A PROBABILISTIC MODEL FOR ROBUST FACE ALIGNMENT IN VIDEOS

Wei Zhang[1], Yi Zhou[2], Xiaoou Tang[2], and Junhui Deng[1]

[1]Department of Computer Science and Technology,
Tsinghua University, Beijing, 100084, China
w-z02@mails.tsinghua.edu.cn

[2]Microsoft Research Asia, 49 Zhichun Road, Beijing,
100080, China
{t-yizhou, xitang}@microsoft.com

***Abstract-* A new approach for localizing facial structure in videos is proposed in this paper by modeling shape alignment dynamically. The approach makes use of the spatial-temporal continuity of videos and incorporates it into a statistical shape model which is called Constrained Bayesian Tangent Shape Model (C-BTSM).Our model includes a prior 2D shape model learnt from labeled examples, an observation model obtained from observation in the current input image, and a constraint model derived from the prediction by the previous frames. By modeling the prior, observation and constraint in a probabilistic framework, the task of aligning shape in each frame of a video is performed as a procedure of MAP parameter estimation, in which the pose and shape parameters are recovered simultaneously. Experiments on low quality videos from web cameras are provided to demonstrate the robustness and accuracy of our algorithm.**

1. RELATED WORK

Localizing facial structure in images or videos is generally a preliminary step for many vision tasks, such as pose estimation, non-rigid motion tracking, and face recognition. And in the vision literature, this problem elicits many visual tracking algorithms for videos and shape alignment algorithms for images.

1.1. *Probabilistic treatment for object tracking*

Tracking targets in videos has been widely studied as a time series inference problem [3][5], in which probabilistic models are used to formulate the temporal information from previous frames and the observation in the current frame along the time line. A typical visual tracking system is described by two probabilistic models: one is a dynamic model and the other is a measurement model. Specifically, the dynamic model predicates the possible states of the tracked object with some kind of assumptions on the temporal continuity in videos, and the measurement model assesses these possible states by incorporating the observation in the current frame. Thus, the object tracking problem becomes a process of generating and then verifying hypotheses.

1.2. *Statistical shape models for alignment in images*

Among shape alignment algorithms for images, statistical shape models [1][6][7][8] have state of the art performance. The statistical models are capable of incorporating a prior density learnt from a set of shape examples into shape registration. In these models, the Bayesian Tangent Shape Model (BTSM) [8] provides a probabilistic framework for shape registration and proposes a set of regularization rules under this framework. With a prior model learnt from examples and an observation model derived from the current image, BTSM solves the problem of aligning a shape structure in an image as a MAP parameter estimation process, by which the underlying shape representation and the pose of the current shape are recovered simultaneously. BTSM shows more robustness and better accuracy with comparison to other alignment algorithms.

1.2.1 Prior shape model in BTSM

A face shape **x** with n landmark points is represented by a $2n$-dimensional vector $(x_1,y_1,\ldots\ldots,x_n,\ y_n)^{\mathrm{T}}$. After aligning all the training shapes to the tangent space **Φ** of the mean shape **μ**, the shape distribution is described by a PPCA model in the tangent space as (1),

$$\mathbf{x} = \boldsymbol{\mu} + \boldsymbol{\Phi}_r \mathbf{b} + \boldsymbol{\Phi}\boldsymbol{\varepsilon}\,. \qquad (1)$$

In (1), the shape parameter $\boldsymbol{b}$ has a prior distribution as $\mathrm{N}(\boldsymbol{0},\boldsymbol{\Lambda})$ .and the isotropic noise $\boldsymbol{\varepsilon}$ distributes as $\mathrm{N}\ (\boldsymbol{0},\ \sigma^2\boldsymbol{I})$.

1.2.2 Observation model in BTSM

The observation model is proposed after the local update step [8] in BTSM, and the observation, a $2n$-dimensional vector **y** called the observed shape, is the updated shape via local texture matching. Thus, the observation model is defined as (2),

$$\mathbf{y} = T_\gamma(\mathbf{x}) + \boldsymbol{\eta} = s\cdot\mathbf{I}_N \otimes \begin{pmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{pmatrix}\mathbf{x} + \mathbf{1}_N \otimes \begin{pmatrix}c_1\\ c_2\end{pmatrix} + \boldsymbol{\eta}\,, \qquad (2)$$

where s is the scale, θ is the rotation, $(c_1,\ c_2)$ is the translation, and $\otimes$ is the kronecker product; $\mathrm{T}\gamma(\cdot)$ is the $2D$ similarity transformation and $\gamma=(s\cdot\cos\theta,\ s\cdot\cos\theta,\ c_1,\ c_2),$; η is the Gaussian image noise, distributed as $\mathrm{N}(\boldsymbol{0},\ \rho^2\boldsymbol{I})$.

2. A PROBABILISTIC APPROACH TO MODELING SHAPE ALIGNMENT DYNAMICALLY IN BTSM

Although face alignment has been well studied for image applications, its application in videos is still an open problem as how to explore the face dynamics from the spatial-temporal continuity in videos remains a key issue to solve. From video-sequences, not only the information of the tracked object can be obtained, but also the temporal continuity can provide a robust representation. In this paper, we propose a Bayesian approach to incorporate the temporal information in model-based shape registration: Constraint Bayesian Tangent Shape Model (C-BTSM). Our model has its merits in two aspects: first, with the usage of face dynamics, the algorithm runs more stably than only implementing alignment algorithm frame-by-frame independently; secondly, the shape prior

0-7803-9134-9/05/$20.00 ©2005 IEEE

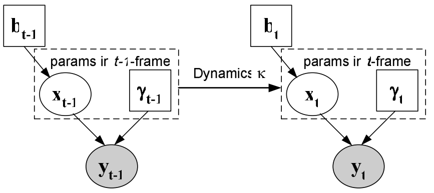

Fig. 1: Temporal constraint in face alignment

learnt from examples, which is not used in traditional feature point tracking algorithms [4][5], provides a robust prior knowledge to regularize the tracked objects in our method.

2.1. Temporal constraint in videos

Applying alignment algorithms frame-by-frame independently in a video sequence might suffer from the inconsistent and unstable results in continuous frames due to the non-linearity property of the alignment methods and the noise in videos. However, such a problem can be alleviated by the temporal continuity from video sequences, so how to incorporate it into alignment methods is fundamental for applying shape registration in videos. In this section, we propose a Bayesian approach to formulate the dynamic evolution of model parameters into the framework of shape registration.

We use the operator κ to denote one of the robust feature point tracking methods [4]. Then, from the previous aligned result $\boldsymbol{x}_{t-1}$, we can get the predicted possible configuration of $\boldsymbol{x}_t$ with the tracker κ. A random noise $\boldsymbol{\zeta}$ is used to measure the discrepancy resulted from both the frame-to-frame difference and the error of κ. Thus, the prediction model is like (3),

$$\mathbf{x}_t = \kappa\left(\mathbf{x}_{t-1}\right) + \zeta . \tag{3}$$

Fig. 1 shows the correlation between two continuous frames we formulate in the prediction model.

A good property of Bayesian formulation is that it is easy to combine extra constraints as long as these constraints can be modeled with probabilities. BTSM provides such a Bayesian formulation, allowing extra constraints to be applied. In this section we formulate the temporal constraints in the BTSM framework. The prediction model can be formulated into BTSM as a constraint model like Fig. 2. The constrained shape variable *z* is exactly the predicted shape configuration $\kappa(\boldsymbol{x}_{t-1})$ from previous frames. However, in the practical application of our algorithm, we allow the constrained shape to be a variable set of feature points, for example, the top *50%* accurately tracked points. Under that condition, *z* is a *2m*-dimensional vector where *m* is the number of selected constrained points. We can see that, in constrained BTSM, the constrained shape *z* incorporates more assumption on the underlying shape $\boldsymbol{x}$ and the pose $\boldsymbol{\gamma}$. And this assumption makes use of the temporal information in videos.

2.2. C-BTSM

From the predicted shape $\kappa(\boldsymbol{x}_{t-1})$ and prediction model in (3), we can select a set of valuable feature points $\{u_i, v_i\}_{i=1:m}$ of $\kappa(\boldsymbol{x}_{t-1})$ with higher confidence. Denote $z=(u_1,v_1,u_2,v_2,\ldots,u_m, v_m)^T$ as the *2m*-dimensional constrained shape vector. Then, the constraint model in Fig. 2 can be written as:

$$\mathbf{Q}^T T_\gamma(\mathbf{x}) = \mathbf{z} + \zeta . \tag{4}$$

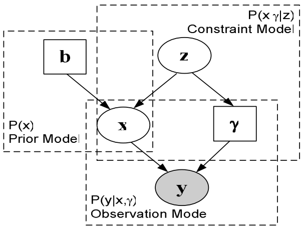

Fig. 2: Graph model for Constraint BTSM

In (4),

1) $\boldsymbol{Q}$ is the *2m*x*2n* matrix related to the constrained shape vector, which indicates the linear relationship between the tangent shape and the constrained shape: i.e. if the *j*th feature point of $\boldsymbol{x}$ is constrained, the *2j* and *2j+1* columns of the matrix $\boldsymbol{Q}$ are correspondently set to be

$$\mathbf{Q}_{\cdot 2j} = \Big(0,\ldots,0,\underset{2j}{1},0,\ldots,0\Big)^T, \mathbf{Q}_{\cdot 2j+1} = \Big(0,\ldots,0,\underset{2j+1}{1},0,\ldots,0\Big)^T ;$$

2) In the constraint model, the estimation error $\boldsymbol{\zeta}$ of the constrained points consist of two parts: the point localization error in previous frame and the points tracking error in the inter-frame motion. Here we combine the two types of error into Gaussian white noise N~ $(0, \delta^2\boldsymbol{I})$:

$$\delta^2 = \frac{1}{m} dist\left(T_{\gamma_{old}} \left(\mathbf{Q}^T \mathbf{x}_{old}\right) - \mathbf{z} \right). \tag{5}$$

Here the estimation error δ^2 is measured by the average point-to-point distance between the estimation result and the constrained shape. Using Euclidean distance for variance measurement is from the intuition that the regularization shape will be stably closed to the constrained points while the estimation error will also converge to zero. The $\boldsymbol{x}_{\text{old}}$ and $\boldsymbol{\gamma}_{\text{old}}$ are the old values in the last iteration of EM algorithm. The EM algorithm will be described in the next section.

By introducing the latent variable $\boldsymbol{x}$**,** C-BTSM connects the prior shape model with both image observation and constrained shape. The likelihood related to $\boldsymbol{x}$ (the latent shape variable) consists of two components: the shape likelihood between *z* (prediction shape) and $\boldsymbol{x}$; the shape likelihood between $\boldsymbol{x}$ and $\boldsymbol{y}$ (the observation shape). The prior model and likelihood model between observation shape and tangent shape are the same as BTSM [8].

3. E-M Inference For Parameter Estimation

After it is formulated by the graphical model in Fig. 2, the shape alignment problem turns to be a parameter estimation problem, that is, given the observation $\boldsymbol{y}$ we need to compute the MAP estimation of p($\boldsymbol{b},\boldsymbol{\gamma}$|$\boldsymbol{y}$). We use the EM algorithm to optimize this posterior. And we first derive the expected log-posterior of all the parameters ($\boldsymbol{b}, \gamma$) given all the variables ($\boldsymbol{x}$, $\boldsymbol{y}$, *z*) which will be optimized directly in each iteration of the EM algorithm.

According to Fig. 2, the posterior of all the parameters ($\boldsymbol{b}$, γ) given all the variables ($\boldsymbol{x}$, $\boldsymbol{y}$, *z*) is factorized as (6). p($\boldsymbol{b}$) is the prior probability of the shape parameter $\boldsymbol{b}$, which is a Gaussian distribution N($\boldsymbol{0}$,$\boldsymbol{\Lambda}$), and the conditional probabilities p($\boldsymbol{y}$|γ,$\boldsymbol{x}$), p($\boldsymbol{x}$|$\boldsymbol{b}$) and p($\boldsymbol{x}$,γ|*z*) are defined by the observation model (2), the prior model (1) and the constraint model (4) respectively.

$$p\left(\mathbf{b},\boldsymbol{\gamma}\,|\,\mathbf{x},\mathbf{y},\mathbf{z}\right)=p\left(\mathbf{b}\,|\,\mathbf{x}\right)p\left(\boldsymbol{\gamma}\,|\,\mathbf{x},\mathbf{y}\right)p\left(\boldsymbol{\gamma}\,|\,\mathbf{x},\mathbf{z}\right). \tag{6}$$

Thus, from (6), we can get the expected log-posterior of all the parameters given the complete data in (7),

$$\begin{aligned}\mathrm{L}\left(\mathbf{b},\boldsymbol{\gamma}\,|\,\mathbf{b}_{old},\boldsymbol{\gamma}_{old}\right)&=\left\langle\log p\left(\mathbf{b},\boldsymbol{\gamma}\,|\,\mathbf{x},\mathbf{y},\mathbf{z}\right)\right\rangle\Big|_{p\left(\mathbf{x},\mathbf{z}|\mathbf{y},\mathbf{b}_{old},\boldsymbol{\gamma}_{old}\right)}\\ &=-1/2\Big\{\mathbf{b}^T\boldsymbol{\Lambda}^{-1}\mathbf{b}+\sigma^{-2}\left\langle\left\|\mathbf{x}-\boldsymbol{\mu}-\boldsymbol{\Phi}_{\mathbf{r}}\mathbf{b}\right\|^2\right\rangle+\rho^{-2}\left\langle\left\|\mathbf{y}-\mathbf{T}_{\gamma}\left(\mathbf{x}\right)\right\|^2\right\rangle\quad.\\ &\quad+\delta^{-2}\left\langle\left\|\mathbf{z}-\mathbf{Q}^T\mathbf{T}_{\gamma}\left(\mathbf{x}\right)\right\|^2\right\rangle\Big\}+const\end{aligned} \tag{7}$$

Computing the L-function in (7) is equivalent to calculate two sufficient statistics $\langle \boldsymbol{x}\rangle$ and $\langle\|\boldsymbol{x}\|^2\rangle$.

3.1. E-Step

The expected tangent shape $\langle \boldsymbol{x}\rangle$ has different forms in the constraint subspace $\boldsymbol{Q}$ and its orthogonal complement $\boldsymbol{H}$, but the common thing is that they are both a weighted average of several shape parts. Then, we decompose x as

$$\left\langle\mathbf{x}\right\rangle=\left\langle\mathbf{Q}\mathbf{x}_Q+\mathbf{H}\mathbf{x}_H\right\rangle.$$

And each part of $\langle \boldsymbol{x}\rangle$ and $\langle\|\boldsymbol{x}\|^2\rangle$ are computed as (8),

$$\begin{aligned}&\left\langle\mathbf{x}_Q\right\rangle=\mathbf{Q}^T\left[\boldsymbol{\mu}+p_{Q1}\boldsymbol{\Phi}_r\mathbf{b}+p_{Q2}\boldsymbol{\Phi}\boldsymbol{\Phi}^T T_{\gamma}^{-1}\left(\mathbf{y}\right)+p_{Q3}\left(\mathbf{Q}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\mathbf{Q}^T\right)T_{\gamma}^{-1}\left(z\right)\right],\\ &\left\langle\mathbf{x}_H\right\rangle=\mathbf{H}^T\left[\boldsymbol{\mu}+p_{H1}\boldsymbol{\Phi}_r\mathbf{b}+p_{H2}\boldsymbol{\Phi}\boldsymbol{\Phi}^T T_{\gamma}^{-1}\left(\mathbf{y}\right)\right],\\ &\left\langle\left\|\mathbf{x}\right\|^2\right\rangle=\left\langle\mathbf{x}\right\rangle^T\left\langle\mathbf{x}\right\rangle+\left(\sigma^{-2}+s^2\rho^{-2}+s^2\delta^{-2}\right)\end{aligned} \tag{8}$$

where the weights are

$$p_{Q1}=\sigma^{-2}\left(\sigma^{-2}+s^2\rho^{-2}+s^2\delta^{-2}\right)^{-1},$$

$$p_{Q2}=s^2\rho^{-2}\left(\sigma^{-2}+s^2\rho^{-2}+s^2\delta^{-2}\right)^{-1},$$

$$p_{Q3}=s^2\delta^{-2}\left(\sigma^{-2}+s^2\rho^{-2}+s^2\delta^{-2}\right)^{-1},$$

$$p_{H1}=\sigma^{-2}\left(\sigma^{-2}+s^2\rho^{-2}\right)^{-1}\quad\text{and}\quad p_{H2}=s^2\rho^{-2}\left(\sigma^{-2}+s^2\rho^{-2}\right)^{-1}.$$

Thus, from (8), we can see that in the constraint subspace Q, the shape representation not only weighs the reconstructed PCA shape $\boldsymbol{\Phi_r b}$ and the projected observed shape $\boldsymbol{\Phi\Phi}^T\mathbf{T}_\gamma^{-1}(\boldsymbol{y})$, but also the projected constrained shape $\boldsymbol{Q\Phi\Phi}^T\boldsymbol{Q}^T\mathbf{T}_\gamma^{-1}(\boldsymbol{z})$.

3.2. M-Step

Given the expected tangent shape $\langle \boldsymbol{x}\rangle$, in the M-step, the shape parameter **b** and pose parameter γ are maximized as,

$$\mathbf{b}_{new}=\sigma^{-2}\left(\sigma^{-2}\mathbf{I}+\boldsymbol{\Lambda}^{-1}\right)^{-1}\boldsymbol{\Phi}_r^T\left\langle\mathbf{x}\right\rangle, \tag{9}$$

$$\boldsymbol{\gamma}_{new}=\left(\rho^{-2}\left\langle\mathbf{X}^T\mathbf{X}\right\rangle+\delta^{-2}\left\langle\mathbf{X}_Q^T\mathbf{X}_Q\right\rangle\right)^{-1}\left(\rho^{-2}\left\langle\mathbf{X}^T\right\rangle\mathbf{y}+\delta^{-2}\left\langle\mathbf{X}_Q^T\right\rangle\mathbf{z}\right),$$

where $X=(\boldsymbol{x}, \boldsymbol{x}^*, \boldsymbol{e}, \boldsymbol{e}^*)$ and $\boldsymbol{X_Q}=(\boldsymbol{x}_q, \boldsymbol{x}_q^*, \boldsymbol{e}_q, \boldsymbol{e}_q^*)$. The * operator means rotating a shape by *90°* and $\boldsymbol{e}=(1,0,1,0,\ldots1,0)^T$. (see [8] for details.)

4. EXPERIMENTS

In this section, we use a series of experiments to demonstrate the accuracy and stability of C-BTSM when the constraints of temporal continuity and consistency in videos are utilized. The shape model and local texture models, which are used in ASM, BTSM and C-BTSM, are trained from *721* manually labeled face images in the FERET and AR databases [2]. In order to quantitatively compare the accuracy and stability of different algorithms, we generate *138* video sequences with the *3D* face models in USF Human ID *3D* database [7]. For each video sequence, we apply different illumination condition via adding different surrounding lighting, and the head pose changes from *0* to *20* degrees in the direction of out-plane rotation. And all frames of each video have labeled ground truth for evaluation. The result of a real video captured from a USB web camera is also shown in Fig. 3.

4.1. Constrained points selection

We select a set of constrained points from the aligned shape in the previous frame based on the confidence we calculate for each feature point. And the confidence is composed of two parts: one is texture difference measured by using the local texture model in the ASM [6] and BTSM [8], and the other is the sum of residual error around the local region around the tracked feature points (see [4] for details). The confidence is then evaluated by weighing the two measurements:

$$confidence(p)=p_{alignment}+\lambda * p_{tracking}. \tag{10}$$

In.(10), the λ is a scalar that tells how big the item of tracking confidence is. In our experiments, we choose λ as a value between *0.5* and *1.0*, and select the top *25%* points of higher confidence as constrained points.

4.2. Alignment accuracy

The accuracy of our method is investigated by three experiments. In the first experiment, we evaluate the effect of frame-to-frame constraints by comparing the alignment errors of BTSM and C-BTSM. And the alignment error is calculated as the average of point-to-point distance between the aligned shape and the labeled ground truth. In another experiment, we evaluate the effect of the constraint accuracy on the alignment accuracy. The last experiment investigates the effects of the number of constrained points on the alignment accuracy. As the result, we draw the conclusions as: the constraints improve the alignment accuracy; and moreover, the more accurate the constraints are or the more constrained points, the more accurate the alignment results are.

Fig. 4 shows the alignment errors of BTSM and C-BTSM in a videos sequence. In the video, the head pose of a person changes from *0* to *20* degrees in the out-plane direction. From the figure we can see that C-BTSM outperforms BTSM especially on the points of eyebrows parts (highlight region). This is mainly because the similar intensity distribution in the around area couldn't be well discriminated in the local texture model while it can be tracked efficiently, so the unstable eye-brow region alignment can be improved with a firm constraint from tracking points. In our experiment, the average improvement of C-BTSM over BTSM is about *4* pixels. However if we concentrate on the eyebrows parts, we can see a larger improvement of C-BTSM over BTSM: it is about *9-10* pixels.

From the derivation in section *3*, we can see the alignment error is related to the estimation of the constraint error. Fig. 5 demonstrates the distribution of constraint error calculated from138 video sequences: the mean constraint error in

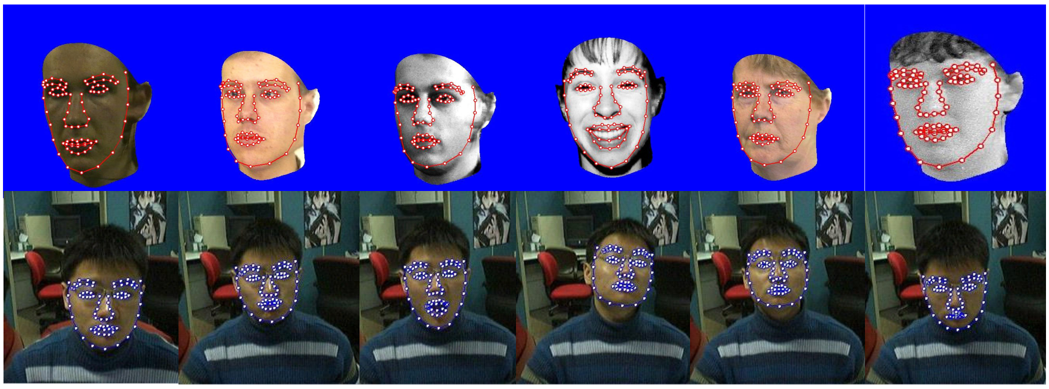

Fig. 3: Results of the intermediate frames by C-BTSM

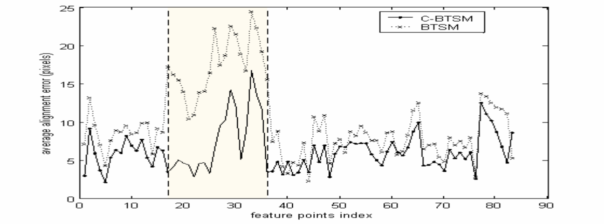

Fig. 4: Alignment error of BTSM and C-BTSM

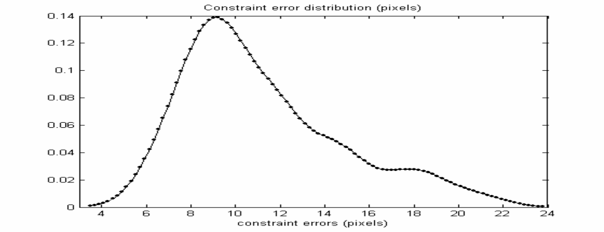

Fig. 5: Constraint Error statistics in C-BTSM

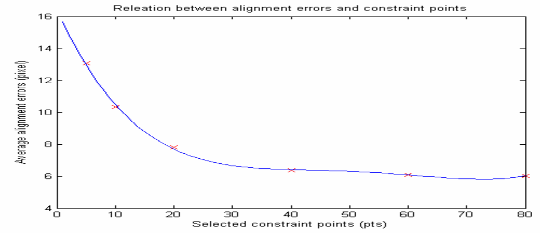

Fig. 6: Alignment error with different constraint points

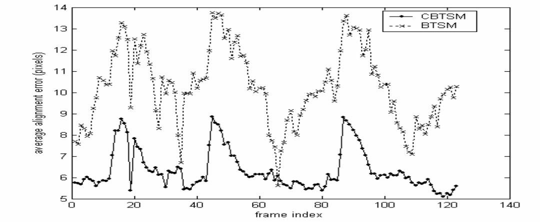

Fig. 7: Average alignment error of BTSM and C-BTSM frame by frame

our experiment is about *9* pixels. Fig. 6 shows the alignment error of the different number of constraint points.

4.3. Alignment Stability

Stability is very essential for the algorithms used in videos. And the main purpose of C-BTSM algorithm is to improve the stability of the alignment results of the continuous frames In order to show the stability of BTSM and C-BTSM algorithms, we plot the alignment errors of each frame in a video for both algorithms. Based on Fig. 7, the maximum of the alignment errors in C-BTSM does not exceed *10* pixels, which is much smaller than that in BTSM. Since the head pose changes between *0* to *20* degrees, we can see a periodicity of *40* frames for the alignment error change. And because our training data are mainly the frontal faces, the alignment error might become larger when the view deviates from the frontal view. Aligned results for the C-BTSM in videos are shown in Fig. 3 for an intuitive understanding of the stability.

5. Conclusion And Discussion

In this paper, we propose C-BTSM to extend the usage of model-based image alignment algorithm into videos by combining the temporal continuity and consistency. As an extension of BTSM, the C-BTSM model has a set of regulation rules similar to those in BTSM, like the shrinking function in the PCA subspace and the weighted representation of the underlying shape. From the derivation in section 3, C-BTSM degenerates to BTSM[8] when the estimation errorδ^2 goes to infinity. Through using the constraints of previous information, C-BTSM shows more stability and better accuracy in videos than BTSM. Moreover, C-BTSM can also be used to combine other feature localization techniques with BTSM. If compared with the traditional optical flow-based tracking algorithms, C-BTSM is more robust to the environmental change because the prior shape model provides a global constraint to formulate non-rigid facial motion.

The C-BTSM algorithm converges after a few iterations on a single frame and its speed in videos is close to real time. For a *320*240* video stream, the C-BTSM algorithm runs at speed of *10* fps on a Pentium *IV-2.7*GHz machine.

Reference

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

[2] A.M. Martinez and R. Benavente. The AR Face Database.*CVC Technical Report* #24, 1998.

[3] G. Welch and G. Bishop. "*An introduction to the kalman filter* (tutorial)". *ACM SIGGRAPH*, 2001.

[4] J. Shi and C. Tomasi. "Good features to track". *IEEE Int. conf. on Computer Vision and Pattern Recognition*, 1994.

[5] M.Isard and A.Blake. "CONDENSATION conditional density propagation for visual tracking". *IJCV 29(1)*, pp.5--28, 1998.

[6] T.F. Cootes and C.J. Taylor. "*Technical Report: Statistical models of appearance for computer vision"*. http://www.isbe.man.ac.uk/~bim/refs.html.

[7] V. Blanz and T. Vetter. "A morphable model for the synthesis of 3D-faces". *ACM SIGGRAPH*, 1999.

[8] Y. Zhou, L. Gu, and H.J. Zhang. "Bayesian Tangent Shape Model: Estimating Shape and Pose Parameters via Bayesian Inference", *CVPR*, 2003.