SIMPLEX OPTIMISATION INITIALIZED BY GAUSSIAN MIXTURE FOR ACTIVE APPEARANCE MODELS

Yasser Aidarous
yasser.aidarous@supelec.fr

Sylvain Le Gallou
sylvain.legallou@supelec.fr

Renaud Seguier
renaud.seguier@supelec.fr
SUPELEC IETR
Signal Communication Electronique Embarque Laboratory
Supelec Rennes, Cesson Sevigne, France

Abstract

Active appearance model efficiently aligns objects which are previously modelized in images. We use it for Human Machine Interface (face gesture analysis, lips reading) to modelize mouth on embedded systems (mobiles phones, game console). However those models are not only high memory and time consumer but also not robust in the case of object with high deformations (different pose of a face or different expressions of mouth): this is the manifold problem [3]. We propose a new optimization method based on Nelder Mead Simplex [12] initialized by Gaussian Mixture (GM). The GM is applied to the learning data in the reduced space. This method reduces memory requirement and improves the efficiency of AAM when we modelize high deformable object at the same time. The test, carried out on France Telecom and BioID data bases, shows that our proposition to align mouth outperformed the classical optimization when applied to mouth alignment and give the same results as classical optimization on common face alignment.

1. Introduction

In Human Machine Interface (HMI) it is necessary to recognize objects (faces, hands, mouths) and analyse them to identify motions and gestures. All of these applications first need to align objects to be analysed. When it is implemented on an embedded system this alignment operation must not be time and memory consuming. We use Active Appearance Model to align faces and mouths in embedded HMI (Mobile phones, and game console). AAM was proposed by Edward, Cootes and Taylor [8] in 1998. They allow synthesizing an object with its shape and texture. Appearance variation is collected in a consistent manner, by establishing a warp function and a Regression Matrix (RM) between the variation of model parameters and modelization error. RM is used to adjust the model to an object. This optimization has 2 problems:

Required memory: memory space required to save RM and the mean model makes it difficult to implement on embedded systems.

Manifolds: object shape and texture can vary with non linear variation. For example mouth presenting different expressions [3] or cars in different pose. Learning examples then define distinct regions (fig.1). During learning phase, RM construction is done for each image based on the real c vector. The RM initialization is the mean model $(c(1), c(2) = \vec{0})$ can't reach any space expression with a linear relation (fig.1). This is why the RM can not be used in manifolds problem.

We propose a new optimization method based on Nelder Mead Simplex initialized by Gaussian Mixture (GM). The gaussian mixture is optimized by Expectation Maximization algorithm. After a brief AAM presentation and the method proposed by community to improve its memory consumption and avoid manifold problem (section 2), in section 3 we'll present our optimization method, then test results are illustrated in section 4. In section 5 we'll conclude and present our future works.

2. Background

2.1. Active Appearance Model

AAM uses PCA to encode both shape and texture variation of training data base. The shape of an object can be

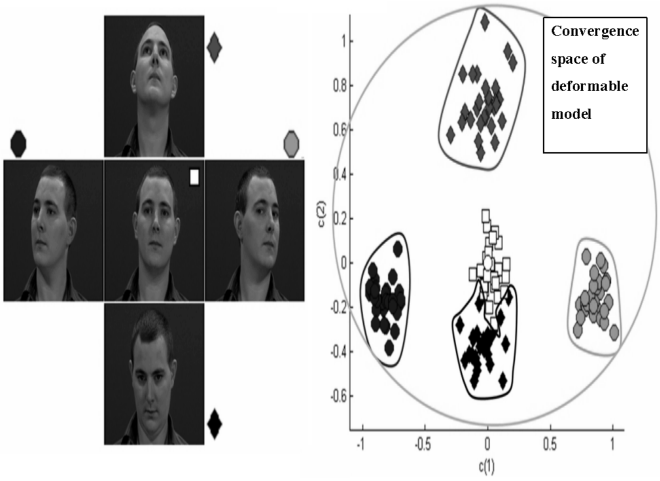

Figure 1. Representation of different face orientation. The space formed by first and second reduced space variable shows the distinct regions.

represented by vector s and the texture (gray level) by vector g. We apply one PCA to the shape and another PCA to the texture in order create the model, given by:

$$\begin{aligned} \mathrm{s_i} &= \bar{s} + \Phi_s * b_s \\ g_\mathrm{i} &= \bar{g} + \Phi_g * b_g \end{aligned} \tag{1}$$

Where s_i and g_i are shape and texture, $\bar{s}$ and $\bar{g}$ are mean shape and mean texture. Φ_s and Φ_g are vectors representing variations of orthogonal modes of shape and texture respectively. b_s and b_g are vectors representing parameters of shape and texture. By applying a third PCA to the vector $b \begin{bmatrix} b_s \\ b_g \end{bmatrix}$ we obtain:

$$b = \Phi * c \tag{2}$$

ϕ is matrix of d_c eigenvectors obtained by PCA. c is appearance parameters vector. The modifications of c parameters change both shape and texture of the object. Each object is defined by the appearance vector c and pose vector t:

$$t = \begin{bmatrix} t_x & t_y & \theta & S \end{bmatrix}^T \tag{3}$$

Where t_x and t_y are x and y axis translation, θ is angle of orientation and S is Scale.
c is appearance parameters vector. AAM learn the linear regression models which give us the predicted modifications of model parameters δc and δt:

$$\begin{aligned} \delta c &= R_c G \\ \delta t &= R_t G \end{aligned} \tag{4}$$

R_c and R_t are the appearance and pose regression matrices respectively. The model search is driven by the residual image G: the difference between the search image and model reconstruction.

2.2. Related works

Many methods were developed to overcome the weakness presented by AAM method. *Memory space problem* was dealt with replacing Principal Component Analysis PCA of AAM for generating texture model by Wavelet [11] to reduce the size of texture RM. In [13] Simulated Annealing was used but proved to be time consuming. *Direct Appearance Model* [10] is derived from the classical AAM by eliminating the joint PCA on texture (Eq.2) and shape. It uses the texture information directly for the prediction of the shape: the estimation of position and appearance. It comes from the fact that we could extract the shape directly from texture. The main difference between the DAM and the AAM is in the third PCA. The dimension of the new space representation is four times less than the dimension given by PCA in classical AAM and the prediction is more stable. The regression in the DAM then requires less memory than the regression used in the AAM. In [10] it is shown that the size of the matrix of regression is $11, 83$ lower than that of AAM.
The method of *Active Wavelet Networks* [11] uses the wavelets as alternative to the PCA in order to reduce the dimension of space. It uses a Gabor Wavelet network [11] to model the variations of the texture of the training base. The given weights are of the shape to preserve the maximum information contained in image for a fixed wavelet number. The DAM and AWN methods make it possible to reduce the required memory to store RM.

Manifolds problem may be treated with the DAM and AWN by executing several AAM (each AAM representing one expressions) but the memory and time consumption must be multiplied by the expressions number. The gaussian mixture [15] is used to make the difference between the different expression classes of the same object (manifolds) modelized by AAM [3]. The mixture of the algorithm is applied on the real learning data images. Each expression class is represented by a gaussian and defines a model with a specific RM. During the search phase, a number of AAM equals the number of expressions applied. The retained solution generates the minimal error between the generated model and the input image. The problem of manifolds was dealt in many approaches like the extension number of models in supervised [5] [6] and unsupervised way [2], and specification of the learning data base in hierarchical approach [17] [16] and identity specification approach [9] [1].

The *Nelder and Mead simplex* was used in face feature detection [7]. Model of each landmark (17 landmarks for face) is created and the simplex optimize the placement of landmarks by using score function given by each model. The features models are in low dimension compared by model of whole face. [7] doesn't optimize AAM parameters but placement of landmarks. The simplex does not use prior knowledge, which makes them efficient in generalization and they don't need too much memory space as well.

In the following section, we propose a method to remove the space allocated to store these matrices and to reduce time execution.

3. AAM- GM simplex optimization

The simplex does not use prior knowledge, which makes them efficient in generalization and they don't need too much memory space as well. We represent in fig.1 the learning data of different poses face in the plan of the two first parameters given by the PCA (2). The face poses which present a manifold problem remain separated even represented in reduced space. We'll exploit this to initialize the simplex with solutions representing different classes. We propose then to initialize the simplex using a gaussian mixture. We pick randomly the initial solutions in each gaussian of the mixture. The use of the GM will accelerate and improve the simplex algorithm.

3.1 Nelder Mead simplex

Simplex of Nelder Mead [12] makes possible to find the minimum of function of several variables in an iterative way. We initialize the algorithm with $n + 1$ solution, where n is the number of parameters to be optimized. Thus the solution where the function is highest $(= E_{max})$ is rejected to be replaced by another solution which will be calculated according to the precedents. The efficiency of Simplex depend on the manner in which it was initialized. The operators of search for solutions minimizing the objective function are as follows:
Reflection: we test the point which is in the opposite direction of the bad solution.
Expansion: we prolong research beyond the point of reflection by testing the solution.
Contraction: if the previous two operators of search fail then we minimize tests points close to share and other of the current solution.
Shrinkage: if all the previous solutions do not minimize E we narrow down the triangle by changing these tops, and tests the preceding disturbances.

3.2 GM Simplex optimization

We propose to optimize AAM using Simplex initialized by GM. The function to be optimized is the error of pixels:

$$E = \sum_{i=1}^{M} (g_i^{\mathrm{mod}} - g_i^{image})^2 \tag{5}$$

Where M is the number of model pixels, g_i^{mod} is the intensity of the pixel i in model generated by new solution and g_i^{image} is the intensity of the pixel i in the image containing the searched object. The use of an GM initialization makes us able to get initial solutions closer to the research optimum. The algorithm is as follows:
Initialization: After creating the model, we get the vectors c (Eq. 2) representing each image of the learning data base, we look for gaussian mixture over the appearance vectors representing the learning data in the reduced space given by the PCA using Expectation Maximization (EM) Algorithm. This is done off line. The weights, the mean and the gaussian variances making the mixture enable us to choose randomly a number of vectors (proportional to the gaussian weights) belonging to each gaussian. These vectors will initialize the SP.

Convergence: A model is described by:

$$v = \begin{bmatrix} c \\ t \end{bmatrix} \tag{6}$$

Size of v is $d_c + d_t$ (d_c and d_t are the c (Eq. 2) and t size (Eq. 3), the simplex must optimize $d_c + d_t$ parameters to align the model on the face. The simplex starts search from $d_c + d_t + 1$ solutions chosen randomly in space under constraint. So as to use the simplex algorithm in AAM optimization, variables constraint and stopping criterion must be defined.

Constraints: They are applied to avoid testing incorrect solutions that we know preliminary. These constraints bound search space on appearance and pose variables. We initialize the vector v (Eq. 6), representing appearance and pose, randomly in interval corresponding to each parameter under Cootes's constraint:

- Appearance constraint: appearance variable c interval is $-2\sqrt{\lambda}$ and $2\sqrt{\lambda}$ [14]. Where λ is the eigenvalues of the eigenvectors corresponding to matrix $G^T G$, G is the gray level difference matrix between modifying model and real image. These constraints are verified during algorithm execution.
- Pose constraints: AAM is robust untill 10% in scale and translation [4]. We initialise the pose variable (x and y axis translation, rotation and scale) in the interval of 10% of the object real pose vector.

Stopping criterion: the algorithm will end after fixed number of iterations (to insure maximum processing time) or when it will converge in population. The population convergence is obtained if difference between the error values (Eq. 5) of the proposed solutions do not pass the threshold S_E. In the case of error normalization, done by Stegmann in [14], the mean value of the error is stable E_{mean} on different images of the same object, at the time when the alignment is correct. We propose to settle $S_E = 0.1 \times E_{mean}$.

Memory consumption: Simplex neither need additional required memory space necessary to store the model nor information on the directions to a priori minimize the error E. The size of the two matrices (of appearance R_c and of pose R_t) is equal to $M \times (d_c + d_t)$ Bytes, M being the number of pixels contained in texture of the model. For our test with images $64 \times 64 pixels$, we have $d_c = 8$ and $d_t = 4$, then the size of the regression matrices (Eq. 4) is $98KB$. The mean model size is $8KB$. The memory require for the AAM is $\approx 106KB$. In the case of the simplex we don't need RM, therefore, the memory used is $8KB$. Memory space required by the simplex is lower than that used in [11] which memorizes the wavelet coefficients in addition to the model and also lower than that needed in [10] which stores the matrix R_c.

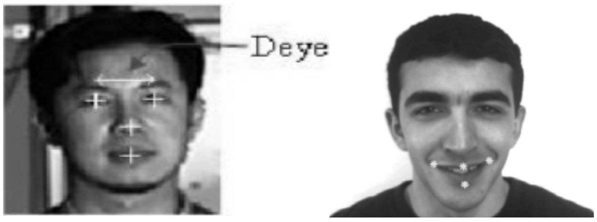

Figure 2. D_{eye} distance and the 4 points used to calculate the error marking.

4 Experiences

4.1 Error marking

To evaluate the performance of our proposed optimization, we do two test:

- Mouth alignment under manifold problem: the test was realized on 116 mouth images from france Telecom data base (FT) which do not belong to the training data base. The training data base is made up of 20 images belonging to the FT data base.
- Face alignment in generalization: the test was realized on 1521 face images belonging to different persons from BioID (Biometric Identity Data Base) data base. The training data base is made up of 15 images belonging to the M2VTS (Multi Modal Verification for Teleservices and Security applications) data base.

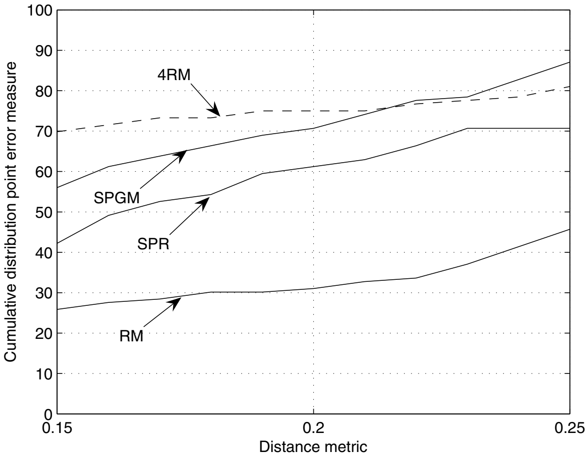

Figure 3. Results obtained on 116 images from FT data base

To qualify the convergence of the AAM we will define an error marking. This error $f_i (i = 1, 2, 3, 4)$ is calculated for each part of the face $i = left\ and\ right\ eye, nose, lips$ such as:

$$\begin{array}{l} f_i = (p_{gi}^{find} - p_{gi}^{real})/D_{eye} \\ e = max(f_i) \\ with \quad p_{gi}^{find} = \frac{1}{Q_i} \sum_{r=1}^{Q_i} p_{ir}^{find} \end{array} \quad (7)$$

Where e is the marking error, p_{ir}^{real} are the coordinates of the ground truth of the marking points of the face part i, p_{ir}^{find} the coordinates of the marking points of the face part i found by AAM and Q_i is the number of landmarks of the i face part. The algorithm converges when the 4 errors are lower than convergence threshold. The threshold was calculated according to the distance between the eyes D_{eye}. In the case of mouth alignment we take into acount only the mouth part.

4.2 Results

We test the efficiency of our method comparing to the RM and SP randomly initialized (SPR) in the PCA space. In order to make the three methods comparable, we fix the iterations's number (error calculations and warping) to the RM necessary number of iterations to converge. We fix this number to 837 error calculation. The fig.4 shows overlay

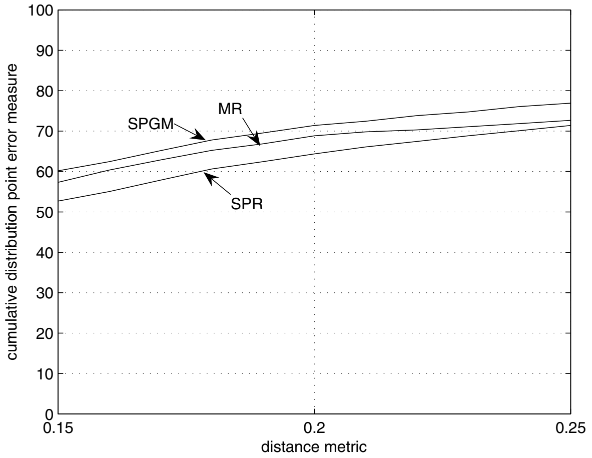

Figure 4. Results obtained on 1521 images from BioID data base

of cumulative distribution of maximum point to point error measure curves of the three methods. We are interested in the interval $\begin{bmatrix} 0.15 & 0.25 \end{bmatrix}$, under 0.15 the error can be generated by the manual annotation and up of 0.25 we consider that the algorithm diverges. The fig.3 shows robustness of SPGM comparing to the RM and SPR in mouth modelization. By fixing the threshold to 0.2, which presents a threshold of a good precision, the SPGM method present convergence rate of 71% whereas the convergence rate is 32% for the RM and 63 for the SPR. it shows that the SPGM optimisation is very efficient comparing to SPR and RM. It allows us to calculate the number of image where the algorithm do not converge. In 116 test images, we notice that the RM diverge in 54%, SPR have a divergence rate of 28% and SPGM diverge in only 12% of the testing data base. The test done on the BioID data base (fig.4) shows that the RM and SPGM are efficient comparing to SPR but the three methods are in the same range in terms of convergence rate. This is due to the compactness of data distribution in the case of faces. In this case data may be represented with overlapping gaussians. Even we have not a manifold problem in the BioID data base the SPGM remains attractive. Fig.4 shows a comparison of the SPGM method and the method used in [3], which we call 4RM, in the case of mouth alignment. We notice that the 4RM method is more efficient than the SPGM method if a very good precision in the alignment is required (between 15% and 22%); it's due to the use of different model (in our case 4 model), one model for each expression. In the interval $\begin{bmatrix} 0.22 & 0.25 \end{bmatrix}$, both methods are equivalent. Neverthless, it must be noticed that 4RM presents a memory consumption and time executing four times higher than RM and SPGM and then cannot be implemented in a real time application. The warping number fixed to be comparable to RM is 837, hence the SPR and SPGM need respectively only about 700 and 600 warping in average to converge. This is added to the decrease of memory space required estimated to 92% in the two cases where we use SP optimisation.

5 Conclusion

We proposed to use Nelder Mead algorithm initialized by gaussian mixture instead of RM to optimize AAM. Gaussian Mixture gives good initialization to the simplex which permits to reach the optimum. The test performed shows that our optimization method present a robust results comparing to RM in the case of manifold problem and the similar results in general case using only one model. The SPGM optimization reduces the memory space required to the AAM, which is a very important factor in the embedded technologies, by 92%. We notice that if the learning data base is bigger, we expect in better results. It's justified by the compactness and the good definition of the initialization space by having more learning examples. Our method allowed us to overcome the manifold problem with handle only one AAM. It permits us to get the real time executing.

References

[1] U. Canzler and B. Wegener. Person adaptive facial feature analysis. *International Conference on Electrical Engineering*, 2004.

[2] Y. Chang, C. Hu, and M. Turk. Manifold of facial expression. *International Workshop on Analysis and modeling of Faces and Gestures*, 2003.

[3] C. Christoudias and T. Darrell. On modelling nonlinear shape-and-texture appearance manifolds. *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[4] T. Cootes and P. Kittipanya-ngam. Comparing variations on the active appearance model algorithm. *In British Machine Vision Conference*, 2002.

[5] T. Cootes, K. Walker, and C. Taylor. View based active appearance models. *International Conference on Automatic Face and Gesture Recognition*, 2000.

[6] T. Cootes, G. Wheeler, K. Walker, and C. Taylor. Coupled view active appearance models. *British Machine Vision Conference*, 2000.

[7] D. Cristinacee and T. Cootes. A comparaison of shape constrained facial feature detectors. *International Conference Face and Gesture Recognition*, 2004.

[8] G. Edwards, C. Taylor, and T. Cootes. Interpreting face images using active appearance models. *Proceedings of the 3rd. International Conference on Face and Gesture Recognition*, 1998.

[9] R. Gross, I. Matthews, and S. Baker. *Generic vs Person Specific Active Appearance Models*. 2005.

[10] X. Hou, S. Li, H. Zhang, and Q. Cheng. Direct appearance models. *Computer Vision and Pattern Recognition*, 2001.

[11] C. Hu, R. Feris, and M. Turk. Active wavelet networks for face alignment. *British Machine Vision Conference, East Eaglia, Norwich, UK*, 2003.

[12] J. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 1965.

[13] A. Saad, A. El-Bialy, A. Kandil, and A. Sayed Ahmed. Automatic cephalometric analysis using active appearance model and simulated annealing. *Special Issue on Image Retrieval and Representation*, 2006.

[14] M. Stegmann. Active appearance models theory, extension and cases. *Master Thesis IMM-EKS, LYNGBY*, 2000.

[15] N. Vlassis and A. Likas. A greedy em algorithm for gaussian mixture learning. *Neural Processing Letters*, 2002.

[16] Z. XU, H. Chen, and S. Zhu. A high resolution grammatical model for face representation and sketching. *Computer Vision and Pattern Recognition*, 2005.

[17] L. Zalewsky and S. Gong. A probabilistic hierarchical framework for expression classification. *Artificial Intelligence and the Simulation of Behavior Symposium on Language, Speach and Gesture for Expressive Characters*, 2004.