Using the EKF for 3D Morphable Model Parameter Estimation

Nathan Faggian, Andrew P. Paplinski
Monash University, Australia
Faculty of Information Technology, Clayton
{nathanf,app}@mail.csse.monash.edu.au

Jamie Sherrah
Clarity Visual Intelligence
jsherrah@clarityvi.com

Abstract

Estimating the structure of the human face is a long studied and difficult task. In this paper we present a new method for estimating facial structure from only a minimal number of salient feature points. The presented method uses the Extended Kalman Filter (EKF) to regress 3D Morphable Model (3DMM) shape parameters and solve rigid body motion using a simplified camera model. A linear method for initializing the recursive filter is provided. The convergence properties of the method are then evaluated using synthetic data. The method is then demonstrated for both single image shape recovery and shape recovery during tracking.

1 Introduction

To accurately determine the fine three-dimensional shape of an object using only temporal information is a difficult and well studied task. Refered to as structure from motion there are many methods, both in the linear and nonlinear domain. This paper presents a recursive filter approach to the problem using an Extended Kalman Filter (EKF.) Through the combination of two EKF's we are able to track across variation in pose and identity. Real-time operation is a complication that is also addressed by our method since it is limited to extrapolation from a minimal number of salient features and the filters are a series of non-complex operations.

2 Background

There are a number of popular methods to compute the motion and structure of the human face either from a single view or from a series of them. Recently there has been much development of methods that combine salient feature tracking and 3D structure estimation as one step [12, 6, 1]. Generally these methods lead to a search for the full set of parameters of a model and its camera matrices. Given this trait such methods are dubbed parametric methods. Such methods are well suited to tracking if the object is known, they restrict the object to a certain domain and can produce highly accurate results.

As it stands there are already a number of parametric methods. Baker and Matthews focus specifically on the extension of the popular Active Appearance Model (AAM) of Cootes et al [5]. They called the extension the 2D+3D AAM [12], where the key difference is a computationally efficient fitting strategy called ICIA. In their method they first track a person specific AAM and then build a 3D Morphable Model using non-rigid factorization. This means that Baker and Matthews method require a two step process of tracking and model construction before it can perform 3D tracking.

Dornaika and Ahlberk also presented a modified AAM fitting algorithm [6]. This was applied to the well known CANDIDE animation model using weak perspective projection. The solution demonstrated that a small generic model was capable of tracking across pose end expression using their modified directed search. They did however make use of the CANDIDE model which is not a true statistical model of shape; it is not constructed from labelled training data.

In what could be considered a continuation Anisetti et al [1], extend the forwards additive approach of Lucas and Kanade [11] to the CANDIDE fitting task. For the purpose of robust tracking they build a template from a single view, using a small number of corresponding points to the model. Using the defined template they are then successfully able to track across pose, illumination and expression. Although in order for tracking to occur Anisetti requires a 3D template to be defined.

In yet another development, Mittrapiyanuruk et al also demonstrated an accurate method for tracking rigid objects [13] across pose. Using a modified Gauss-

Newton optimization (to accommodate M-estimation) they were able to demonstrate rotational accuracy below **5** degrees for estimates of yaw, pitch and roll. Similar to Anisetti, Mittrapiyanuruk requires a user labeled 3D template.

3 Novelty

Our solution does not require hand labeling of a 3D template. It only requires a somewhat accurate tracking of the feature points, which could be provided by an AAM or other tracking algorithm. Tracking can contain missing or inaccurate results because we make use of the EKF, which allows for missing or bad data to be factored out of the fitting process. Our solution also extrapolates a higher order fitting by determining the relationship between the 3DMM and the subset of 2D feature points, this was inspired by the work of Volker Blanz [3]. Our work can therefore be considered a new method for point-based 3DMM fitting.

4 3D Morphable Models

A 3D Morphable Model (3DMM) is a statistical representation of both the 3D shape and texture of an object in a certain domain. 3DMMs are popular in the face domain [16, 14] and are used for this task in our work. A 3DMM is built from 3D laser scans of human faces, which are then put into dense correspondence [4]. Using the aligned data the 3DMM is composed of two models built using PCA for shape and texture variation:

$$\hat{\mathrm{s}} = \bar{\mathrm{s}} + \mathrm{S} \cdot \mathrm{diag}(\sigma_\mathrm{s})\mathrm{c_s} \quad \hat{\mathrm{t}} = \bar{\mathrm{t}} + \mathrm{T} \cdot \mathrm{diag}(\sigma_\mathrm{t})\mathrm{c_t} \qquad (1)$$

where $\hat{\mathrm{s}}$ and $\hat{\mathrm{t}}$ are novel $(3\mathrm{N} \times 1)$ shape and texture vectors, $\bar{\mathrm{s}}$ and $\bar{\mathrm{t}}$ are the $(3\mathrm{N} \times 1)$ mean shape and texture vectors, S and T are the $(3\mathrm{N} \times \mathrm{M})$ column (eigenvectors) spaces of the shapes and textures, σ are the corresponding eigenvalues, $\mathrm{c_s}$ and $\mathrm{c_t}$ are shape and texture coefficients. These linear equations describe the variation of shape and texture within the span of 3D training data. The coefficients $\mathrm{c_s}, \mathrm{c_t}$ are scaled by the corresponding eigenvalue $\sigma_\mathrm{s}, \sigma_\mathrm{t}$ of the eigenvector S, T. In comparison 3DMMs are similar to the 2D Active Appearance Model (AAM) [5], although 3DMM model sizes are larger and pose and illumination changes can be addressed in the 3DMM fitting process.

4.1 3DMM Fitting

One of the most accurate fitting methods for 3DMMs is the gradient descent approach [16]. This is analysis by synthesis where the coefficients are inferred from a difference between a rendered head (image) and the input image. Effectively it is the minimization of the cost function:

$$\epsilon = ||\mathrm{F}(\mathrm{c_s}, \mathrm{c_t}) - \mathrm{I}||^2 \qquad (2)$$

where F is a rendering function that when provided with shape and texture coefficients produces an image that is aligned with the input image, I. Speed is an issue for such a method, generally model coefficients are determined in a number of minutes. This was recently improved by Romdhani [14] when he introduced a fast gradient descent method. It uses a multi-feature fitting strategy to reduce the cost function quickly and robustly, although the solution is not real-time it fits in a number of seconds, not minutes.

4.2 3DMM Point Based Fitting

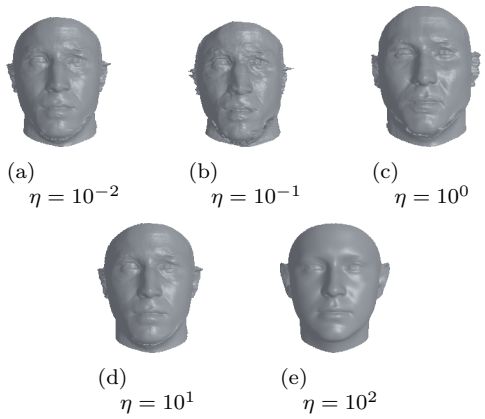
If it is possible to ignore the constraint of a photo-realistic result then a viable alternative to gradient descent methods is available. This is the recently proposed method by Blanz et al [3] which is a concise and mathematically optimal method to reconstruct a 3DMM from a sparse set of either 2D or 3D feature points. The method has two advantages 1) it relies on only linear operators and 2) operates in real-time (at the expense of model accuracy.) Using the assumption that only a small set of corresponding points are available the method minimizes the cost function:

$$\epsilon = ||\mathrm{L} \cdot \mathrm{V} \cdot \mathrm{S} \cdot \mathrm{diag}(\sigma)\mathrm{c_s} - \mathrm{r}||^2 \qquad (3)$$

where L is a camera matrix, containing the projection parameters, V is a subset selection matrix, S is the $(3\mathrm{N} \times \mathrm{M})$ column (eigenvectors) space of the training shapes, σ are the corresponding eigenvalues, $\mathrm{c_s}$ are shape coefficients and r is a $(2\mathrm{P} \times 1)$ set of feature points. Solving for the coefficients is done in a statistically optimal way using the pseudo-inverse operation and a regularization term. The requirement for a regularization term (η) is a problem since it is not intuitively set. It was empirically shown on Blanz' data to be close to optimal at a value of $\eta = 0.1$, although this can not be extrapolated to different 3D data. Figure **1** demonstrates the effect of changing η, as it increases the estimate approaches the mean model.

5 EKF 3DMM Point Based Fitting

Our contribution is a new method that makes use of the well known EKF to achieve the same benefits as Blanz' method and without any regularization term



(a) $\eta = 10^{-2}$ (b) $\eta = 10^{-1}$ (c) $\eta = 10^{0}$ (d) $\eta = 10^{1}$ (e) $\eta = 10^{2}$

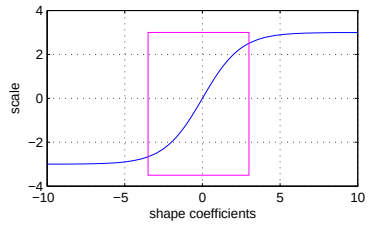
Figure 1. Variation of η, Using only 46 farkas [8] feature poins, of the mean 3D head.

and extend it to 3DMM tracking. We minimize a similar cost function:

$$\epsilon = ||\mathrm{V} \cdot \mathrm{L} \cdot \ (\bar{\mathrm{s}} + \mathrm{S} \cdot \mathrm{diag}(\sigma) \cdot \mathrm{f}(\mathrm{c_s})) - \mathrm{r}||^2 \qquad (4)$$

where L is a camera matrix, containing projection the parameters, V is a $(2\mathrm{P} \times 3\mathrm{N})$ subset selection matrix, S is the $(3\mathrm{N} \times \mathrm{M})$ column (eigenvectors) space of the training shapes, σ_s are the corresponding eigenvalues, $\mathrm{c_s}$ are shape coefficients and r is a $(2\mathrm{P} \times 1)$ set of feature points. The addition that removes the requirement for a threshold is the modified sigmoid function, $\mathrm{f}(\mathrm{c_s})$, shown in figure **2**.

$$\mathrm{f}(\mathrm{c_s}) = \frac{2}{1 - \mathrm{e}^{-\log\frac{5}{2}\cdot \mathrm{c_s}}} - 1 \qquad (5)$$



(a) scaling factor / shape coefficients

Figure 2. A graphical representation of the sigmoid function, $\mathrm{f}(\mathrm{c_S})$ (equation 5), demostrating the approximately linear scaling range between ± 3.

The sigmoid function is a way to ensure that the predicted shape coefficients are within a range of 3 standard deviations with respect to the training data. This restricts the estimate provided by our method to be within the span of the 3D shape model during tracking. The sigmoid coefficients were chosen so that the signal in the range of ± 3 was approximately linear (rectangle region in sigmoid plot.)

5.1 Kalman Filtering

Since its introduction by Rudolph Kalman in 1960 [9], the kalman filter has been applied to a great many parameter estimation taks. This work demonstrates another application for the discrete filter. The Kalman filter is composed of two components, where the first is a set of time update equations:

$$\hat{\mathrm{x}}_{\mathrm{n}} = \hat{\mathrm{x}}_{\mathrm{n-1}} + \mathrm{Bu}_{\mathrm{n-1}} \qquad (6)$$

$$\mathrm{P_n} = \mathrm{AP_{n-1}A^T} + \mathrm{Q} \qquad (7)$$

The time update equations project forward the current state ($\hat{\mathrm{x}}_{\mathrm{n}}$) and error covariances ($\mathrm{P_n}$.) This leads to the estimate of the system state which is used in the measurement update equations:

$$\mathrm{K_n} = \mathrm{P_nH^T(HP_nH^T + R)^{-1}} \qquad (8)$$

$$\hat{\mathrm{x}}_{\mathrm{n}} = \hat{\mathrm{x}}_{\mathrm{n}}\mathrm{K_n}(\mathrm{z_n} - \mathrm{H}\hat{\mathrm{x}}_{\mathrm{n}}) \qquad (9)$$

$$\mathrm{P_n} = \mathrm{I} - \mathrm{K_nHP_n} \qquad (10)$$

The measurement update equations are responsible for feedback. The equations change the apriori error covariance estimate ($\mathrm{P_n}$) and obtain an improved posteriori state estimate ($\hat{\mathrm{x}}_{\mathrm{n}}$) of the system. The Kalman gain ($\mathrm{K_k}$) is also calculated, a variable that could be considered how much to "trust" the estimate versus the measurement. Given these equations a Kalman filter is thus capable of predicting state, as well as noise, based only on measurements information.

The Extended Kalman Filter (EKF) is a linearization of the measurement and time update equations for the Kalman filter [10]. It is applied when a non linearity is present in the update or measurement equations. The difference in equations is subtle and well covered in the literature. For our work the EKF is used because the relationships are non-linear in measurement; both the projection and rotations of the 3DMM introduces the non-linearity.

5.2 Pose Estimation

Solving for three-dimensional pose and structure is accomplished, in our framework, using two EKF's. The first (pose, Ω_{p}) filter is used to solve the well known exterior-orientation problem, minimizing the cost function:

$$\epsilon = ||(\mathrm{s} \cdot \mathrm{R} \cdot \mathrm{V} \cdot \ (\bar{\mathrm{s}} + \mathrm{S} \cdot \mathrm{diag}(\sigma) \cdot \mathrm{f}(\mathrm{c_s})) + \mathrm{t}) - \mathrm{r}||^2 \qquad (11)$$

This is the task of solving the rigid body motion between the 3D model (given a set of model coefficients) and the 2D measurements, r. As a projection model we chose the scaled orthographic camera model. Such a model is a reasonable approximation of the true perspective case when the distance between the object and the camera is not small. We also chose to encode the rotation as a first order approximation, using the canonical exponential form, shown as the Rodriguez equation:

$$\mathrm{R} = \mathrm{I}_3 + \hat{\mathrm{v}}\sin\theta + \hat{\mathrm{v}}^2(\cos\theta - 1) \tag{12}$$

where $\hat{\mathrm{w}}$ is a skew symmetric matrix:

$$\hat{\mathrm{v}} = \begin{bmatrix} 0 & -\mathrm{v_z} & \mathrm{v_y} \\ \mathrm{v_z} & 0 & -\mathrm{v_x} \\ -\mathrm{v_y} & \mathrm{v_x} & 0 \end{bmatrix} \tag{13}$$

If it is assumed that rotations are relatively small then the first order approximation for rotation becomes:

$$\mathrm{R} = \begin{bmatrix} 1 & -\mathrm{v_z}\cdot\sin\theta & \mathrm{v_y}\cdot\sin\theta \\ \mathrm{v_z}\cdot\sin\theta & 1 & -\mathrm{v_x}\cdot\sin\theta \\ -\mathrm{v_y}\cdot\sin\theta & \mathrm{v_x}\cdot\sin\theta & 1 \end{bmatrix} \tag{14}$$

or in a more concise form:

$$\mathrm{R} = \begin{bmatrix} 1 & -\mathrm{w_z} & \mathrm{w_y} \\ \mathrm{w_z} & 1 & -\mathrm{w_x} \\ -\mathrm{w_y} & \mathrm{w_x} & 1 \end{bmatrix} \tag{15}$$

and in combination with the scaled projection forms the measurement equation for the EKF:

$$\begin{bmatrix} \mathrm{x_r} \\ \mathrm{y_r} \end{bmatrix} = \begin{bmatrix} \mathrm{s} & 0 & 0 \\ 0 & \mathrm{s} & 0 \end{bmatrix} \begin{bmatrix} 1 & -\mathrm{w_z} & \mathrm{w_y} \\ \mathrm{w_z} & 1 & -\mathrm{w_x} \\ -\mathrm{w_y} & \mathrm{w_x} & 1 \end{bmatrix} \begin{bmatrix} \mathrm{X_m} \\ \mathrm{Y_m} \\ \mathrm{Z_m} \end{bmatrix} + \begin{bmatrix} \mathrm{t_x} \\ \mathrm{t_y} \\ \mathrm{t_z} \end{bmatrix} \tag{16}$$

where $(\mathrm{x_r}, \mathrm{y_r})$ are a measurement point which correspond to the current iterations 3D model point $(\mathrm{X_m}, \mathrm{Y_m}, \mathrm{Z_m})$, s is scale, $(\mathrm{t_x}, \mathrm{t_y}, \mathrm{t_z})$ are translation and $(\mathrm{w_x}, \mathrm{w_y}, \mathrm{w_z})$ encode the incremental rotation of the model. After estimation, the parameters are packed into the camera matrix, L.

5.3 Structure Estimation

Structure is determined using the second (identity, Ω_I) filter to estimate the parameters for the 3DMM. These parameters span the facial structure variation that is present in our database of 3D heads. The kalman filter is used to effectively minimize the cost function shown in equation 4. Where it is assumed that the exterior orientation has been determined by the estimates from the previous (pose only) filter, thus providing L. As before the use of a sigmoid function limits the possible range of the parameters estimated to be within the span of the 3D shape model. This forms the EKF measurement equation (given one corresponding vertex, j):

$$\begin{bmatrix} \mathrm{x_r} \\ \mathrm{y_r} \end{bmatrix} = \mathrm{L}\left(\begin{bmatrix} \bar{\mathrm{X}}_\mathrm{j} \\ \bar{\mathrm{Y}}_\mathrm{j} \\ \bar{\mathrm{Z}}_\mathrm{j} \end{bmatrix} + \begin{bmatrix} \mathrm{S_{j,0}} & \cdots & \mathrm{S_{j,n}} \\ \cdots & \vdots & \cdots \\ \mathrm{S_{j+3,0}} & \cdots & \mathrm{S_{j+3,n}} \end{bmatrix} \begin{bmatrix} \sigma_0 & 0 & 0 \\ 0 & \vdots & 0 \\ 0 & 0 & \sigma_\mathrm{n} \end{bmatrix} \mathrm{f(c_s)} \right) \tag{17}$$

where $(\mathrm{x_r}, \mathrm{y_r})$ are a measurement point which correspond to the current iterations 3D model point j, $(\bar{\mathrm{X}}_\mathrm{j}, \bar{\mathrm{Y}}_\mathrm{j}, \bar{\mathrm{Z}}_\mathrm{j})$ is the mean model, $\mathrm{S_{i,j}}$ is the shape column space, σ_s are the associated eigenvalues and $\mathrm{c_s}$ is the desired set of model coefficients that represent the measurement. Once estimated $\mathrm{c_s}$ is the new 3D shape for the next fitting iteration.

5.4 Initialization

To improve both the stability and convergence properties of the method the pose filter is initialized using a simple linear approach. The method aligns the (reduced) scaled orthographic projection of the mean 3D head, $(\mathrm{V}\cdot\bar{\mathrm{s}})$, to the measurements, r. Using the same parametrization as the pose filter the linear estimate for rotation parameters can be determined using the pseudo-inverse of:

$$\begin{bmatrix} -\bar{\mathrm{X}}_\mathrm{y} & \bar{\mathrm{X}}_\mathrm{z} & 0 \\ \bar{\mathrm{X}}_\mathrm{x} & 0 & \bar{\mathrm{X}}_\mathrm{z} \end{bmatrix} \begin{bmatrix} \mathrm{w_z} \\ \mathrm{w_y} \\ \mathrm{w_x} \end{bmatrix} = \begin{bmatrix} \mathrm{x_r} \\ \mathrm{y_r} \end{bmatrix} \tag{18}$$

scale, and translation can also be computed in a similar fashion, assuming rotation is fixed:

$$\begin{bmatrix} \bar{\mathrm{X}}_\mathrm{x} & 1 & 0 \\ \bar{\mathrm{X}}_\mathrm{y} & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{s} \\ \mathrm{t_x} \\ \mathrm{t_y} \end{bmatrix} = \begin{bmatrix} \mathrm{x_r} \\ \mathrm{y_r} \end{bmatrix} \tag{19}$$

this simple method was found to work and produce stable results if iterated first to solve translation and then rotation. Most importantly for both EKFs we set the state update equations to be identity, assuming no non-linear relationship between states. For good results we also set the initial covariances both for the model and pose parameters to be very low, in the range of 10^{-5} for all estimated parameters.

5.5 DUAL EKF Algorithm

There are 6 degrees of freedom (d.o.f) in the pose filter[1], and a maximum of N-1 d.o.f for the structure

[1] Translation in z is not observable using a scaled orthographic camera model: $\frac{\partial \mathrm{x}}{\partial \mathrm{t_z}} = 0$, $\frac{\partial \mathrm{y}}{\partial \mathrm{t_z}} = 0$

estimate where N is the number of 3D heads used to construct the 3DMM. When the number of d.o.f exceed the number of feature points in a measurement vector then the system can become unstable. This was a similar problem that was solved in the work of Azarbayenjani and Pentland by using a simplified 3D representation [2]. We achieve the same result by dividing the task and forming a DUAL EKF framework.

Algorithm 1 DUAL EKF Algorithm

Require: rows(r) $> 6 + N$
  while r **do**
    $[R_n, s_n, t_n] = \Omega_P(c_{n-1}, R_{n-1}, s_{n-1}, t_{n-1})$
    {pose filter estimates the new incremental pose parameters}
    $R_n = R_n \cdot R_{n-1}$
    {incremental rotation is composed with global rotation}
    $[R_n, s_n, t_n] \rightarrow L_n$
    {the camera matrix L is composed of the pose and scale estimates}
    $[c_n] = \Omega_I(L_n, c_{n-1})$
    {identity filter estimates the new identity}
  end while

The DUAL EKF algorithm is a simple way to process tracking data. We use the output of the pose filter as inputs to the identity filter and vice-versa. The initial condition for identity is the mean 3DMM face, which is refined as the EKF receives more measurements, this also improves the pose estimates. As a by product we also implemented a Kalman filtered variant of Blanz popular fitting method:

Algorithm 2 EKF+Blanz Algorithm

Require: rows(r) $> 6 + N$
  while r **do**
    $[R_n, s_n, t_n] = \Omega_P(c_{n-1}, R_{n-1}, s_{n-1}, t_{n-1})$
    {pose filter estimates the new incremental pose parameters}
    $R_n = R_n \cdot R_{n-1}$
    {incremental rotation is composed with global rotation}
    $[R_n, s_n, t_n] \rightarrow L_n$
    {the camera matrix L is composed of the pose and scale estimates}
    $[c_n] = B_I(L_n, r_n, \eta)$
    {Blanz method is used to estimates the new identity[3], η is set by the user }
  end while

In both cases the detail behind the EKF persistence is not shown. Readers can refer to Blanz text [3], and to section **5.2** and **5.3** for the EKF measurement equations.

6 Experiments and Results

Using 75 laser scanned heads from the USF database [15], we constructed a 3DMM that retained 95% variance in shape and texture. This 3DMM was then used as the input to our DUAL EKF fitting algorithm to test pose and identity estimates. It was also used to generate testing data for the method.

6.1 Pose

Using our 3DMM it was possible to accurately test pose; the 3DMM provides ground truth data. Thus using a specified set of yaw, pitch and roll functions the method was examined by generating 100 random heads and using the DUAL EKF to track the head motion. The accuracy was measured as the RMS error between the DUAL EKF prediction and the actual measurement. When this error was below **1** pixel then the model was deemed to have accurately converged.

The results (figure **5**) show that the DUAL EKF is capable of tracking the motion quiet successfully. In the case of the 100 random heads, the mean error in yaw pitch and roll was **2, -2** and **-3** degrees respectively. We found empirically that the DUAL EKF pose estimates converged to a solution 95% of the time within as few as **10** frames.

6.2 Structure

To test the structure estimates we used the 3DMM to generate a random head and then attempted to fit the DUAL EKF in two situations, 1) given only a single measurement and 2) as the random head was rotated in a similar fashion to the pose experiment. Since the ground truth coefficients of shape were known we decided to use two metrics to compare our results. The first being the normalized dot-product and the second the L2-norm, this was between the ground truth and the DUAL EKF estimate of shape coefficients. The behavior that was expected was that as the dot-product increased the L2-norm would decrease, we found this to be true.

The first experiment demonstrated that the DUAL EKF could be used to reconstruct the dense 3D shape given only one frame. The fitting was achieved by repeating the single frames measurement essentially pretending the measurement was a new frame. In this case the pose of the model was estimated in only 3 frames

of measurements. This fitting performed well and is shown in figure **4**. Note the expected decrese in error.

The second experiment showed that the EKF could refine an identity estimate across pose changes. Figure **5** demonstrates that the DUAL EKF method is capable of improving its identity estimate across pose changes. Validating our assumption that identity estimates should improve as more frames are provided. Once again the desired convergence behavior is shown, as the dot product increases the L2-norm decreases

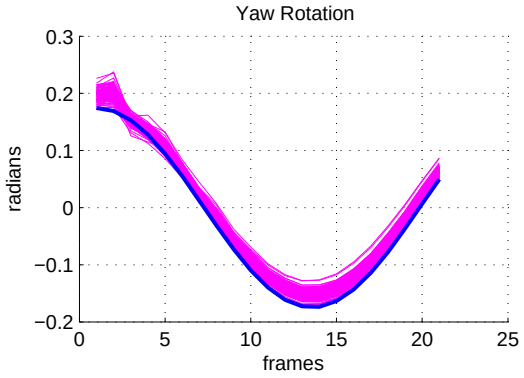
7 Discussion

One significant problem with both Blanz' and our method is the origin of the mapping matrix, V and the measurement, r. The matrix V maps the smaller set of 2D coordinates (r) in an image to corresponding 3D vertices in the 3DMM. One way we believe that this can be solved is to use AAMs trained within the span of the 3DMM 3D head basis to automatically determine V and r. Specifically the AAMs are constructed using the 2D projections of the 3DMM data and inherently provide the important mapping, V. This was identified in [7] and provides an elegant solution to the problem that would otherwise require human intervention.

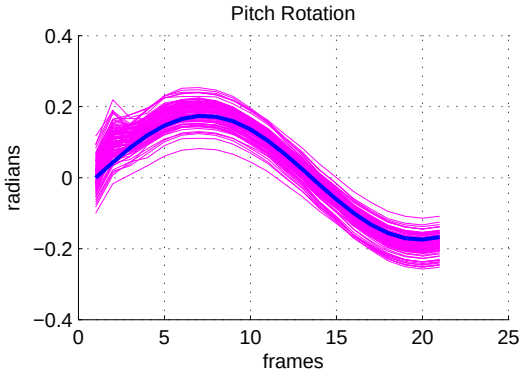
As a by product of the our DUAL EKF work we developed a second algorithm which uses Blanz' fitting method for the identity parameter search. Blanz' method requires an additional integration step (we used a median filter) to improve identity estimates across measurements. The method shows promise if the regularization term could be automatically set. Figure **6** shows preliminary results that we have obtained where we are able to generate many new views from a single image using the second algorithm.

8 Acknowledgments

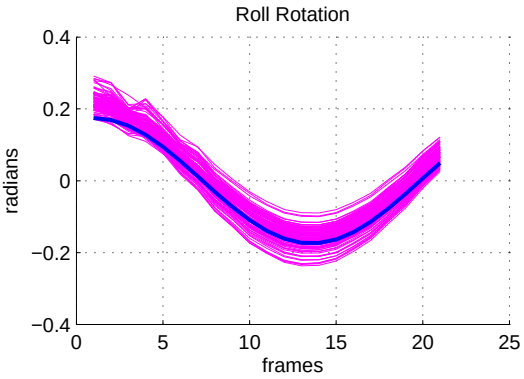
The authors would like to thank Sami Romdhani from the faculty of Computer Science at UniBas for putting the USF data [15] into correspondence and the Australian Research Council for its continued funding.



(a) Yaw motion of 3D head, rotation (radians) / frames



(b) pitch motion of 3D head, rotation (radians) / frames



(c) Roll motion of 3D head, rotation (radians) / frames

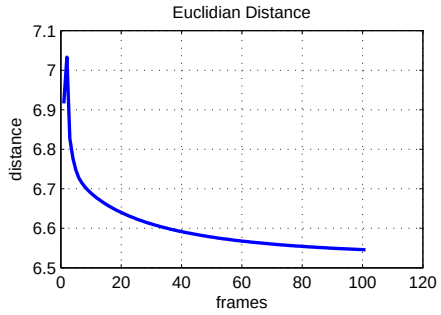
Figure 3. 3D Pose Estimation Plots, Light magenta lines represent the 100 trackings, dark blue lines represent the ground truth motion patterns

References

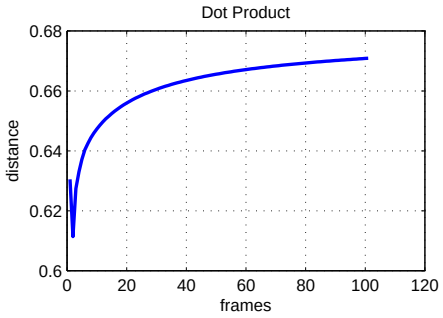
[1] M. Anisetti, V. Bellandi, L. Arnone, and F. Bevernia. Face tracking algorithm robust to pose, illumination and face expression changes: a 3D parametric approach. In *Proceedings of Computer Vision Theory and Applications*, 2006.

[2] A. Azarbayejani and A. P. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6), 1995.

[3] Volker Blanz, Albert Mehl, Thomas Vetter, and Hans-Peter Seidel. A statistical method for ro-



(a) Euclidian distance (between ground truth shape coefficients and DUAL EKF estimate) / frames

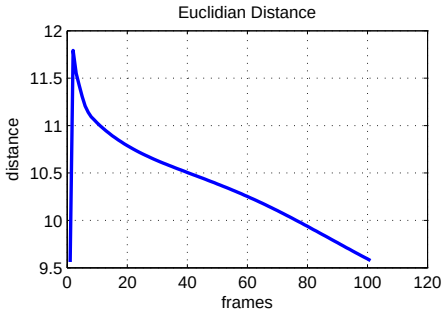


(b) Dot product (between ground truth shape coefficients and DUAL EKF estimate) / frames

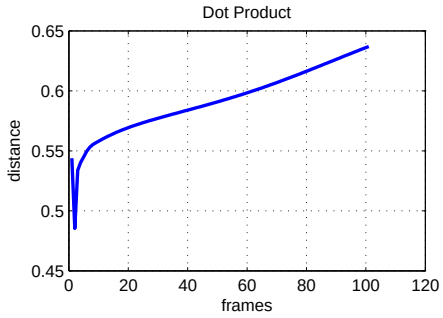
(c) Ground truth 3D Head Render

(d) Converged DUAL EKF Estimate

Figure 4. Plots demonstrating the DUAL EKF's Identity convergence performance given only a single frame of information



(a) Euclidian distance (between ground truth shape coefficients and DUAL EKF estimate) / frames



(b) Dot product (between ground truth shape coefficients and DUAL EKF estimate) / frames

(c) Ground truth 3D Head Render

(d) Converged DUAL EKF Estimate

Figure 5. Plots demonstrating the DUAL EKF's Identity convergence performance across a video sequence (tracking)

(a) original input image

(b) view #1 (c) view #2 (d) view #3

(e) view #4 (f) view #5 (g) view #6

(h) view #7 (i) view #8 (j) view #9

Figure 6. Preliminary results for novel view creation, with a regularization term of $\eta = 1$, 9 (figure 6.a to 6.j) views are generated.

bust 3D surface reconstruction from sparse data. In *Second International Symposium on 3D Data Processing, Visualization and Transmission*, pages 293–300, 2004.

[4] C.Basso, T. Vetter, and V. Blanz. Regularized 3D morphable models. In *Workshop on: Higher-Level Knowledge in 3D Modeling and Motion Analysis*, 2003.

[5] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. In *Proc. European Conference on Computer Vision*, volume 2, pages 484–498. Springer, 1998.

[6] F. Dornaika and J. Ahlberg. Face model adaptation for tracking and active appearance model training. In *British Machine Vision Conference*, 2003.

[7] N. Faggian, S. Romdhani, J. Sherrah, and A. Paplinski. Color active appearance model analysis using a 3D morphable model. In *Digital Image Computing: Techniques and Applications*, December 2005.

[8] Leslie G. Farkas. *Anthropometry of the Head and Face*. Raven Press, New York, 2 edition, 1994.

[9] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[10] T. Lefebvre, H. Bruyninckx, and J. De Schutter. Kalman filters for non-linear systems: a comparison of performance. *Control*, 77(7):639–653, May 2004.

[11] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[12] I. Matthews, J. Xiao, S. Baker, and T. Kanade. Real-time comined 2D+3D active appearance models. In *Computer Vision and Pattern Recognition*, volume 2, pages 535–542, 2004.

[13] P. Mittrapiyanuruk, G. DeSouza, and A. Kak. Accurate 3D tracking of rigid objects with occlusion using active appearance models. In *Proceedings of the IEEE Workshop on Motion and Video Computing*, 2005.

[14] S. Romdhani and T. Vetter. Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[15] Prof. Sudeep Sarkar. USF DARPA humanID 3D face database. University of South Florida, Tampa, FL.

[16] T. Vetter and V. Blanz. A morphable model for the synthesis of 3D faces. In *Siggraph 1999, Computer Graphics Proceedings*, pages 187–194, 1999.