TOWARDS GENERIC FITTING USING MULTIPLE FEATURES DISCRIMINATIVE ACTIVE APPEARANCE MODELS

Pedro Martins, Jorge Batista

Institute of Systems and Robotics, Dep. of Electrical Engineering and Computers
University of Coimbra - Portugal
{pedromartins,batista}@isr.uc.pt

ABSTRACT

A solution for Discriminative Active Appearance Models is proposed. The model consists in a set of descriptors which are covariances of multiple features evaluated over the neighborhood of the landmarks whose locations are governed by a Point Distribution Model (PDM). The covariance matrices are a special set of tensors that lie on a Riemannian manifold, which make it possible to measure the dissimilarity and to update them, imposing the temporal appearance consistency. The discriminative fitting method produce patch response maps found by convolution around the current landmark position. Since the minimum of the responce map isn't always the correct solution due to detection ambiguities, our method finds candidates to solutions based on a mean-shift algorithm, followed by an unsupervised clustering technique used to locate and group the candidates. A mahalanobis based metric is used to select the best solution that is consistent with the PDM. Finally the global PDM optimization step is performed using a weighted least-squares warp update, based on the Lucas Kanade framework. The weights were extracted from a landmark matching score statistics. The effectiveness of the proposed approach was evaluated on unseen data on the challenging Talking Face video sequence, demonstrating the improvement in performance.

Index Terms— Point Distribution Model, Discriminative Active Appearance Models, Riemannian Manifolds.

1. INTRODUCTION

Facial image alignment is the key aspect in many computer vision applications, such as tracking and recognition. In the past years, most existing methods have used generative based methods, where the shape and texture variation were learned from training images. The Active Appearance Models (AAM)[1] is one of the most effective techniques with respect to fitting accuracy and efficiency. Although, it consists on generative holistic representations (in sense that all pixels belonging to the object are used). This representation generalization performs poorly when the target exhibits large amounts of variability, such as the case of the human face under variations of identity, expression, pose, lighting or non-rigid motion due to the huge dimensional representation of the appearance. The main drawback with the generative approaches is that typically they only work well for the individuals held in the training dataset due to the fact that the appearance is eigen based and captured by a linear Principal Components Analysis (PCA). Recently, methods such as the Constrained Local Model (CLM)[2] or [3] [4] have been proposed. These methods use a set of discriminative template regions surrounding individual landmarks whose locations are governed by a Point Distribution Model (PDM). The CLM uses as response surfaces the normalized correlation. In [3] [4] the discriminant descriptor is obtained using machine learning methods, i.e. a linear SVM, which require a extensive training, labeling lots of positive and negative samples. Our approach fits on the discriminative class of methods using shape and appearance models. The shape model is an ordinary PDM that deals with the position of the landmarks. The appearance is composed by a set of descriptors for each of the landmarks in the PDM. The descriptors are covariance matrices of multiple features evaluated on the surrounding location of the landmarks. Since the covariance matrices are a special set of tensors that lie on a Riemannian manifold, it is possible to measure the dissimilarity between two covariances, and also to update them, imposing the temporal appearance consistency. The method starts using a generic covariance (the average covariance observed in the training set) which is then continuously updated. Although, the patch response maps found by convolution around the current landmark position suffers from detection ambiguities. It will be shown that the minimum (in covariance dissimilarity) of the responce map isn't always the desired solution. A solution based on a mean-shift algorithm is proposed, finding candidates to solutions, followed by an unsupervised clustering technique[5] locating and grouping the candidates. A mahalanobis based metric is used to select the best solution. Finally the global optimization step is performed using a weighted least-squares warp update based on the Lucas Kanade framework[6]. The weights were extracted from landmark matching score statistics. This paper is organized as follows: section 2 describes the background required, namely the basics on Riemannian Manifolds and PDM building. Section 3 presents the approach in detail and section 4 and 5 are devoted to experimental results and conclusions, respectively.

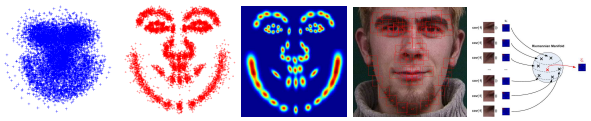
2. BACKGROUND

2.1. Shape Model

The shape of a (2D) Point Distribution Model (PDM) is defined by the vertex locations of a mesh. The representation used for a single v-point shape is a $2v$ vector given by $\mathbf{s} = (x_1, \ldots, x_v, y_1, \ldots, y_v)^T$. The PDM training data consists of a set of annotated images with the shape mesh marked (usually by hand). All the shapes are then aligned to a common mean shape using a Generalized Procrustes Analysis (GPA), removing location, scale and rotation effects. Principal Components Analysis (PCA) are then applied to the aligned

This work was supported by the Portuguese Science Fundation (FCT) through the project "Dinâmica Facial 4D para Reconhecimento de Identidade" with grant PTDC/EIA-CCO/108791/2008. The first autor also acknowledges the FCT for support through the grant SFRH/BD/45178/2008.

shapes, resulting on the linear parametric model $s = s_0 + \Phi\mathbf{p}$, where new shapes, s, are synthesized by deforming the mean shape, $\mathbf{s}_0$, using a weighted linear combination of eigenvectors, $\phi_i, i = 1, \ldots, n$. n is the number of eigenvectors that holds a user defined variance, typically 95%. $\mathbf{p}$ is a vector of shape parameters which represents the weights. See figure 1-a)b)c). Notice that the GPA makes that



(a) (b) (c) (d) (e)

Fig. 1. a) Shape raw data. b) Aligned landmarks after GPA. c) Shape covariance Σ_k around each landmark. d) Patches $\mathbf{P}_k$, $l \times l$ around each landmark. e) Illustration of finding the average covariance $\overline{\mathbf{C}}_k$ for a specific patch (left side of left eye corner). Each training image provide a normalized patch. The covariances for the feature vector $\mathbf{f}$ are evaluated and using eq.2, $\overline{\mathbf{C}}_k$ is found.

the PDM do not model the similarity transformation which is required onto the target image. To overcome this we use the approach proposed by [1], i.e., we include a special set of 4 eigenvectors $\psi_1, \ldots, \phi_4$. A full shape is then described by a linear system $s = s_0 + \sum_{i=1}^{n} p_i\phi_i + \sum_{j=1}^{4} q_j\psi_j$ where $\mathbf{q}$ represents the 2D pose parameters with $q_1 = s\cos(\theta) - 1$, $q_2 = s\sin(\theta)$, $q_3 = t_x$, $q_4 = t_y$ where $s, \theta, (t_x, t_y)$ represents the scale, rotation and translation w.r.t. the base mesh s_0.

2.2. Texture Model - Covariance of Features

The discriminative appearance model used is based on a descriptor of the texture around each one of the v landmarks. Inspired on the work of [7], a quadrangular region $\mathbf{P}$ (patch) with size l is sampled around each landmark. See figure 1-d. On each of the regions, $\mathbf{P}_k, k = 1, \ldots, v$, several features $\mathbf{f}$ are extracted for each pixel $\mathbf{x} = (x, y)^T, \in \mathbf{P}_k$ where $\mathbf{f} = [x \quad y \quad I_x \quad I_y \quad \sqrt{I_x^2 + I_y^2} \quad \arctan\left(\frac{I_y}{I_x}\right) \quad I_{xx} + I_{yy}]$. The features used are the pixel position (x, y), horizontal and vertical gradients (I_x, I_y), gradient magnitude, gradient phase, and the Laplacian. The main advantages of our formulation is that it can always allow more features in order to find a better descriptor for $\mathbf{P}_k$ without changing the remaining formulation. Stacking all measures of $\mathbf{f}$, i.e. $\mathbf{F}_k = \mathbf{f} \in \mathbf{P}_k$, the $d \times d$ covariance matrix for the features is given by $\mathbf{C}_k = \frac{1}{l^2-1}\sum_{i=1}^{l^2}(\mathbf{F}_{k_i} - \mu_{\mathbf{P}_k})(\mathbf{F}_{k_i} - \mu_{\mathbf{P}_k})^T$ where $\mu_{\mathbf{P}_k}$ is the vector of feature means within the region $\mathbf{P}_k$. The covariance $\mathbf{C}_k$ is used as region descriptor (which represents the correlations between the features $\mathbf{f}$ for the entire region $\mathbf{P}_k$). The main advantage of using covariances of features is that, if they are positive definite matrices, $\mathbf{C}_k$ lie in a Riemannian Manifold and is possible to measure dissimilarities and make updates.

2.2.1. *Dissimilarity Between Covariances*

The covariance matrices do not lie on Euclidean space. Based on the Riemannian invariants, a distance metric[8] is used. The dissimilarity between two covariances matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ is given by

$$\rho(\mathbf{C}_1, \mathbf{C}_2) = \sqrt{\sum_{i=1}^{m} ln^2\lambda_i(\mathbf{C}_1, \mathbf{C}_2)} \qquad (1)$$

where $\lambda_i(\mathbf{C}_1, \mathbf{C}_2)_{i=1,\ldots,m}$ are the generalized eigenvalues of $\mathbf{C}_1$ and $\mathbf{C}_2$, computed from $\lambda_i\mathbf{C}_1\mathbf{x}_i - \mathbf{C}_2\mathbf{x}_i = 0$, $i = 1, ..., d$ and $\mathbf{x}_i \neq 0$ are the generalized eigenvectors. Note that $\rho(\mathbf{C}_1, \mathbf{C}_2) \geq 0$.

2.2.2. *Updating Covariances - Finding $\overline{\mathbf{C}}_k$*

The mean of the points on the manifold minimizes the L_2 norm of $\overline{\mathbf{C}} = \arg\min \sum_{t=1}^{T} \rho^2(\mathbf{C}, \mathbf{C}_t)$. [8] proposed a gradient descent approach to compute the $\overline{\mathbf{C}}$ by $\overline{\mathbf{C}}^{i+1} = \exp_{\overline{\mathbf{C}}^i}\left(\frac{1}{T}\sum_{t=1}^{T}\log_{\overline{\mathbf{C}}^i}(\mathbf{C}_t)\right)$. To prevent the model from contamination, it is possible to weight the data points by a factor proportional to its similarity to the current model, resulting

$$\overline{\mathbf{C}}^{i+1} = \exp_{\overline{\mathbf{C}}^i}\left(\frac{1}{\rho^*}\sum_{t=1}^{T}\rho^{-1}(\mathbf{C}_t, \overline{\mathbf{C}}^*)\log_{\overline{\mathbf{C}}^i}(\mathbf{C}_t)\right) \qquad (2)$$

where ρ is defined in eq.1, $\rho^* = \sum_{t=1}^{T}\rho^{-1}(\mathbf{C}_t, \overline{\mathbf{C}}^*)$ and $\overline{\mathbf{C}}^*$ is the model computed at the T previous frames. Each training image provide a set of v covariances matrices, $\mathbf{C}_k$ (for each landmark k). For N images in the set, the average covariance matrix, $\overline{\mathbf{C}}_k$, is computed over the Riemannian Manifold using eq.2. See figure 1-e for a graphical interpretation of this process. The mean covariance, $\overline{\mathbf{C}}_k$, is used as the initial descriptor for that specific landmark k.

2.3. Image Normalization - Affine Warp

Since the covariance isn't invariant to scale and rotation effects, a normalization at image level is required. The normalization is based on an affine warp of the entire image in a way that the current mesh s is mapped into the reference base mesh s_0.

3. OUR APPROACH - DAAM-R

After building the PDM and evaluating the average covariance $\overline{\mathbf{C}}_k$ for each landmark k (in a training stage), fitting the Discriminative AAM embedded on a Riemannian Manifold (DAAM-R) consists on finding k local optimal displacements, $\Delta\mathbf{x}^\dagger$, from the PDM current mesh position s. The local updates, expressed in the base mesh, will be constrained to lie in the subspace spanned by Φ by an nonlinear optimization based on the Lucas Kanade framework[6]. (See section 3.1). The goal is to find the deviation from the PDM, $\Delta\mathbf{x}$, for each landmark. The sequencial steps of proposed approach are enumerated and figure 2 shows the overall view for the fitting methodology.

(1) Scanning by convolution around a local region finding a response map of covariances dissimilarities (figure 2-a).

(2) The minimum of the responce map, i.e. the lower dissimilarity, doesn't always correspond to the correct landmark location. Actually, in some cases it can be a poor estimate, since the features consists of small image patches that often contain limited structure, leading to detection ambiguities. Since the global minimum couldn't be always the correct solution it is however assumed that the correct solution is a local minima. A modified version of a mean-shift algorithm (section 3.2) is used to detect all the local minima (see figure 2-b) providing a set of candidates to the landmark solution.

(3) The mean-shift will produce clusters with the candidates regions to solutions, $\Delta\mathbf{x}_k^*$. At this stage is important to define the number and location of these clusters. For this propose an unsupervised clustering method proposed by [5] it used. See figure 2-c and section 3.3.

(4) Knowing the clusters and their locations, it is required to select the best cluster, $\Delta\mathbf{x}_k^\dagger$, (section 3.4). The selection is based on the cluster that will be more consistent with the PDM (figure 2-d).

(5) Finally, establish the landmark matching score assigning weights to the found solution (section 3.5) and performing global PDM optimization (section 3.1).

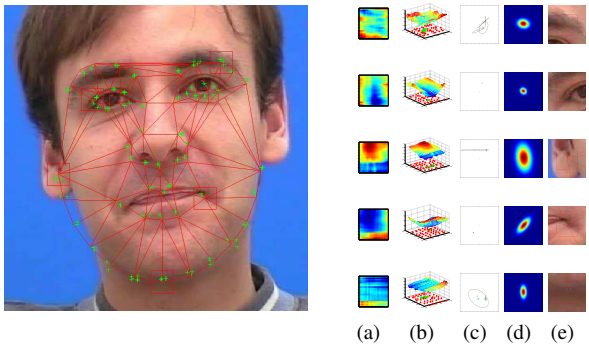


Fig. 2. Overview of the DAAM-R. The left main figure represents the first iteration of the method. Starting with an initial estimate of the position of the face (by AdaBoost[Viola-Jones]). a) Response maps of covariance of features dissimilarity around each $\mathbf{x}_k$ (blue small dissimilarity). b) 3D mesh for the response maps. At the ground level with red color is represented the mean-shift seeds starting grid. The green circles are the seeds final position (local minima). c) Unsupervised clustering to find the clusters and their locations. The red cross at the center represents the current landmark position $\mathbf{x}_k$. The small green circles are the mean-shift seeds at a near local minima location and the ellipses are the clusters found. d) Representation for Σ_k. e) Detailed matching solutions. The green dots are the centroid locations, $\mathbf{x}^*_{k_i}$, and the selected solution $\mathbf{x}^\dagger_k$ is the one pointed by the green arrow.

3.1. Global Optimization - Fitting the PDM

The PDM fitting is accomplish using the Lucas Kanade framework[6]. The warp function is given by $\mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q}) = s_0 + \Phi\mathbf{p} + \Psi\mathbf{q}$, where $\mathbf{p}$ is the shape parameters and $\mathbf{q}$ the similarity parameters. The Jacobian of the warp is given by $\frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{p}} = \Phi^T$ and $\frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{q}} = \Psi^T$. The non-rigid alignment can be posed into the following optimization problem

$$\arg\min_{\mathbf{p},\mathbf{q}} \sum_{k=1}^{v} \rho(\mathbf{C}_k\{s_0 + \Phi\mathbf{p} + \Psi\mathbf{q}\}), \overline{\mathbf{C}}^*_k) \tag{3}$$

minimizing the covariance dissimilarity $\rho(.)$ between the *model* covariance, $\overline{\mathbf{C}}^*_k$, and the covariance computed on a shifted location, but constrained to be consisted with the PDM, $\mathbf{C}_k\{s_0 + \Phi\mathbf{p} + \Psi\mathbf{q}\}$, for all the v patches in the model. The *model* covariance, $\overline{\mathbf{C}}^*_k$, starts by being the average $\overline{\mathbf{C}}_k$ on the Manifold. It is computed from the training images and is weighted updated every frame enforcing the temporal appearance consistency using the approach described in section 2.2.2. A T sized buffer is used to evaluate $\overline{\mathbf{C}}^*_k$. This update process is only done after the PDM fitting of the target frame. For solving the cost function (eq.3), a weighted least-squares optimization is used. It requires finding v local translations by exhaustively search the region around each patch such that $\Delta\mathbf{x}^\dagger_k = \arg\min_{\Delta\mathbf{x}_k} \rho(\mathbf{C}_k\{\mathbf{x}_k + \Delta\mathbf{x}_k\}), \overline{\mathbf{C}}^*_k)$ where $\Delta\mathbf{x}^\dagger_k$ is the optimal, in some sense, local displacement for the patch k. The evaluation of $\Delta\mathbf{x}^\dagger_k$ is described on the following subsections. The weighted least-squares warp update is given by

$$\Delta\mathbf{p} = \left(\frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{p}} W \frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{p}}^T\right)^{-1} \frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{p}} W \Delta\mathbf{x}^\dagger \tag{4}$$

where W is a $2v \times 2v$ diagonal matrix of weights, $W = diag(w_1, ..., w_v, w_1, ..., w_v)$. Each w_k weight measures the fitting importance for landmark k. See section 3.5 for details on how to estimate the weights. The parameters update equation for $\Delta\mathbf{q}$ is similar to eq.4 but insted of using $\frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{p}}$ it uses $\frac{\partial \mathbf{W}(\mathbf{x},\mathbf{p},\mathbf{q})}{\partial \mathbf{q}}$. For this particular case, the Jacobian of the warp is constant and the forward additive update method[6] can be used. Solving the PDM consists on iteratively use eq.4 and update the parameters by $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ and $\mathbf{q} \leftarrow \mathbf{q} + \Delta\mathbf{q}$ until $\|\Delta\mathbf{p}\| \leq \epsilon$, or a maximum number of iterations is reached. Note that the image normalization process, described earlier in section 2.3, is performed at each iteration.

3.2. Weighted Mean-Shift - Find Candidates

Mean-shift algorithm is a robust clustering technique which does not require prior knowledge on the number of clusters[9]. The *weighted* mean shift vector at point $\mathbf{x}$ is defined as $\mathbf{m}_h(\mathbf{x}) = \frac{\sum_{i=1}^m \mathbf{w}_i \mathbf{x}_i g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^m \mathbf{w}_i g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)}$ where $g(s) = -k'(s)$, with $k(s)$ being the kernel profile (it is used $k(s) = e^{-\frac{s}{2}}$) and the h bandwidth. $\mathbf{w}_i$ is the *normalized* weight assigned to each data point $\mathbf{x}_i$. The algorithm starts at the data points and at each iteration t moves in the direction of the mean shift vector $\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{m}_h(\mathbf{x}_t)$. The mean-shift vector always points toward the direction of the maximum increase in the density. If weights are assigned as $\mathbf{w}_{i,i=1,...,m} = \rho^{-1}(\mathbf{C}_k\{\mathbf{x}_k + \Delta\mathbf{x}_k\}, \overline{\mathbf{C}}^*_k)$, where m is the number of seeds (i.e. equal to the inverse of the dissimilarity of the response maps), the seeds will move into the local minima. Note that the weights, $\mathbf{w}_i$, should be normalized such that $\sum_{i=1}^m \mathbf{w}_i = 1$. The selection of the number and the starting position of the seeds is very important. A sparse grid of 3×3 blocks is used, where a single seed is assigned to the position inside the block that has the higher weight. The number of seeds used, m, will be the number of 3×3 blocks inside the grid of the scanned areas.

3.3. Finding Clusters

The mean-shift will provide candidates to solutions. Searching for those candidates is a clustering problem. For this propose the unsupervised clustering method proposed by [5] was used. Usually the mean-shift seeds converge around a few locations, forming clusters where the seeds are not positioned at the exact same place. The clustering also filters this effect by taking the centroid position as the candidate solution.

3.4. Selecting the Best Cluster

Knowing the clusters and their centroid locations it is required to select the best cluster. The selection is based on the cluster that is more consistent with the PDM. Recalling figure 1-c, the individual shape localization covariance, Σ_k, was estimated. The selected cluster is the one that has the lower mahalanobis distance w.r.t. the correspondent PDM landmark. Formally, the centroid locations $\mathbf{x}^*_{k_i}$ are given by $\mathbf{x}^*_{k_i} = \mathbf{x}_k + \Delta\mathbf{x}^*_{k_i}$ with $i = 1, \ldots, c$, and c the number of centroids found. The deviation update found for each cluster is $\Delta\mathbf{x}^*_{k_i}$. The selected candidate for the solution, $\mathbf{x}^\dagger_k$, is the one that has the

lower mahalanobis distance, i.e. is more close to the PDM distribution $\mathbf{x}_k^\dagger = \arg\min d_m(\mathbf{x}_{k_i}^*, \Sigma_k)$, where $d_m(.)$ is the mahalanobis distance that is evaluated for all $\mathbf{x}_{k_i}^*$, and Σ_k is the shape position covariance of the landmark k.

3.5. Landmark Matching Score

The PDM optimization is based on a weighted least-square warp update, dealing with some possible landmarks mismatches. From eq.4 this information is included as a diagonal matrix of weights and those weights are based on landmark confidences. The statistics for the landmarks covariance of features matching score can be learnt previously from the training images. The residual error on matching, $\overline{\mathbf{C}}_k$, follows a half normal which is approximated by a normal distribution with zero mean and a given standard deviation $\sim \mathcal{N}(0, \sigma_{\overline{\mathbf{C}}_k})$. The error standard deviation $\sigma_{\overline{\mathbf{C}}_k}$ can be estimated from the training set as $\sigma_{\overline{\mathbf{C}}_k} = \sqrt{\frac{\sum_{i=1}^{N} \rho(\mathbf{C}_i, \overline{\mathbf{C}}_k)^2}{N-1}}$, where N is the total number of images. Knowing $\sigma_{\overline{\mathbf{C}}_k}$ and defining $\mathbf{C}_k^\dagger$ to be the covariance of features evaluated at the solution $\mathbf{x}_k^\dagger$, the weights for the matrix W can be assigned as $w_k = \exp\left(-\frac{\rho(\mathbf{C}_k^\dagger, \overline{\mathbf{C}}_k^*)^2}{2\sigma_{\overline{\mathbf{C}}_k}}\right)$.

4. EXPERIMENTAL RESULTS

The experimental results were conducted using two free available independent datasets. The IMM dataset[1] annotated with $v = 58$ landmarks, see figure 1-a and the FGNet Talking Face sequence (TF)[2]. The main experience consists on training the DAAM-R with about 160 near frontal images from the IMM set and test the ability of fitting in unseen images, comparing it with other fitting algorithms trained with the same input data. The DAAM-R training consists in building the PDM, compute the average covariance of features for each landmark, $\overline{\mathbf{C}}_k$, and find the matching statistics $\sigma_{\overline{\mathbf{C}}_k}$ using only the images from the IMM set. The fitting accuracy is evaluated using the initial 1000 frames of the TF sequence. Our method (DAAM-R) is compared with the standard AAM algorithms: the Project Out (PO)[1], the Simultaneous Inverse Compositional (SIC) and the SIC Efficient Approximation (SIC-EA)[6]. The method is also tested against the robust extensions: Roboust Normalization Inverse Compositional (RNIC)[6] and the Roboust SIC (RSIC)[6]. Regarding the remaining details about the DAAM-R, the sampled patches $\mathbf{P}_k$ have the size of 11×11, $(l = 11)$, (intraocular distance of about 80 pixels). Durring the fitting process the search area is a window of 21×21 pixels around each landmark. For the mean-shift, the system works well with a bandwidth of $h = 8$, $\epsilon = 0.01$, and maximum number of iterations 50. The number of seeds used were $m = 49$. The unsupervised clustering requires the minimal and maximal number of clusters to find, 1 and 5 respectively. The *model* covariance buffer is $T = 30$, which means that that for every landmark the $\overline{\mathbf{C}}_k^*$ is computed from 30 previous weighted samples. The termination criteria in DAAM-R was set to $\epsilon = 0.75$ and the maximum iterations was set to 10. For the other algorithms $\epsilon = 0.75$ and maximum iterations of 20. The robust algorithms also require the choose of a error norm, the Talwar function is used (gives a weight of 1 to inliers and 0 to outliers), where the scale parameter is estimated from the error image assuming that there exists 15% of outliers. The figure 3 shows the RMS fitting error in the TF sequence for all the evaluated methods. Since the IMM uses a 58 landmark scheme and the TF uses 68, the error was only measured over the correspondent landmarks. The colored circles over the graphic represent reinitializations of the models. Note that our approach, DAAM-R, never make a restart. The results show that the method is generally very accurate.

[1] www2.imm.dtu.dk/ aam

[2] www-prima.inrialpes.fr/FGnet/data/01-TalkingFace/talking_face.html

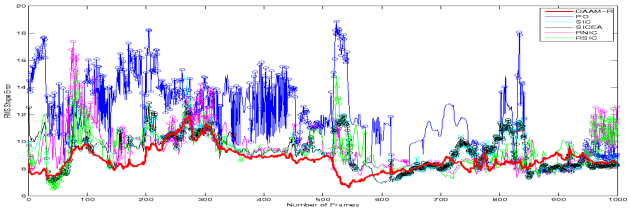


Fig. 3. RMS error in fitting the first 1000 frames of the TF sequence.

5. CONCLUSIONS

The DAAM-R uses independent shape and texture models. The texture is composed by a set descriptors for the landmarks. These descriptors are covariances of multiple features evaluated around the landmark locations which are governed by a PDM. The covariance matrices lie on a Riemannian manifold, which make possible to measure the dissimilarity and to update them, imposing the temporal appearance consistency. Using a discriminative fitting approach, response maps are found. Since the minimum of the responce map isn't always the correct solution a strategy based on mean-shift is used to find candidates to solutions (local minima). An unsupervised clustering technique is used to locate and group the candidates and a mahalanobis based metric is used to select the best solution consistent with the PDM. The global optimization for the PDM is performed using a weighted least-squares warp update, where weights were extracted from the landmark matching confidences statistics. The DAAM-R trained with mostly frontal images taken from the IMM dataset is evaluated by fitting to unseen data on the challenging Talking Face video sequence (1000 frames). The model performs well without lose track during all the sequence.

6. REFERENCES

[1] I.Matthews and S.Baker, "Active appearance models revisited," *International Journal of Computer Vision*, November 2004.

[2] D.Cristinacce and T.F.Cootes, "Automatic feature localisation with constrained local models," *Pattern Recognition*, , no. 10, 2008.

[3] Y.Wang, S.Lucey, and J.Cohn, "Enforcing convexity for improved alignment with constrained local models," in *IEEE CVPR*, June 2008.

[4] Y.Wang, S.Lucey, J.Cohn, and J.M.Saragih, "Non-rigid face tracking with local appearance consistency constraint," in *IEEE International Conference on Automatic Face and Gesture Recognition*, October 2008.

[5] M.Figueiredo and A.Jain, "Unsupervised learning of finite mixture models," *IEEE Transactions on PAMI*, March 2002.

[6] S.Baker, R.Gross, and I.Matthews, "Lucas kanade 20 years on: A unifying framework: Part 3," Tech. Rep. CMU-RI-TR-03-35, CMU Robotics Institute, November 2003.

[7] F.Porikli, O.Tuzel, and P.Meer, "Covariance tracking using model update based on lie algebra," in *IEEE CVPR*, June 2006.

[8] X.Pennec, P.Fillard, and N.Ayache, "A riemannian framework for tensor computing," *International Journal of Computer Vision*, 2006.

[9] D.Comaniciu and P.Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on PAMI*, May 2002.