3D Head Model Fitting Evaluation Protocol on Synthetic Databases for Acquisition System Comparison

Catherine Herold[1,2,3,4], Vincent Despiegel[1,2], Stéphane Gentric[1,2],
Séverine Dubuisson[4], Isabelle Bloch[1,3]

[1]*Identity & Security Alliance (The Morpho and Telecom ParisTech Research Center), France*

[2] *Morpho, Safran Group, 11 boulevard Galliéni, Issy-les-Moulineaux, France*
{catherine.herold, vincent.despiegel, stephane.gentric}@morpho.com

[3]*Institut Mines-Telecom, Telecom ParisTech, CNRS LTCI, Paris, France*
isabelle.bloch@telecom-paristech.fr

[4]*ISIR, UPMC Sorbonne Universités, Paris, France*
severine.dubuisson@isir.upmc.fr

Keywords: 3D Face reconstruction, Metrics, Evaluation protocol, Comparison of acquisition system configurations

Abstract: Automatic face recognition has been integrated in many systems thanks to the improvement of face comparison algorithms. One of the main applications using facial biometry is the identity authentication at border control, which has already been adopted by a lot of airports. In order to proceed to a fast identity control, gates have been developed, to extract the ID document information on the one hand, and to acquire the facial information of the user on the other hand. The design of such gates, and in particular their camera configuration, has a high impact on the output acquisitions and therefore on the quality of the extracted facial features. Since it is very difficult to validate such gates by testing different configurations on real data in exactly the same conditions, we propose a validation protocol based on simulated passages. This method relies on synthetic sequences, which can be generated using any camera configuration with fixed parameters of identities and poses, and can also integrate different lighting conditions. We detail this methodology and present results in terms of geometrical error obtained with different camera configurations, illustrating the impact of the gate design on the 3D head fitting accuracy, and hence on facial authentication performances.

1 Introduction

With the recent improvements of face recognition algorithms, facial biometry now offers very high performances in terms of recognition rate when acquisitions are performed in good conditions. Thanks to these advances, a lot of automatic face recognition systems have emerged, implying different levels of cooperation from the user. Among them, automatic border control gates have already been validated and deployed in several airports. Nevertheless, the main systems evaluated until now require the passengers to position themselves in front of a captor in order to acquire a frontal view, which is constraining from the user point of view. A new challenge today is to provide a simpler system for users, while ensuring high biometric performances. In case of such unconstrained scenarios, an important criterion impacting the face recognition quality is the pose of the face in the images (frontal or not), besides other factors such as the resolution or the illumination conditions.

In this paper, we consider on-the-fly systems which do not require any specific behavior of users with respect to the cameras. To optimize the system performances, it is therefore necessary to carefully position the sensors in order to deal as well as possible with the various poses of faces in the system. As 3D face fitting is an important step for face recognition against the frontal image of a passport, we compare different acquisition systems in terms of camera number and positions by their 3D fitting accuracy. We propose therefore a complete methodology to validate the 3D head model estimated from the corresponding acquisitions using geometric evaluation. Further studies on biometric evaluation and impact of ageing and expression are not part of this paper.

A crucial point when comparing different systems with respect to a given parameter is to fix all the remaining ones. However, when proceeding to real acquisitions, it is impossible to reproduce exactly the same illumination conditions and to ask users to have identical behaviors and face positions. To completely control the parameters which should be stable when evaluating the camera configuration, we propose to do the evaluation on synthetic data in order to fix all other acquisition parameters (identity, pose, illumination). Hence, no noise is introduced by variations between different not studied parameters.

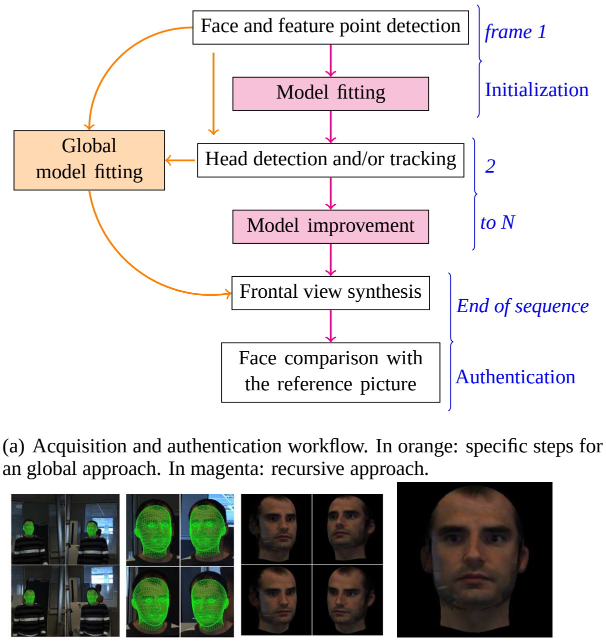
We first present our global face recognition workflow, and detail the 3D model we use in Section 2. In Section 3, we propose a methodology to evaluate different acquisition systems for face recognition gates without any real acquisition. This includes a synthetic database generation step and the metrics characterizing the quality of a configuration on these simulated sequences. We briefly present two algorithms we use for the evaluation in Section 4. The corresponding results with our methodology are detailed in Section 5, and show the impact of the gate design on the 3D head fitting quality.

2 Context

2.1 Face recognition workflow

Face recognition systems can be based on different types of sensors, such as range scanners, infrared or visible cameras. As visible range cameras are the most commonly used, we limit our study to acquisition systems based on these last sensors. Thus, the input of the face recognition algorithms is a set of video sequences, and the final output is a binary decision corresponding to the face authentication result.

The different steps of the algorithm are as follows. While the person walks in the gate, a first step of face and fiducial point detection is performed on each available view. After the initialization step, tracking and/or detections are performed in the next frames, to obtain the features needed in each frame to estimate the specificities of the face seen in the videos. As the pose is unconstrained in gate scenarios, this is done using a 3D model which offers robustness to pose variations (Blanz et al., 2005). This model is fitted to the observations, to extract the specificities of the person to authenticate. As illustrated in Figure 1(a), this fitting can be performed in a recursive way, by making a first estimation at the beginning, and then updating the model with the new observations, or globally, by using all observations together. Once the model has been fitted to the observations (Figure 1(b)), a frontal view can be generated (Figure 1(c)) to proceed to the face comparison. For our study, we focus specifically on the quality of the model fitting (Figure 1(b)). In the next part, we briefly present this shape model and the associated parameters to be estimated.



(a) Acquisition and authentication workflow. In orange: specific steps for an global approach. In magenta: recursive approach.

(b) Model fitting on the observations (c) Frontal view

Figure 1: Global workflow: from detection to authentication.

2.2 3D head reconstruction

Among the different face models which have been proposed in the past, we choose a 3D deformable shape model constructed in a similar way as the *3D Morphable Model* (*3DMM*) introduced in (Blanz and Vetter, 1999). As the final aim is to establish a comparison score between the frontal view of the estimated face and its corresponding ID picture, it is necessary to adapt the model such that it fits as well as possible the observed identity. The *3DMM* describes the face space on the two following aspects:

- The shape space, characterized by a mean shape $\overline{S}$ and a set of eigenvectors $\left\{s^i, i = 1, \ldots, M\right\}$ computed by principal component analysis over a database of aligned head scans. These vectors correspond to deformations describing shape variations in the face class. Each instance of this model can then be written as:

$$S = \kappa(\overline{S} + \sum_{i=1}^{M} \alpha_i s^i), \tag{1}$$

where α_i are the weighting parameters which characterize the similarity with the mean shape and κ is a scaling factor. The mean shape $\overline{S}$ is defined by a set of n_v 3D vertices, and each vector s^i corresponds to deformations associated with this set of points. An equivalent equation can be written for a vertex v, as $s_v = \kappa(\overline{s_v} + \sum_{i=1}^{M} \alpha_i s_v^i)$, where s_v and $\overline{s_v}$ are positions and s_v^i a deformation relative to the vertex v. A mesh is then defined from these vertices, by adding facets definition to describe the entire head surface.

- The texture, that associates a color with each vertex of the mesh.

The shape and the texture of each instance can be adapted in order to fit to the observations.

In this article, we will only evaluate the quality of the estimation for the geometrical part of the model, given various system configurations. Some instances of the morphable shape model are given in Figure 2, illustrating its variations depending on the different sets of parameter values $\{\alpha_i, i = 1, ..., M\}$.

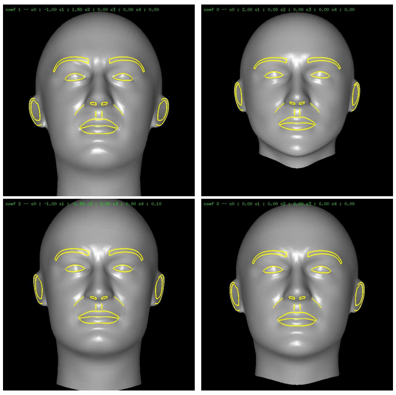

coef 1 -- c0 : -1.00 c1 : 1.50 c2 : 0.00 c3 : 0.00 c4 : 0.00
coef 0 -- c0 : 2.00 c1 : 0.00 c2 : 0.00 c3 : 0.00 c4 : 0.00
coef 2 -- c0 : -1.00 c1 : -1.50 c2 : 2.00 c3 : 0.00 c4 : 0.10
coef 0 -- c0 : 0.00 c1 : 0.00 c2 : 0.00 c3 : 0.00 c4 : 0.00

Figure 2: Some instances of the deformable shape model (all faces are generated at the same pose and scale, and with identical lighting conditions). The global shape changes for each instance, and more specifically the nose shape, the ear orientation or the chin.

The first algorithms to estimate the shape and texture parameters of the *3DMM* used stochastic gradient descent (Blanz and Vetter, 1999) or Levenberg-Marquardt optimization (Romdhani and Vetter, 2005) on a single image only. Nevertheless, the information is not complete when only single images are used to perform the fitting, especially in the case of low-resolution images. Moreover, due to the projection from the 3D world into the image plane and the occlusions of some parts of a face in an image, some information is missing and the estimation might be erroneous. This is why new algorithms based on multiple image fitting have been proposed to take multi-views or video sequences into account, thus increasing the estimation accuracy. In (Amberg et al., 2007), the fitting algorithm proposed in (Romdhani and Vetter, 2005) was adapted to a set of images acquired simultaneously, which improves the results of algorithms using only a single image. In (Van Rootseler et al., 2011), two experiments were proposed to exploit video sequences: the first one consists in estimating independently the parameters at each instant before linearly combining these estimations. The second one uses all the input images together to optimize the parameters, leading to a single estimation based on the whole sequence. The offline method we chose in this paper is close to the latter, as it estimates the set of shape parameters using all images together. Besides, we also use the recursive method proposed in (Herold et al., 2012) and based on a particle filter. Thus, temporal constraints can also be used to improve the pose and shape fitting. These two methods are summarized in Section 4 and used for our evaluation.

3 Database generation and quality measures

3.1 Methodology

The validation of real systems raises several issues. First, wide acquisition campaigns have to be performed to collect video sequences with different persons. Moreover, to compare the different acquisition systems, any parameter that could impact the performances should be fixed, in order to evaluate properly each system's characteristics. Unless the different systems are acquiring simultaneously sequences of users passing through the gate, there is no way to reproduce exactly the same trajectory of a person, thus making the comparison on identical inputs impossible. Finally, each of the systems has to be materially conceived, which is costly and time consuming. We propose a methodology based on evaluations over different sets of synthetic databases to evaluate the accuracy of pose and shape estimation algorithms with respect to different gate configurations, thus providing a way of comparing different system configurations.

In the remainder of this section, we describe the type of video sequences which have been generated to proceed to the evaluation. The process of synthetic sequence generation is summarized in Figure 3.

3.2 Identities

Each identity definition is composed of shape and texture information. The generation of synthetic views is possible using these two aspects together.

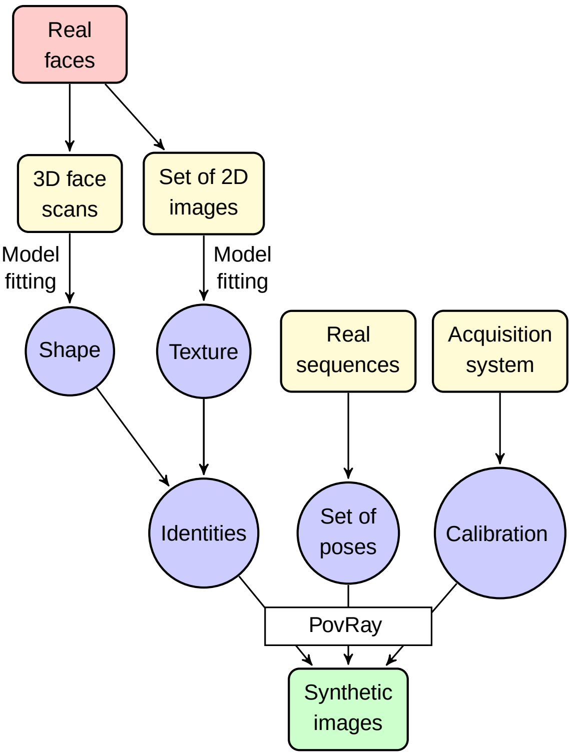


Figure 3: Synthetic sequences generation.

Shape. 40 man and 35 woman real head scans have been acquired to obtain a raw 3D representation of each face. A 3D fitting procedure has then been applied to represent these shapes with the same mesh structure as the model introduced in Section 2.2. Thus, we obtain the 3D position of each vertex of the model for the given scan. This step is necessary to compare the estimated face to the real one. The shape of each synthetic face S_{id}^{j} is then created by a combination of four 3D real head scans chosen from those available:

$$S_{id}^{j} = \sum_{i=1}^{4} c_i^j S^{\sigma(i,j)}, \tag{2}$$

with the constraints $0 \leq c_i^j \leq 1$ and $\sum_{i=1}^{4} c_i^j = 1$. $\{\sigma(i,j), i \in \{1,...,4\}\}$ defines which shapes have been used to generate the resulting one. The parameters c_i^j have been sampled randomly, the proportions of the corresponding shapes $S^{\sigma(i,j)}$ are therefore all different. Synthetic faces of men and women have been created with the corresponding real scans to respect the morphology differences.

Texture. The texture associated with the shape gives a color definition for each facet of the model. To obtain the texture for each ID, the following process is applied. Our model is fitted on images of a real face seen under various poses to extract the visible part of the texture in each of them. The extracted textures are then merged together to obtain the complete texture. For each synthetic ID, images of a different person have been used to diversify the generated textures.

Both shape and texture components of faces generated in this way come from real faces and characterize therefore realistic identities. A total of 47 identities (36 men and 11 women) has been created following this process. We consider here that the combination of independent shape and texture does not alter the validity of the resulting faces. Nevertheless, other acquisition systems generating simultaneously depth maps and corresponding 2D color images could be used to recreate synthesized sequences corresponding entirely to real faces (with the Microsoft Kinect$^{\text{TM}}$ for instance).

3.3 Associated sequences

Once the identities are defined, we have to specify scenarios to generate synthetic sequences of people walking through a gate. To this aim, we have to simulate an acquisition system and its possible configurations. In our experiments, we used the configurations illustrated in Figure 4. One of them is equivalent to a real system we have already built in our laboratory, the others are simulated variants which have not been constructed yet. For the first one, four cameras are considered, two on each side of the outdoor frame; the second one has only one camera on each side, in addition to one camera above the door. All cameras are pointing towards the center of the gate, located about two meters in front of the door. The following configurations are considered, by using some or all cameras of one of the systems (numbers refer to Figure 4):

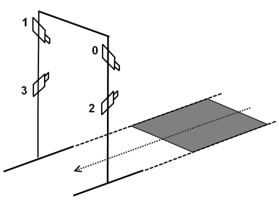


Figure 4: Configuration of the acquisition system in the 4 and 3-cameras gate.

- 2A: 2 cameras aligned vertically (0,2)
- 2B: 2 cameras aligned horizontally (0,1)
- 2C: 2 crossed cameras (0,3)
- 3A, 3B, 3C: 3 cameras (0,1,3), (1,2,3) and (4,5,6)
- 4A: 4 cameras (0,1,2,3)

Additionally to the extrinsic parameters of each camera, the impact of the image resolution can also be evaluated by generating images of different sizes. Indeed, the face and feature point detection quality depends on the resolution of the face, and this parameter should then be taken into account when evaluating an acquisition system. For further studies, lighting systems can also be added in the scene definition to evaluate their impact.

Finally, a pose has to be defined for each timestamp of the sequence. We define this set of poses given the real poses of heads observed in sequences acquired with persons using our real 4-camera system. Thus, we describe usual trajectories done by users in real systems. The poses defined in this way characterize the move of a person walking regularly from the entrance of the gate to the limit of the visible area by the cameras. Ten poses cover this move, which correspond to camera acquisitions at $5-8$ frames per second for a medium speed walk.

Figure 5 gives some examples of images which have been generated for different poses and identities. The software POV-Ray (PovRay, 2012) has been used to generate these sequences. The lighting or the image resolution can easily be modified to generate other sequences in order to evaluate the various parameters of the acquisition system outline above. The use of real data to generate the sequences in terms of faces, cameras and trajectories ensures that the generated synthetic data are close to the real ones.

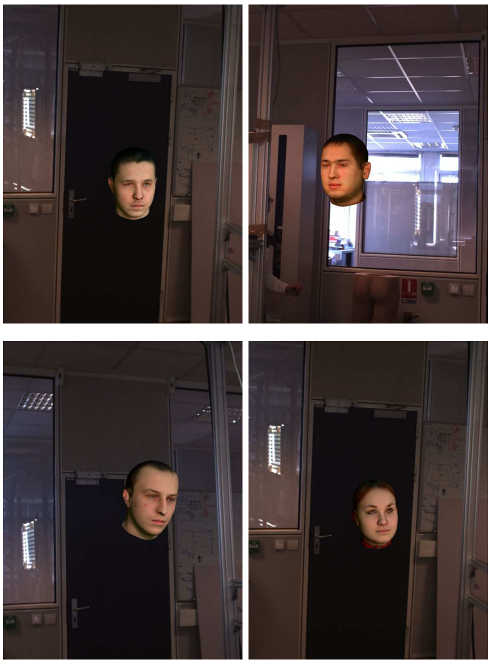


Figure 5: Examples of synthetic images generated from a 4-cameras configuration.

3.4 Quality measures

Different metrics have been proposed to evaluate the quality of a shape fitting or reconstruction (Park et al., 2002), and their significance depends on the purpose of this estimation. In our case, with the aim of comparing face information with an ID picture, we perform the evaluation via geometrical measures computed over a subset of vertices corresponding to the frontal part of the face V_f, as shown in Figure 6. First, the following 3D point-to-point error can be computed in the gate coordinate system $\mathcal{G}$:

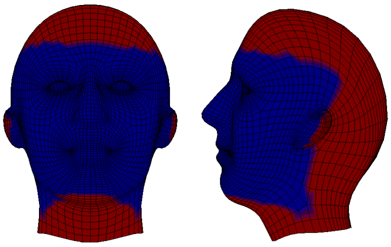


Figure 6: In blue: V_f, which represents the set of vertices selected to compute the mean error between a 3D head scan and the estimation of the same face in the video. The red vertices are not taken into account, as they are not used to compare the faces in the biometric step afterwards, and are not fitted to the observations.

$$Err_{3D} = \frac{1}{N_{V_f}} \sum_{v=1}^{N_{V_f}} \|s_v^s - s_v^e\|_2 \,, \qquad (3)$$

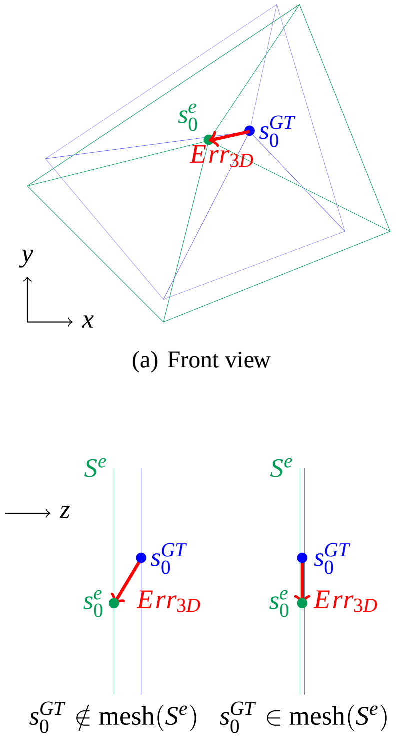
where $\|\cdots\|_2$ is the Euclidean norm, N_{V_f} is the number of vertices belonging to V_f, and s_v^s is the true position of the v^{th} vertex of the head scan in $\mathcal{G}$ computed as:

$$s_v^s = R_{GT} s_{v,0}^s + T_{GT} \,, \qquad (4)$$

where $s_{v,0}^s$ is the same vertex of the scan at frontal pose, R_{GT} the rotation and T_{GT} the translation used to generate the images. The position s_v^e is given by the estimated pose (R^e, T^e) and shape $(\kappa^e, \{\alpha_i^e, i = 1, ..., M\})$ as follows:

$$s_v^e = \kappa^e R^e (\bar{s_v} + \sum_{i=1}^{M} \alpha_i s_v^i) + T^e. \qquad (5)$$

It is necessary to take the shape and the pose estimation together into account to estimate the fitting quality. Indeed, as they are estimated jointly, several solutions of joint pose and shape can verify good head fitting on the observations, this is why we compare the solution on the vertex positions computed in $\mathcal{G}$. This measure, illustrated in Figure 7, is the closest to the error which is minimized in the shape and pose fitting procedure. It characterizes how close are the



(a) Front view

(b) Side view

Figure 7: Error Err_{3D} for vertex s_0 calculated from the real shape and the estimated one.

estimated vertices to their real positions in the gate coordinate system.

Other measures can also be computed, such as the following point-to-surface error, which compares more specifically the shape estimation to the ground truth shape:

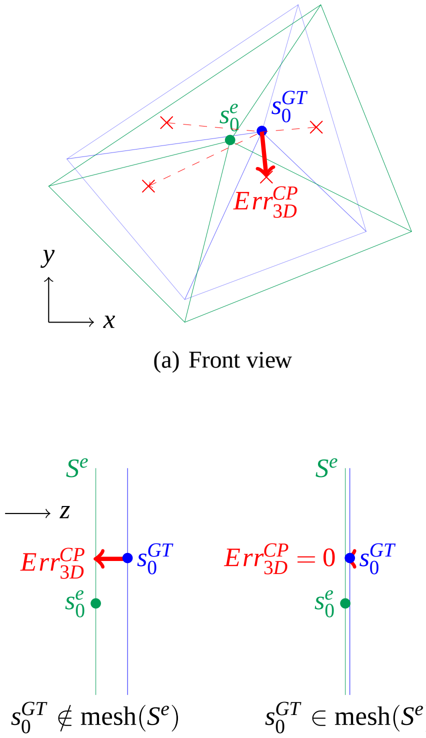
$$Err_{3D}^{CP} = \frac{1}{N_{V_f}} \sum_{v=1}^{N_{V_f}} d(s_v^s, S^e), \qquad (6)$$

where d characterizes the distance between a ground truth vertex s_v^s and the closest point of the surface described by the estimated mesh S^e. This allows local misalignment (which can happen due to missing textures in some face areas) as long as the surfaces are close to each other (Figure 8).

Finally, for comparison of 2D frontal views, the following 2D point-to-point error can also be used:

$$Err_{2D} = \frac{1}{N_{V_f}} \sum_{v=1}^{N_{V_f}} \left\| s_{f,v}^s.xy - s_{f,v}^e.xy \right\|_2, \qquad (7)$$

where $X.xy$ corresponds to the 2-dimensional vector composed of the x and y coordinates, which are the image coordinates when making the orthographic projection. $s_{f,v}^s$ is the vertex position in the frontal head scan, and the estimated vertices are computed as:



(a) Front view

(b) Side view

Figure 8: Error Err_{3D}^{CP} for vertex s_0 calculated from the real shape and the estimated one.

$$s_{f,v}^e = R_{GT}^{-1}(s_v^e - T_{GT}). \qquad (8)$$

4 Face reconstruction algorithms

The proposed evaluation protocol can be tested with any algorithm which estimates the pose and the shape in video sequences. Similar conclusions can be drawn concerning the different gate configurations which are evaluated, whatever the tested method. This is illustrated in this paper by providing results obtained with one method performing the estimation globally over the whole sequence and one sequential method. Both of them take as input the images and a set of facial fiducial points which have been automatically detected. This information is called *observations* and denoted generically y_t at time t. The output is the 3D shape estimation (the scale parameter κ and the shape deformation parameters $\{\alpha_i, i = 1, \ldots, M\}$), which is denoted by θ, and pose estimation at each instant $\{T_1, R_1, \ldots, T_T, R_T\}$. The pose and the shape estimations must be handled together, as both parameters impact the observations, and as different combinations of (shape, pose) can explain the sparse set of observations used for the fitting. We detail here shortly the two standard methods used to evaluate the pose and shape from video sequences.

4.1 Levenberg-Marquardt optimization

The Levenberg-Marquardt (LM) (Marquardt, 1963) method iteratively minimizes an energy E combining gradient descent and Gauss-Newton algorithms. In our case, we applied it in an offline manner (Figure 1(a)), estimating jointly the poses for all frames and the shape parameters (the same for the whole sequence) given the video.

This algorithm starts from an initial guess $u_0 = \left(T_1^0, R_1^0, ..., T_T^0, R_T^0, \theta^0\right)$ of all unknown values to be estimated. The 3D pose R_t^0, T_t^0 of the face at each time t is estimated given a set of 3D points reconstructed from the corresponding detections in the different images acquired at this instant using the calibration parameters. Following the method in (Umeyama, 1991), the pose parameters are adapted by fitting the mean model to these points. The initial shape deformation parameters are set to zero, which corresponds to the mean model used for the pose fitting. Given the function which associates the state u to the corresponding observations, an error can be computed between the real observations and the ones generated from u. Considering only the feature point criterion, the aim of the algorithm is to minimize the associated energy: $E = \sum_{t=1}^{T} \frac{1}{D(t)} \sum_{p=1}^{D(t)} ||m(p,t,u) - o(p,t)||_2^2$, where $D(t)$ is the number of detected feature points at time t, $o(p,t)$ their 2D positions and $m(p,t,u)$ the projection of the corresponding points from the model on the images given the current pose and shape estimations. We aim at minimizing this error, by applying recursively correction steps to u, given the current error and the Jacobian of the function f.

This method uses all frames together to proceed to the optimization. Thus, a single value θ is estimated, common to all frames. Indeed, as the shape parameters characterize the identity, these are supposed to be constant (assuming that the person does not change its facial expression).

4.2 Particle filter optimization

The particle filter (PF) method used to evaluate the pose and shape throughout a sequence is inspired from (Herold et al., 2012). The idea of this algorithm is to integrate the shape parameters θ to be estimated in the particle state, and to update the density $p(\theta)$ with each new observation. The particle weights are computed by comparing the projection of the landmarks given the particle state (a pose and a set of parameters) to the ones detected in the images. This method is applied recursively (Figure 1(a)), meaning that the shape estimation is updated at each instant given the new observations.

At each time t, the following procedure is applied given the set of N particles and the new observations y_t:

- for each particle i: (i) move the static shape parameters to obtain a new hypothesis $\theta_t^{(i)}$; (ii) estimate the pose $R_t^{(i)}, T_t^{(i)}$ given a subset of the feature point detections and the particle shape parameters $\theta_t^{(i)}$; (iii) update its weight by computing the likelihood of the state with the observations;
- compute the current output state (R_t, T_t, θ_t). This is done by choosing the particle with the highest weight, or by computing the weighted mean over the set of particles.

Unlike the LM method, only the observations until time t are used when computing the evaluation at this instant. As only few features are used in each view to evaluate the pose and the shape of the face, this method allows us to maintain different shape parameters hypotheses and to validate them when new discriminant observations are available.

5 Evaluation

In this section, we apply the proposed methodology to evaluate the head model fitting quality depending on the number and the positions of the cameras used to acquire the images in the gate. This evaluation is done considering the results obtained on the synthetic sequences with the two fitting algorithms presented in Section 4. The LM implementation is based on the *levmar* library available online (Lourakis, 2004). We do not use the known feature point positions as inputs for the two fitting algorithms. Instead, we launch the feature point detectors used for real sequences, in order to have the same noise and eventual bad or missing detections associated to these detectors.

Errors presented below are not given in pixels but in percentage of the distance between the two eyes to have an absolute measure. Figure 9 illustrates the error distribution for a subset of sequences of the database using some of the configurations listed above. The PF method has been used to generate these results, which are given as a percentage of the inter-eye distance (*ied*). We can see that in most cases, using 4 cameras (green bars) outperforms all other configurations, as no sequence has an error above 11% of the *ied*. The 3-cameras configuration (blue bars) comes after, with almost all errors below 11% of the *ied*. Finally, with only 2 cameras, some errors reach 14% of the *ied*. The errors are however smaller with the *2C* configuration (in purple) where more points can be seen thanks to the viewpoint change. Figure 10 illustrates how well the estimated model is consistent with the observations. Only a small set of feature points related to the edges plotted on the left are used to evaluate the pose and shape parameters.

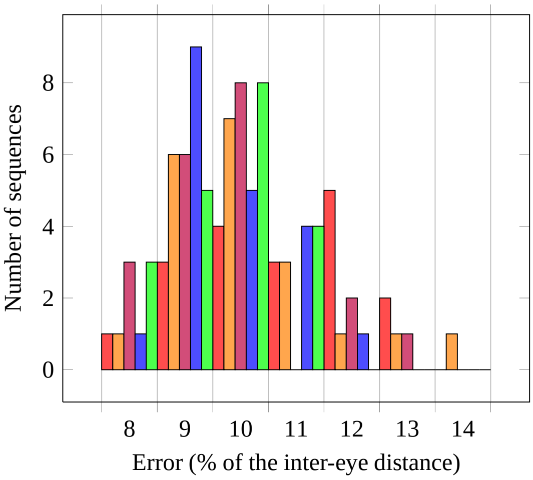


Figure 9: Error (Err_{3D}) variations (in % of the inter-eye distance) with different camera configurations for some sequences: two cameras (2A 🟥, 2B 🟧, 2C 🟪), three cameras (3B 🟦) and four cameras (4A 🟩).

Table 1 shows the mean errors computed over all the sequences of the database with respect to the camera configurations with the PF and the LM methods. We can see that the error and the number of cameras are correlated. The general trend is that the fitting is improved when more views are used. Nevertheless, for a given number of cameras, their positions can also impact the quality, as shown for instance for 2*A* (vertical alignment) and 2*C* (crossed cameras). The improvement with the PF method is due to the sampled subset of feature points which leads to more robustness to the outliers. The stability of this sampling based method has been verified by running it five times for the whole set of sequences. The standard deviation of the means of Err_{3D} computed on configuration *2A* (resp. *4A*) is 0.12% (resp. 0.05%) of the inter-eye distance, which is small relatively to the highest error variations observed between the different configurations.

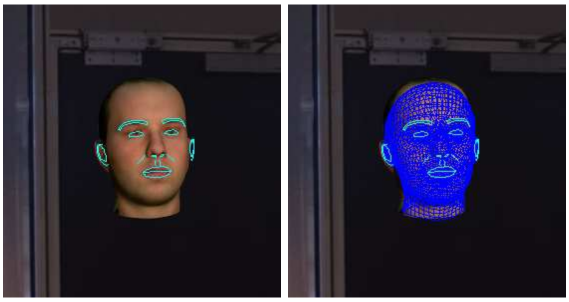


Figure 10: Example of edge and mesh fitting with the particle filter method (zoom on the face area). The inter-eye distance on the input image is 36 pixels.

System	2A	2B	2C	3A	3B	3C	4A
LM Err_{3D}	23.3	22.8	22.9	22.7	22.9	26.2	22.2
PF Err_{3D}	12.2	11.8	11.0	10.8	10.5	10.4	10.5
LM Err_{3D}^{CP}	16.7	16.0	16.1	16.0	16.2	18.5	15.4
PF Err_{3D}^{CP}	6.6	5.8	5.6	4.6	4.5	5.0	4.7
LM Err_{2D}	9.2	9.5	8.1	7.9	8.3	8.9	7.6
PF Err_{2D}	8.4	8.2	8.2	7.6	7.2	6.5	6.9

Table 1: Mean errors with the PF and the LM methods given different camera configurations. For the PF method, the mean is computed over 5 runs. Errors are given as a percentage of the inter-eye distance.

The error magnitude in this table should be correlated with the resolution of the images (600 × 800), the distance of the person to the sensors (between 1.5 and 2 meters), and the sparse distribution of the features used for the fitting. Nevertheless, the relative gain between the worst and the best configuration reaches 14.7% (resp. 4.7%) for the PF method (resp. LM method) considering the error Err_{3D}.

Figure 11 illustrates the error repartition over the face for three faces of our database, using the 4-cameras configuration. The 3D errors are not distributed uniformly over the mesh, because we only use a few fiducial points to perform the fitting. Indeed, in some areas of the mesh such as on the neck, above the ear or on the cheeks, there are therefore no clues to guarantee the fitting. This explains the higher errors in these areas, in comparison with the eyes areas, where the error is less than 10% of the inter-eye distance.

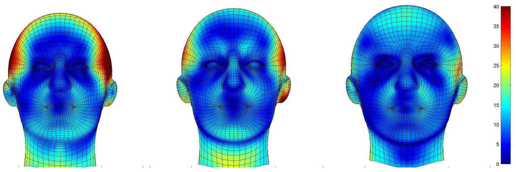


Figure 11: 3D error (Err_{3D}) distribution over the face for three faces of the synthetic database. The particle filter method with 4-cameras has been used to estimate the shape. Errors are given as a percentage of the inter-eye distance.

Influence of the shape and texture on the accuracy. We now verify the influence of texture or shape on the pose and shape estimation quality. To this end, we used two new sets of synthetic data:

- base B_{shape}: ten sequences, changing only the shape from one sequence to another one, all other parameters remaining fixed;
- base B_{tex}: ten sequences, changing only the texture from one sequence to another one.

We evaluated the pose and shape estimation using the Levenberg-Marquardt algorithm. The accuracy variation for each of these bases is given in Table 2. We report only the 2D errors (Err_{2D}) using the 3-cameras configuration 3A.

Variation	Mean	Sigma	Min	Max
Shape (B_{shape})	7.65	1.8	5.72	11.87
Texture (B_{tex})	7.57	0.45	7.03	8.46

Table 2: Error variations depending on shape or texture variations only. The Levenberg-Marquardt optimization has been used on configuration 3A.

The results are significantly more stable with the base B_{tex} than with the base B_{shape}. This can be explained by the fact that texture variations slightly alter the detector quality at fixed pose and shape. For instance, the appearance of an eye corner does not change considerably for different facial textures. The detected points are therefore almost the same for all sequences of B_{tex}, leading to very similar estimations.

For the base B_{shape}, the texture and the poses are fixed for all sequences, so we can assume that the quality of the detections is equivalent for all of them. Nevertheless, the errors obtained for this base are more varied than for B_{tex}, which is due to the shape variability in the sequences. Indeed, some real shapes cannot be generated because of the model constraints. Some faces will therefore be easy to represent and lead to low errors, but for others, it will not be possible to fit correctly the model to the data. This explains why it is important to use real head scans when generating the synthetic sequences, in order to reproduce this problem when evaluating the pose and estimation algorithms.

6 Conclusion and future work

We have presented a complete workflow to evaluate configurations of face recognition gates in terms of 3D fitting quality. The methodology we propose is based on synthetic data, which can be generated with any number and configuration of cameras, lighting condition and resolution, while maintaining other conditions fixed (identities, face poses). This allows us to test an unlimited number of alternatives, without bias introduced by people behavior and trajectory variations, or constraints related to real campaign acquisitions and material conception. The evaluation is based on the accuracy measure of the 3D head fitting, which is easily computable as we benefit from the ground truth used to generate the sequences. The general trend shows that increasing the number of cameras improves the accuracy of the estimation. Moreover, for a fixed number of cameras, their position also impacts the accuracy: diversifying the points of view increases the estimation quality (two crossed cameras are better than two vertical cameras...). This factor can be optimized with simulations, thus limiting the number of real systems to build when making the real data evaluation (for instance, evaluation of the configuration 3C is not available with the initial 4-cameras system). In the future, such studies could be extended to other factors, such as lighting and expression.

We limited our evaluation to geometrical results on synthetic data. Another extension to this work would be to develop the following aspects. First, it would be interesting to compute geometrical measures on real data. The difficulty of this point is to get the real position of each face vertex during a sequence. Additional depth sensors should be used to this aim, or, at least, the ground truth of the face should be known (using a 3D scanner for instance). Besides, the relation between biometric performances and errors on the estimation (3D pose and shape) should be deepened, with respect to different face comparison algorithms.

REFERENCES

Amberg, B., Blake, A., Fitzgibbon, A., Romdhani, S., and Vetter, T. (2007). Reconstructing High Quality Face-Surfaces using Model-Based Stereo. In *International Conference on Computer Vision*, pages 1–8.

Blanz, V., Grother, P., Phillips, P., and Vetter, T. (2005). Face Recognition Based on Frontal Views Generated from Non-Frontal Images. In *Conference on Computer Vision and Pattern Recognition*, pages 454–461.

Blanz, V. and Vetter, T. (1999). A Morphable Model for the Synthesis of 3D Faces. In *SIGGRAPH*, pages 187–194.

Herold, C., Despiegel, V., Gentric, S., Dubuisson, S., and Bloch, I. (2012). Head Shape Estimation using a Particle Filter including Unknown Static Parameters. In *International Conference on Computer Vision Theory and Applications*, pages 284–293.

Lourakis, M. (2004). levmar: Levenberg-Marquardt Nonlinear Least Squares Algorithms in C/C++. `http://www.ics.forth.gr/~lourakis/levmar/`.

Marquardt, D. (1963). An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441.

Park, I. K., Lee, K. M., and Lee, S. U. (2002). Efficient Measurement of Shape Dissimilarity between 3D Models Using Z-Buffer and Surface Roving Method. *EURASIP*, 2002(10):1127–1134.

PovRay (2012). Persistence of Vision Raytracer (version 3.6). `http://www.povray.org/download/`.

Romdhani, S. and Vetter, T. (2005). Estimating 3D Shape and Texture using Pixel Intensity, Edges, Specular Highlights, Texture Constraints and a Prior. In *Conference on Computer Vision and Pattern Recognition*, pages 986–993.

Umeyama, S. (1991). Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380.

Van Rootseler, R. T. A., Spreeuwers, L. J., and Veldhuis, R. N. J. (2011). Application of 3D Morphable Models to Faces in Video Images. In *Symp. on Information Theory in the Benelux*, pages 34–41.