FITTING AND TRACKING 3D/4D FACIAL DATA USING A TEMPORAL DEFORMABLE SHAPE MODEL

Shaun Canavan, Xing Zhang, and Lijun Yin

Department of Computer Science
State University of New York at Binghamton

ABSTRACT

In this paper, we propose a novel method for detecting and tracking landmark facial features on purely geometric 3D and 4D range models. Our proposed method involves fitting a new multi-frame constrained 3D temporal deformable shape model (TDSM) to range data sequences. We consider this a temporal based deformable model as we concatenate consecutive deformable shape models into a single model driven by the appearance of facial expressions. This allows us to simultaneously fit multiple models over a sequence of time with one TDSM. To our knowledge, it is the first work to address multiple shape models as a whole to track 3D dynamic range sequences without assistance of any texture information. The accuracy of the tracking results is evaluated by comparing the detected landmarks to the ground truth. The efficacy of the 3D feature detection and tracking over range model sequences has also been validated through an application in 3D geometric based face and expression analysis and expression sequence segmentation. We tested our method on the publicly available databases, BU-3DFE [15], BU-4DFE [16], and FRGC 2.0 [12]. We also validated our approach on our newly developed 3D dynamic spontaneous expression database [17].

1. INTRODUCTION

Detecting and tracking landmark features on 3D range data is the first step toward geometric based vision research for object modeling, recognition, visualization, and understanding. Applications in this area of research include 3D face recognition and expression interpretation for biometrics and human computer interaction [6]. With the rapid development of 3D imaging technologies, 3D range data is becoming one of the most popular modalities for applications in computer vision. While research in 2D modality based tracking has produced a number of successful and widely used algorithms, such as Active Shape Model [3] and Local Binary Pattern [9], research in 3D modality based analysis still faces the challenges of 3D geometric landmark detection, 3D mesh registration, and 3D motion tracking. Therefore, there is a strong demand for novel and robust algorithms for handling 3D datasets. Morphable Model [2] is a successful algorithm for this 3D problem. Another commonly used method for registering two meshes is the Iterative Closest Point algorithm (ICP) [1].

This material is based upon work supported in part by the National Science Foundation under grants IIS-1051103, CNS-1205664, CNS-0855204, and IIS-0541044.

This method relies on finding the closest pairs of points between the two meshes being registered; however, it shows limitations in handling largely deformed mesh models. X. Lu et al. [7] developed an approach using ICP to detect the nose tip and mouth corner landmarks to help register the meshes for classification. Wang et al. [14] used key facial landmarks selected semi-automatically to segment the face and perform facial expression analysis by evaluating the principal curvatures in those segmented regions.

Active shape models (ASM) have been widely used to address the problem of landmark detection and tracking, although mainly on 2D data [3][4] or volumetric data [5] for medical data segmentation. In a 2D-based ASM, due to the lack of explicit 3D shape representation of texture data, the construction and tracking of an ASM relies on both 2D shape components and 2D texture components. The fitting process relies on a regression procedure guided by shape constraints and texture primitive constraints (e.g., edge, intensity, and color, etc.) The quality of the results, however, is limited by the accuracy of these constraints and the degree of pose variance.

Recent work has addressed the problem of fitting a deformable model to 3D range data, however, the problem of fitting and tracking 4D range data remains largely unsolved. Sun et al. [13] used active appearance models (AAM) to track features of 3D range models. However, the detection and tracking of facial features were performed on 2D videos, while the 3D features themselves were obtained by mapping the 2D features to the corresponding parts of the 3D models. Nair et al. [8] developed an approach to fit an active shape model to 3D face meshes using candidate landmarks for the inner eye corners and nose tip. Their active shape model is fit by finding a similarity transformation between the candidate landmarks of the mesh and the corresponding landmarks within their active shape model. Perakis et al. [11] compared candidate landmarks on the face mesh, obtained through shape index calculation, to their 3D active shape model. However, there is no single fitting or temporal fitting process for finding candidate landmarks. Zhao et al. [18] used a patch based method to fit a statistical facial feature model to 3d range data using probability. In all of the above approaches, static 3D data was used for training and testing. None of the approaches investigate applications for 3D dynamic (aka 4D) range sequences.

In this paper, we propose a method to construct a novel 3D temporal deformable shape model (TDSM) to detect and track 3D landmarks across *3D/4D* model sequences. The new model is driven by the temporal expression appearance, and constructed from the temporal 3D point distribution, without use of textures. The temporal deformable shape models are built individually from different expression model sequences as well as multiple-frame sequences (e.g. fear, anger, happiness, sadness, surprise, and

disgust). The basic method for model fitting relies on finding the closest points in the range mesh model that correspond to an instance of the TDSM, where an instance is defined as a sampling along the modes of variation in the TDSM. Whether a set of landmarks is considered an acceptable candidate for a good fit is determined by the variance of the TDSM weighted matrix. For each adaptation from the TDSM to the range mesh surface we compute a distance score between the newly detected landmarks and the original instance of the TDSM. The lowest score is considered the best fit to the range mesh model.

Our primary contribution is the advancement of a 2D active shape model to a 3D temporal deformable shape model for fitting 3D and *4D range data*. The use of a TDSM allows us to adapt multiple input mesh models with a single temporal deformable shape model through one pass of the process. A TDSM also allows for inter-frame constraints on the fitting process with 4D data. This is due to the nature of any expression behaving from neutral, to onset, to peak, to offset, and back to neutral. To validate the efficacy of the TDSM, we apply the detected landmarks for both subject identification and expression classification on multiple public databases. We also evaluate the accuracy of 3D landmark detection through applications of 3D video segmentation.

2. 3D TEMPORAL DEFORMABLE SHAPE MODEL

3D range data exhibits shapes of facial surfaces explicitly. This shape representation provides a direct match with the 3D active shape model due to its inherent and explicit shape representation in 3D space. In considering this property, our feature detection and tracking algorithm can rely solely on 3D geometric shape without assistance of any texture information, thus resulting in less sensitivity to pose and lighting variations.

To take advantage of this property, we would like to model the shape variation, as well as the implicit shape constraints imposed between consecutive frames in a sequence. Given a training set of M mesh models each with N annotated landmarks, the data is separated into 3 groups consisting of the neutral, onset, or peak phases of a given expression. To construct a temporal point distribution model (TPDM), a parameterized model, S, is constructed where $S = P_1^1, \dots, P_{N,}^1 P_1^2, \dots, P_N^2, \dots, P_N^k$. P_i^k is the i^{th} landmark of the k^{th} model, where $P_i^k = (x_i^k, y_i^k, z_i^k)$ and $1 \le k \le M$ (M is the total number of training models). To construct this model, alignment of the training landmarks must be performed. To do so a modified version of Procrustes analysis is used [3].

Procrustes analysis determines a linear transformation that aligns two sets of points (shapes). It minimizes the distance D, which is a minimized summation of the squared errors. Once alignment has been performed, principal component analysis (PCA) is then performed on the aligned feature vector. This is done to estimate the different variations of all the training data in a $k \times N \times 3$ dimensional space. For PCA, each shape deviation from the mean is calculated as

$$ds_i = s_i - \bar{s}. \quad (1)$$

Then the covariance matrix C is calculated:

$$C = \frac{1}{N}\sum_{i=1}^{N} ds_i ds_i^T \quad (2)$$

This equation yields the modes of variation, V, of the training shapes along the principal axes. Given V and a weight vector, w, that controls the shape; we can approximate any temporal shape from the training data by:

$$S = \bar{s} + Vw. \quad (3)$$

The weight vector, w, allows us to generate new samples by varying its parameters within certain limits. These limits are imposed to ensure only valid shapes are constructed (*i.e.* a correct facial expression). For the temporal deformable models we constrain the allowable shapes to be within 2 standard deviations from the mean, giving us:

$$-2\sqrt{\lambda_i} \le \sqrt{b_i} \le 2\sqrt{\lambda_i} \quad (4)$$

Where λ_i is the i^{th} eigenvalue of C. When approximating a new shape S, the temporal point distribution model is constrained not only by the variations in shape but also by the inter-frame constraints that the consecutive frames impose. Given a k-frame TDSM, k consecutive input mesh models are ensured to vary in a manner that is consistent with the TDSM. For example, we can assume that during the course of an expression, facial appearance is developed gradually. If we have a frame displaying a neutral expression at the start of the sequence, the next frame cannot display the peak of the expression, as there needs to be some form of the onset of the expression before the peak occurs. The feature vector will not allow the shape to have a neutral expression next to the peak. Therefore, during the adaptation of the TDSM, if we come across k mesh models that do not vary in a way that is consistent with our TDSM, we attribute this to noises or other anomalies. Figure 1 shows an example of a k-frame TDSM where N=83.

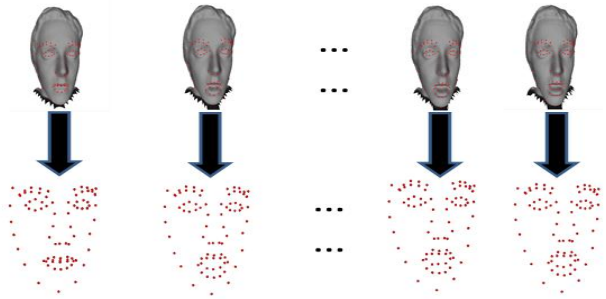

Fig.1 Example illustration showing a k-frame TDSM where N=83. Top row shows fit mesh models, bottom row shows visual representation of TDSM vector.

3. FITTING/TRACKING 3D/4D RANGE DATA

3.1. Fitting 3D Range Data Using A TDSM

When dealing with static 3D range data we can construct a TDSM where k=1, allowing us to fit a single frame in the absence of a sequence of frames. To fit the TDSM to 3D range data we create an off-line table of weight vectors (w) that will control the shape of the TDSM, each with a uniform amount of variance. We have chosen to instantiate the weight vectors off-line as this gives us more control over which shapes are constructed and to help ensure the new shapes are consistent within the allowable shape domain. Having this offline weight vector also allows us to speed up the fitting process as we can quickly find the instance of the TDSM that gives us the best fit.

Once we have created each of the instances of the TDSM, they are then fit to the 3D input data. This is done by finding which vertex in the range mesh model corresponds to the closest point of each landmark in the TDSM instance. *We are able to do a simple closest point search as the final Procrustes distance will be large if good points have not been found. This is a key difference between*

the TDSM and an active shape model. The TDSM does not require an explicit initialization phase as each fit is based on pure geometric data as well as the inter-frame constraints that every k-frames impose. To find the closest points in the model, instead of using a brute force search each mesh is constructed as a k-d tree to speed up computation time. After we find the closest points for all landmarks in the model, we then determine if the newly detected landmarks for the TDSM instance correspond to an allowable shape based on the constraint that the weight vector (w) must fall within 2 standard deviations from the mean. To do this, we must transform our detected landmarks into the model parameter space by constructing a new w vector. Since equation (3) gives us $S = \bar{s} + Vw$, we can then find the corresponding w vector of the detected landmarks by the following:

$$w = V^T(S - \bar{s}). \quad (5)$$

We then compare this new w vector with the allowable domain. If it is within this range it is accepted as a candidate model that will give us the best fit for the 3D range data. If it is outside of this domain it is discarded for the range mesh we are trying to fit. For the candidates that are acceptable, the TDSM instance that gave us these candidate landmarks, as well as the candidate landmarks themselves are saved. Once this is complete, each candidate model has a distance score computed between the newly detected landmarks and the TDSM instance. This distance score is the Procrustes distance, which is a metric used to determine the shape difference between two objects. Given the original instance of our TDSM $m_1 = (x, y, z)$ and the detected landmarks on the range data $m_2 = (u, v, w)$, the Procrustes distance can be defined as:

$$D = \sum_{i=1}^{N} \sqrt{(u_i - x_i)^2 + (v_i - y_i)^2 + (w_i - z_i)^2}. \quad (6)$$

We find the Procrustes distance for each TDSM and its corresponding candidate landmarks on the range data for all candidate models. The smallest D value is considered the best fit. We are able to quickly find the smallest D due to our offline vector w, as the computation is linear in terms of the number of landmarks as shown in (6). Table 1 summarizes this algorithm.

Table 1. Algorithm of 3D landmark detection.

$C_x(m1) = closest\ point\ of\ m1\ to\ X$

Construct w vectors off-line
Construct model instances from available w vectors
for each input range mesh do
for each model instance do
for each landmark in model do
$S_i = C_x(m1_i)$
end for
$w = V^T(S - \bar{s})$
if$(-2\sqrt{\lambda_i} \leq \sqrt{b_i} \leq 2\sqrt{\lambda_i})$
save model
else then
discard model
end if
end for
for each saved candidate do
for each landmark in model do
$D_s = \sum_{i=1}^{N} \sqrt{(u_i - x_i)^2 + (v_i - y_i)^2 + (w_i - z_i)^2}$
end for
end for
Select shape model with smallest D for best fit
end for

Figure 2 shows sample frames from the fitting process. Figure 3 shows examples from the BU-3DFE database [15] with best and worst fits of the tracked points with comparison to the ground truth which are manually picked points. Note that the distance measure is very high for the worst fit. Figure 4 shows the feature detection and tracking after the pose rotations in roll, pitch, and yaw, illustrating robustness to pose variation.

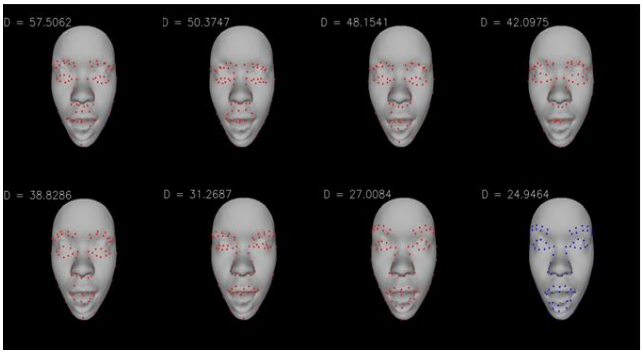

Fig. 2. Sample frames from fitting process (k=1). Higher D values show poor fits, lowest D selected as best fit (in blue).

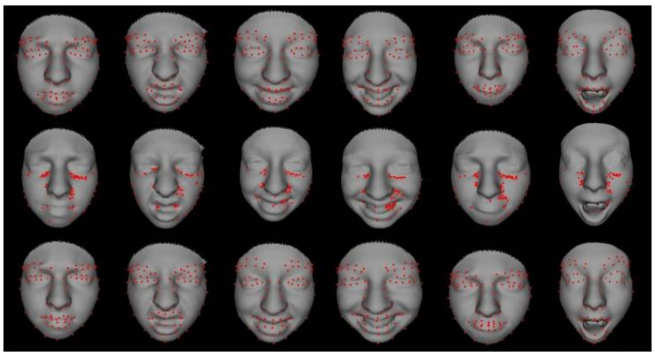

Fig. 3. Top row: best fit, Middle: worst fit, bottom: ground truth.

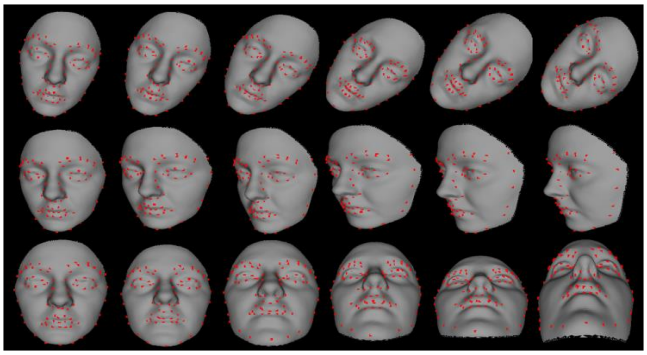

Fig. 4. TDSM fit on models displaying roll, yaw, and pitch.

3.2. Fitting 4D Range Data Using a TDSM

Given an input sequence of M frames we can also detect and track landmarks using a TDSM where (k > 1, so-called multi-frame TDSM). Similar to the algorithm in Table 1, we extend the w vector to the length kxN where k is the number of mesh models and N is the number of landmarks. Each of the k mesh models to be fit is also represented as a k-d tree. The search is once again the

closest points on each of the mesh models. However, instead of searching for all kxN landmarks in the TDSM for k mesh models, the TDSM is still searched using N landmarks for each individual model. Then, the k-frame TDSM (with N=83) is applied using the same criteria as in algorithm 1 in Table 1. Since any expression exhibits in the form of five durations: neutral, onset, peak, offset, and neutral, we can define the k-frame TDSM based on the samples of these durations. For example, in this implementation, we define a multi-frame TDSM with k=2. Given an expression with durations from neutral, to onset, to peak, to offset, and back to neutral, we construct 8 TDSMs for each expression, which is a combination of two frames in different durations. The 8 TDSMs where $S_i = \{S_1, \dots, S_8\}$ are neutral to neutral, neutral to onset, onset to onset, onset to peak, peak to peak, peak to offset, offset to offset, and offset to neutral. Note that such an inter-frame relationship (or temporal constraint) makes the landmark detection across multiple frames occur simultaneously and accurately. Such a relationship is applicable to any expression with any speed. In other words, the multi-frame TDSM can handle variable speed expressions. The temporal constraint can filter out some impossible cases (e.g. neutral-peak, onset-offset, etc.), thus resulting in a consistent fitting. Any violation of the inter-frame relationship will cause a large fitting error.

Using the 8 TDSMs we fit each one to k=2 frames in the expression sequence to find the best fit. A sample surprise expression sequence from the BU-4DFE database [16] can be seen in Fig. 5. Shown in Fig. 6 are sample frames from the BU-4DFE [16] and our newly developed 4D spontaneous expression database [17] in the second and bottom rows respectively.

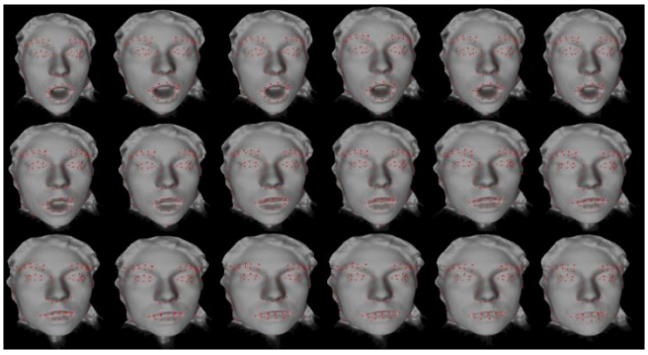

Fig. 5. Sample Sequence fit with k=2 TDSM.

4. EXPERIMENTS AND EVALUATION

4.1. Databases

Four face databases have been used for our study, including two 3D static model databases (BU-3DFE [15] and FRGC 2.0 [12]) and two 3D dynamic (4D) model databases (BU-4DFE [16] and our newly developed 4D spontaneous expression database [17]. Our new spontaneous database consists of 41 subjects (56% female and 44% male), each consisting of 10 different spontaneous expression sequences. The expressions are elicited activities including film watching, interviews, experiencing cold pressor test, and others. Ten different spontaneous expressions are evoked (joy, embarrassing, surprise, disgust, nervous, scared, sad, pain, upset, sympathetic). Each task could have multiple expressions or mixed emotions. The database includes the 3D dynamic model sequences, texture videos, and annotated action units (AU). Figure 6 (bottom row) shows an example of the database (details are described in [17]). Table 2 lists details on each database.

Table 2. Database Summaries.

DB	Modality	Type	#Sub	#Exp	#Models
3DFE	Static	Deliberate	100	7	2500
4DFE	Dynamic	Deliberate	101	6	606 Seq.
FRG C 2.0	Static	Deliberate	466	2	932(select)
4D Spon. Exp.	Dynamic	Spontan.	40	10	240 seq.

Figure 6 also shows the examples of FRGC 2.0 [12] (top row), BU-4DFE [16] second row, and BU-3DFE database [15] (third row).

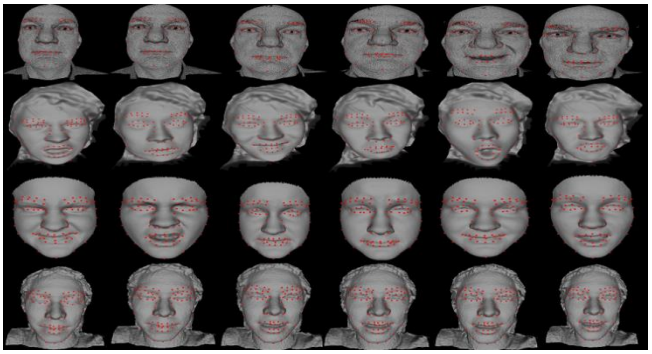

Fig. 6. Top row: FRGC 2.0(k=1), second row:BU-4DFE(k=2), third row:BU-3DFE(k=1), bottom row:4D spontaneous database(k=2). (Note: the tracked feature points are overlapped on the model sequences).

4.2. Evaluation on Accuracy of Feature Points Tracking

4.2.1. Error Statistics

To evaluate the accuracy of the TDSM fitting algorithm we calculate the error between the fit landmarks and manually selected ground truth. To do so, we calculate the mean square error between the two sets of landmarks. We define the one-point spacing as the closest pair of points on the 3D scans (0.5mm on the geometric surface). If we treat the unit error being equivalent to 1 point-spacing, the mean error can be computed by the average of point differences between the two sets. The average errors on the four databases are listed in Table 3. As can be seen from this table, the average fitting errors are much less when k > 1, this can be attributed to the extra temporal constraints that are imposed when k > 1. In the cases of the BU-4DFE and 4D spontaneous expressions databases k=2.Figure 7(a) shows the error statistics (average error and standard deviation for each of 83 key points) of the BU-3DFE database. Figure 7(b) shows the error statistics for the BU-4DFE database.

Table 3. Average error in points spacings.

Database	BU-3DFE	BU-4DFE	FRGC 2.0	4D Spontaneous Expressions
Average Error	5.6	1.5	6.7	1.6

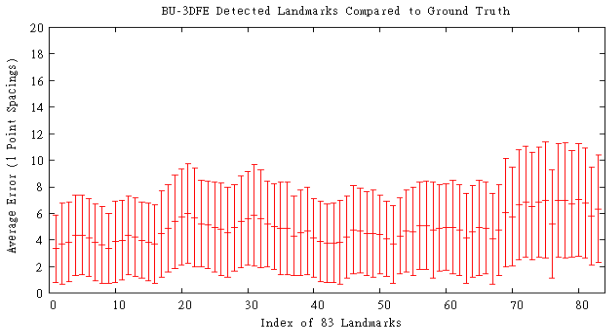

Fig. 7(a). Error statistics of 83 landmarks (BU-3DFE).

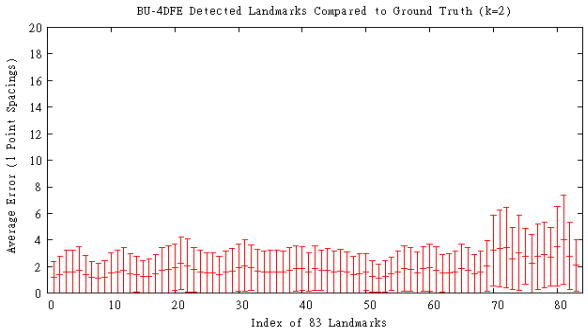

Fig. 7(b). Error statistics of 83 landmarks (BU-4DFE).

4.2.2. Error Statistics

We have also compared our result of MSE of the average point spacings to the work reported in [13]. Our result of MSE on BU-4DFE is 3.7, which shows a significant improvement over the result of 6.25 reported in [13]. Figure 8 shows the average errors on each of 83 points using our approach and the approach in [13].

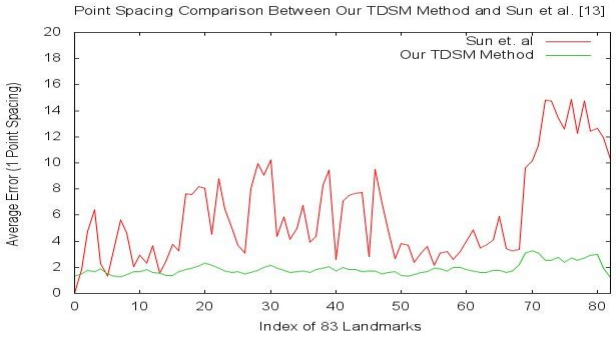

Fig. 8. Comparison with Sun et al. [13].

In addition, we have also compared our results to the work reported by Nair *et al* [8] on the BU-3DFE database. Following their method we selected four landmarks (inner and outer eye corners) to compare to the ground truth. We achieved an error rate of 0.09 as compared to their rate of approximately 0.44. Figure 9(a) shows our mean normalized error, and Figure 9(b) shows the normalized error of [8]. The evaluation shows that our feature tracking approach outperforms [8] as our approach does not rely on candidate landmarks to guide the fitting.

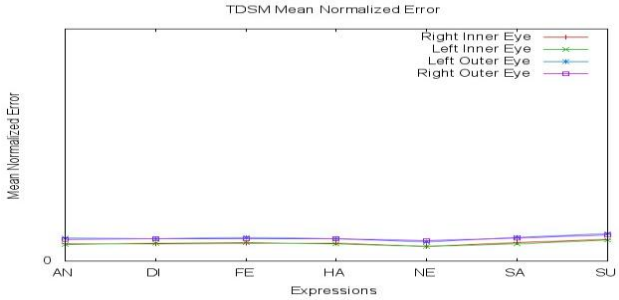

Fig. 9(a). Mean normalized error of our TDSM method. (Key: AN=angry, DI=disgust, FE=fear, HA=happy, NE=neutral, SA=sad, SU=surprise).

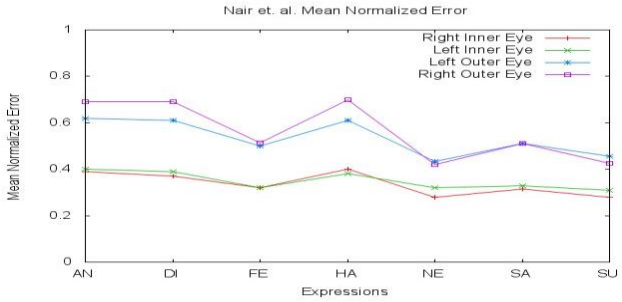

Fig. 9(b). Mean normalized error of Nair et. al [8].

4.3. Subject and Expression Verification

To validate our proposed method, we apply it to subject verification and facial expression classification problems. We are able to compare each set of detected landmarks to the original instances of the TDSM, which is the same method as described in Section 3. Given Equation (6), we can find the distance *D* from the detected landmarks to each of TDSM instances. The smallest *D* value must correspond to a minimum threshold for a correct classification. As each of the TDSMs has been labeled as a subject, expression, and constraint (neutral, onset/offset, or peak), this verification and classification process is realized simultaneously.

The BU-3DFE database consists of 4 levels of each expression for each subject, while the BU-4DFE and our 4D spontaneous expression databases both consist of sequences of multiple frames for each subject and expression. A majority voting strategy is implemented to help verify the subject and classify the expression. Given *n* frames that correspond to an expression of a subject, the subject is verified to be subject Y if it is voted as subject Y among the majority of those *n* frames. The expression is also classified as expression E if it is voted as expression E among the majority of those *n* frames. Experimental results for both subject verification and expression classification are as follows.

4.3.1. Subject Verification

Using the BU-3DFE database with n=4, we achieved an approximate subject verification rate of 94%. Using the BU-4DFE database, when k=2 and n=20, the verification rate is increased to 98%. This increase can be attributed to two major factors: (1) the lower average error of using a multi-frame TDSM, and (2) the applied majority voting strategy. Begin able to use a larger n with the BU-4DFE database, for verifying a subject, allows us to remove noise and exclude high distance scores.

4.3.2. Face Expression Classification

Using the BU-3DFE and BU-4DFE databases the six prototypic facial expressions are classified at the accuracy of 87% and 98% respectively. This increase in correct classification rates can also be attributed to expression based TDSM construction, lower error rates of the detected landmarks, and the applied majority voting strategy. Table 4 shows the confusion matrix of six expressions classification on BU-3DFE database.

Table 4. Expression verification confusion matrix: BU-3DFE.

	Angry	**Disgust**	**Fear**	**Happy**	**Sad**	**Sur-prise**
Angry	**90%**	2%	0%	0%	8%	0%
Disgust	8%	**87%**	1%	2%	2%	0%
Fear	6%	6%	**80%**	5%	3%	0%
Happy	5%	3%	5%	**87%**	0%	0%
Sad	14%	1%	1%	0%	**83%**	1%
Surprise	4%	1%	0%	0%	3%	**92%**

4.4. Expression Segmentation (Action/Non-Action)

A natural extension of our proposed method is the application of expression segmentation (or facial event detection) across a sequence of facial models. Each of the TDSM instances has the information of a subject, an expression, and the inter-frame constraint label. For the purposes of expression segmentation, we classify the results into one of two categories: either an action (onset/offset, and peak), or non-action (neutral expression). Comparing the fit landmarks with the TDSM instances, we are able to determine whether the current input model is performing an action or a non-action. Given the fit points of the mesh model, we compare them with the instances of our TDSM. The smallest *D* value from these comparisons is used as a measure for the classification. For example if our smallest *D* corresponds to the model classified as non-action, then the mesh model is classified as such. To analyze these results we manually segmented sequences from the BU-4DFE (see supplemental material for tracked and segmented sequence from this database) and 4D spontaneous expression databases and compared the automatic segmentation with this ground truth data. We achieved 86% and 81% correct classification rates in terms of action vs. non-action segmentation across all expressions for the BU-4DFE and 4D spontaneous expressions databases respectively. Note that the data is very challenging for all expressions in the spontaneous database. Figure 10 illustrates an example of segmentation on this database.

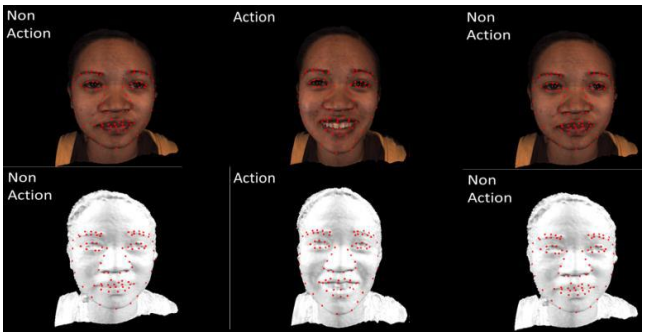

Fig. 10. An example of spontaneous expression segmentation (action/non-action) on the 4D spontaneous expression database.

5. DISCUSSION AND FUTURE WORK

In this paper, we have presented a new 3D temporal deformable shape model for both detecting and tracking key landmarks on 3D range mesh models. We have evaluated the accuracy of the feature detection and validated its utility for subject verification and expression classification in multiple public databases. We have also validated its utility for expression segmentation.

In our future work, we plan to develop a method for estimating the direction of motion for the landmarks by including 3D edge information. The proposed 3D TDSM is, in principle, extendible to other object types with 3D/4D mesh representation. Our future work will also include the evaluation on 3D feature detection on other geometric mesh databases.

6. REFERENCES

[1] Besl, P., McKay, N., A method of registration of 3d shapes. IEEE Trans. on PAMI vol. 14, pp. 239-256, 1992.

[2] Blanz, V., Vetter, T., A morphable model for the synthesis of human haves. SIGGRAPH, 1999.

[3] Cootes, T. Taylor, C. Cooper, D. Graham, J., Active shape models-their training and application. CVIU vol. 61 pp. 18-23, 1995.

[4] Cootes, et al. Active shape models: Eval. of a multi-res. approach method for improving the image search. BMVS 1994.

[5] De Bruijne et al., Adapting active shape models for 3d segmentation of tubular structures in medical images. IPMI 2003.

[6] Kakadiaris, I.A., Passalis, G., Toderick, G., Murtuza, M.N., Lu, Y., Karampatzikas, N., Theoharis, T., Three-dimensional face rec. in the presence of facial expressions: An annotated deformable model approach. IEEE Trans. on PAMI 29 2007.

[7] Lu, X., Jain, A., Automatic feature extraction for multiview 3d face recognition. IEEE Conf. FGR, 2006.

[8] Nair, P., Cavallaro, A., 3d face detection landmark localization, and registration using a point distribution model. IEEE Trans. on Multimedia (11), 2009.

[9] Ojala, T., Pietkained, M., Maenpaa, T., Multiresolution gray-scale and rotation invariant texture classification with local binary pattersn. IEEE Trans. on PAMI, vol. 24, 2002.

[10] Ong, E. and Bowden, R., Robust facial feature Tracking using shape-constrained multiresolution-selected linear predictors, IEEE Trans. on PAMI, 33(9):1844-1859, 2011.

[11] Perakis, P., Passalis, G., Theoharis, T., Toderici, G., Kakadaris, I., Partial matching of interpose 3d facial data for face rec. BTAS 2009.

[12] Phillips, P., Flynn, P., Scruggs, T., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, J., Overview of the face rec. grand challenge. IEEE CVPR 2005.

[13] Sun, Y., Chen, X., Rosato, M., Yin, L., Tracking vertex flow and model adaption for 3d spatio-temporal face analysis. IEEE Trans. on SMC A vol. 40 2010.

[14] Wang, J., Yin, L,. Wei, X., Sun, Y., 3d facial expression recognition based on primitive surface feature distribution. CVPR 2006.

[15] Yin, L., Wei, X., Sun, Y., Wang, J., Rosato, M., A 3d facial expression database for facial behavior research. FGR 2006.

[16] Yin, L., Chen, X., Sun, Y., Worm, T., Reale, M., A high-res 3d dynamic facial expression database. FGR 2008.

[17] Zhang, X., Yin, L., Cohn, J, et al., A High Resolution Spontaneous 3D Dynamic Facial Expression Database. FGR 2013.

[18] Zhao,. X et al., Accurate Landmarking of Three-Dimensional Facial Data in the Presence of Facial Expressions and Occlusions Using a Three-Dimensional Statistical Facial Feature Model. IEEE Trans. SMC Part B, 41(5): 1417-1428, 2011.