 0, t = 1, 2, ..., N \tag{20}$$

and its matrix form is

$$\begin{bmatrix} b_j^1 & c_j^1 & & & & \\ a_j^2 & b_j^2 & c_j^2 & & & \\ & a_j^3 & \cdots & \cdots & & \\ & & \cdots & \cdots & c_j^{N-2} & \\ & & & a_j^{N-1} & b_j^{N-1} & c_j^{N-1} \\ & & & & a_j^N & b_j^N \end{bmatrix} \begin{bmatrix} \hat{X}_j^1 \\ \hat{X}_j^2 \\ \cdot \\ \cdot \\ \hat{X}_j^{N-1} \\ \hat{X}_j^N \end{bmatrix} = \begin{bmatrix} d_j^1 \\ d_j^2 \\ \cdot \\ \cdot \\ d_j^{N-1} \\ d_j^N \end{bmatrix}. \tag{21}$$

Efficient solution to SP_1 can be obtained by solving a set of linear equation (eqn. 21) in following steps:

1) Recursion: for row t ($t = 2, 3, ..., N$), eliminate lower band a_j^t in the band matrix by subtracting with row $t-1$.

2) Termination: solve $\hat{X}_j^N$ using linear algebra.

3) Trajectory backtracking: recursively track back to previous row t ($t = N-1, N-2, ..., 1$), eliminate upper band in the band matrix and solve $\hat{X}_j^t$ using linear algebra.

As this way is similar to Viterbi algorithm for d-HMM (c.f. page 264 in [22]), we call it analytic Viterbi algorithm. Due to space limitation, details of exact equation are omitted. In the inference point of view, this way can also be viewed as a belief propagation (BP) process (c.f. section IV-C in [20]), but it is more concise especially enabling us to consider high order c-HMM straightforwardly in future.

4.3. Solution to Shape Registration, SP_2

Under the regularization in eqn.10, minimizing the functional in eqn. 19 w.r.t. function f, we get a TPS

$$f^t(X) = C^t X + D^t + \sum_{k=1}^{L} w_k^t \phi\big(\|X - Y_k\|\big), \tag{22}$$

with TPS kernel $\phi(r) = r^2 \ln r$. A least-squares approach can be applied to solve C^t, D^t and w_k^t , details are referred to [13].

Minimizing the energy in eqn. 19 w.r.t. Y_j, we get

$$\sum_{t=1}^{N} \frac{\partial [f^t(Y_j)]}{\partial Y_j} \Big(f^t(Y_j) - \hat{X}_j^t B \Big) = 0. \tag{23}$$

By substituting eqn.22 into eqn. 23, and its simpler form is

$$\begin{aligned} &\sum_{t=1}^{N} [C^t]^T . \Big(C^t Y_j + D^t + \\ &\sum_{k=1}^{L} w_k^t \phi\big(\big\|Y_j^{old} - Y_k^{old}\big\| - \hat{X}_j^t B\Big) = 0. \end{aligned} \tag{24}$$

Finally the updating of Y is

$$\begin{aligned} Y_j \leftarrow &\Big(\sum_{t=1}^{N} [C^t]^T C^T \Big)^{-1} \sum_{t=1}^{N} [C^t]^T . \\ &\Big(\hat{X}_j^t B - D^t - \sum_{k=1}^{L} w_k^t \phi(\big\|Y_j^{old} - Y_k^{old}\big\|) \Big). \end{aligned} \tag{25}$$

During annealing EM iteration, mean shape undergoes gradually updating, thus Y^{old} is used to get a simpler form in eqn. 24.

5. Outlier and Missing Data Handling

In analytic Viterbi algorithm, all point instances $\hat{X}_j^t$ at frame t of the trajectory j are considered together to give the global optimization results of the trajectory tracking problem SP_1 (eqn. 18 and eqn. 21). If one point instance is entrapped to outliers or far neighbors (due to missing data), its location will be inconsistent with others thus temporal continuity will be broken. Consequently, by solving eqn. 21, all these points are pulled back and a smooth trajectory is obtained, thus yielding a mechanism for handling outlier and missing data.

However, too much temporal constraint may introduce a bias in $\hat{X}_j^t$. In order to control it, temporal parameter τ^t in eqn. 18 is designed as

$$\frac{1}{(\tau^t)^2} = \alpha \frac{N_t}{L} \frac{NL}{\sum_t \sum_j (\sigma_j^t)^2} = \alpha \frac{N N_t}{\sum_t \sum_j (\sigma_j^t)^2}. \tag{26}$$

During EM iteration, α is decreased to a small value with the passage of time (e.g. initially set as 1 and decrease to 0.05), thus towards the end, τ^t has a very tiny bias effect for normal data. But it will retain its power against missing data; suppose edge points corresponding to $\hat{X}_j^t$ are missed at frame t, then $\sum_i \frac{Q(m_i^t = j)}{N_t / L} \to 0$. Although τ^t possess small value at the end, it is still significant w.r.t. $\sum_i Q(m_i^t = j)$, therefore the second term (temporal constraint) inside sum in eqn. 18 will dominate and missing data is still handled.

Since τ^t will be very small in the end, outliers must be removed ASAP, otherwise it will make trouble without τ^t's watch. A reweighting procedure during EM iteration is used in our work to remove outliers. When τ^t still has its power, outliers will be given a low weight, thus it's no chance for them in the later iterations. Because point instance $\hat{X}_j^t$ couldn't be entrapped to outliers under the protect of τ^t, thus outliers will far away from all point instances, so a low weight can be given to them according to the distance between them and point instances.

6. Experiments

Three situations are considered in our experiment (corresponding to dancing ballerina, talking face, and walk-

ing person, respectively), two are presented in the paper. Below we introduce the situations and corresponding data sets, along with the generation of input edge sequence. Complete results with vivid animation are available at *http://media.cs.tsinghua.edu.cn/ dhj/cvpr07/*.

Moving camera: Ballet sequence is used for testing of this case. Video stabilization is first applied and then simple intra-frame subtraction result is used for moving object detection and used as ROI for edge sequence generation.

Simple environment: Face capturing environment is considered, a fixed rectangle is used as ROI.

6.1. Parameters Setting

In the initialization of EM for MoTcHMMs, transformation f for each frame is set as identical transform, initial soft assignment $Q(m)$ is set as equivalent probability, and mean shape Y can be random initialized under the help of deterministic annealing. At the end of each EM iteration, regularization parameter λ and temperature parameter T_D^t are decreased under exponential annealing scheme, i.e. $\lambda \leftarrow \lambda^{old}\xi$, $T_D^t \leftarrow {T_D^t}^{old}\xi$, where ξ is annealing rate and is set to 0.9 in our experiment. Initial value of λ is not crucial based on our experimental observation and set to 5 always. The key parameter is the temperature parameter T_D^t, one needs a careful manual setting when apply the work of [13][14]. We present an empirical way to set T_D^t automatically: ${T_D^t}^{init} = var(Z^t)/\varepsilon = \frac{1}{\varepsilon N_t}\sum_{i=1}^{N_t}\left\|Z_i^t - \bar{Z}^t\right\|^2$, $\varepsilon \approx 6 \sim 10$, where ε isn't a critical value, and is fixed to 8 for all our experiments.

Only L (the number of points in mean shape) is needed to set according to input in our experiment. L implies the complexity of the object's shape, i.e. how many points are needed to represent it. It is easy to set L, just give a sufficient number, e.g. 250 for face sequences, and 200 for ballet sequences.

6.2. Capabilities of Our Algorithm

The color convention used for below figures is, black dots depict the input edges, green stands for outliers and red represents the attained registration. A challenging ballet sequence is used for testing and figure 4 shows the input edge maps with significant non-rigid deformations. Results are presented in figure 5, respective estimated correspondences and deformation field are superimposed on the raw edges. Despite the ambiguity present in cluttered edges, a meaningful mean shape is obtained. Outliers and massive loss of data in this sequence is also handled successfully by our algorithm. Details are shown in figure 6, there is a massive loss of edges for left arm, but we still can attain a good registration. The work of [14] can't deal with this sequence because of missing data and outliers; in their algorithm TPS was folded during iteration, and finally failed.

Figure 4. Input example of a ballet sequence

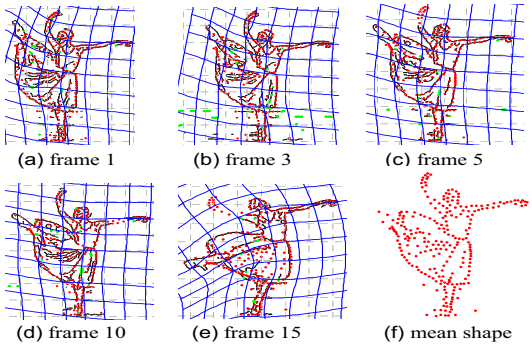

Figure 5. Results of ballet sequence

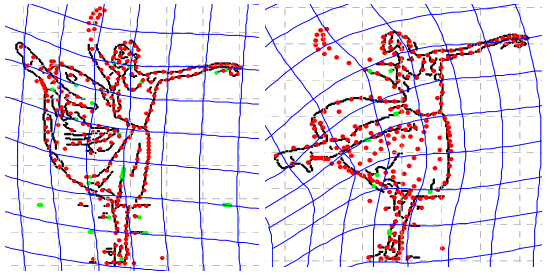

Figure 6. Missing data handling

6.3. Face Alignment

Our algorithm can be used for automatic face alignment from vide sequence (see figure 1 and figure7). Raw input edge maps along with registration results are shown in figure 7a-c, face images superimposed by estimated correspondence points are presented in (d)-(f) and (g) shows the estimated mean shape. To validate the result of our edge level face alignment, estimated transformations are used to warp all input face images into the common coordinate of mean shape, and generate a mean face image (shown in figure 7h). For comparison, an overlay image is generated by simple averaging of face images (i.e. without any warp) and is shown in figure 7i. The mean image is significant less blurred than the overlay image due to our edge based registration and this result can give a sufficient initialization for further image based registration and be used to build appearance model.

7. Conclusions and Future Work

This paper presents a novel approach for groupwise shape registration which is performed directly on raw edge sequence rather than pre-segmented training shapes. It en-

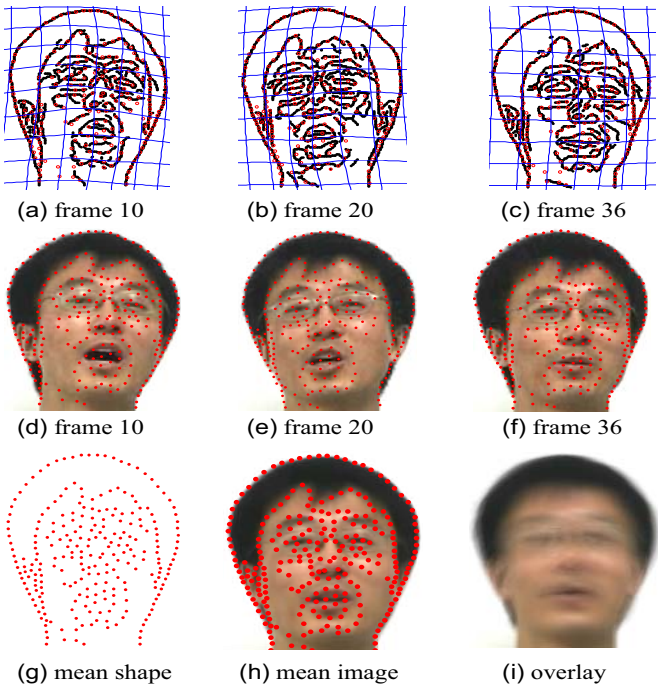

Figure 7. Face alignment at edge level by our algorithm

ables fully automatic scheme for shape modeling in future. We introduce the view point of trajectory tracking into shape registration and they are combined by a spatio-temporal generative model: mixture of transformed continuous-state HMMs (MoTcHMMs). Our algorithm is tested on real video sequences of a dancing ballerina, talking face, and walking person; their mean shape, correspondences among complete edge sequence and associated non-rigid transformations can be jointly estimated from cluttered edges.

Currently, we assume one video segment as input to make situation not so hard and also to make a compact discussion of main idea. In future, we will consider how to automatically segment a long video into pieces and perform shape registration for each one or multiple ones. High order HMM can also be considered. We will apply our work for automatic face modeling and other related applications such as recognition and object tracking.

8. Acknowledgement

This research was supported by NSFC Project 60673189. We would like to thank Haili Chui and Anand Rangarajan for sharing their code. We are grateful for the CVPR reviewers for their constructive comments.

References

[1] T.F. Cootes, D. Cooper, C.J. Taylor, and J. Graham. Active Shape Models—Their Training and Application. *CVIU*, 61(1):38–59, 1995.

[2] A. Blake and M. Isard. *Active Contours*. Springer, 1998.

[3] K.N. Walker, T.F. Cootes, and C.J. Taylor. Automatically Building Appearance Models from Image Sequences using Salient Features. In *BMVC*, 1999.

[4] CAVIAR benchmark datasets. *http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1*.

[5] A. Yilmaz and M. Shah. Actions Sketch: A Novel Action Representation. In *CVPR*, 2005.

[6] A. Veeraraghavan, R. Chellappa, and A.K. Roy-Chowdhury. The Function Space of an Activity. In *CVPR*, 2006.

[7] R.H. Davies, C.J. Twining, T.F. Cootes, J.C. Waterton, and C.J. Taylor. A Minimum Description Length Approach to Statistical Shape Modelling. *IEEE Trans. Medical Imaging*, 21(5):525–537, 2002.

[8] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *PAMI*, 24(4):509–522, 2002.

[9] S. Manay, D. Cremers, B. Hong, A. Yezzi, and S. Soatto. Integral Invariants for Shape Matching. *PAMI*, 28(10):1602–1618, 2006.

[10] A. Rangarajan, J.M. Coughlan, and A.L. Yuille. A Bayesian Network Framework for Relational Shape Matching. In *ICCV*, 2003.

[11] S. Xiang, F. Nie, and C. Zhang. Contour Matching Based on Belief Propagation. In *ACCV*, 2006.

[12] Y. Zheng and D. Doermann. Robust Point Matching for Non-rigid Shapes by Preserving Local Neighborhood Structures. *PAMI*, 28(4):643–649, 2006.

[13] H. Chui and A. Rangarajan. A New Point Matching Algorithm for Non-rigid Registration. *CVIU*, 89(2-3):114–141, 2003.

[14] H. Chui, A. Rangarajan, J. Zhang, and C.M. Leonard. Unsupervised Learning of an Atlas from Unlabeled Point-Sets. *PAMI*, 26(2):160–172, 2004.

[15] Y. Tsin and T. Kanade. A Correlation-based Approach to Robust Point Set Registration. In *2004*, ECCV.

[16] B. Jian and B.C. Vemuri. A Robust Algorithm for Point Set Registration Using Mixture of Gaussians. In *ICCV*, 2005.

[17] F. Wang, B.C. Vemuri, and A. Rangarajan. Groupwise Point Pattern Registration Using A Novel CDF-based Jensen-Shannon Divergence. In *CVPR*, 2006.

[18] D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, Inc., 2003.

[19] T. Poggio and F. Girosi. Networks for Approximation and Learning. *Proceedings of the IEEE*, 78(9):1481–1497, 1990.

[20] F.R. Kschischang, B.J. Frey, and H. Loeliger. Factor Graphs and the Sum-Product Algorithm. *IEEE Trans. Information Theory*, 47(2):498–519, 2001.

[21] B.J. Frey and N. Jojic. A Comparison of Algorithms for Inference and Learning in Probabilistic Graphical Models. *PAMI*, 27(9):1–25, 2005.

[22] L.R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.