Is it a face ? How to find and validate a face on 3D scans

Przemyslaw Szeptycki, Mohsen Ardabilian, Liming Chen

Ecole Centrale de Lyon, 36 av. Guy de Collongue, 69134 Lyon, France

{przemyslaw.szeptycki, mohsen.ardabilian, liming.chen}@ec-lyon.fr

Abstract

Rapid development of 3D scanning technologies lead researchers to use them for people recognition, cameras are faster, models are less noisy and with higher resolution. 3D facial models have been widely used for many biometrics applications. Nevertheless 3D face recognition topics mainly assume that scans contain a face and the face is mostly in a frontal position. In real world we might have a situation, where a scanned model is not sufficient for recognition.

In this article we propose a generic face model validation algorithm which can exclude non-face models from recognition query. The algorithm was tested on more than 1500 range scans including face and non-face models. Obtained results prove, that the generic model validation approach can be used to reject non-face models from the recognition pipeline.

Key words

3D face, Curvature, face validation.

1 Introduction

The use of 3D face models has emerged as a major face recognition solution in the last years to deal with unsolved issues, e.g. lighting and pose variations, for reliable 2D face recognition solutions [1]. Nevertheless for recognition purposes faces are generally detected manually and registered in standard position. The face detection/validation literature mostly concerns face detection on texture images of scanned scene. Such detection is dependent to the face rotation and the lighting conditions. Likewise challenges arise from the fact that the scanned persons are non-cooperative.

To make 3D face recognition algorithms automatic and insensitive to the lighting and pose variations changes, a face has to be detected directly on a 3D model without reinforce from the texture. In many cases the face validation problem has been decomposed to a problem of face anchor points localization but with strong assumptions about the position and the orientation.

In this paper we present an algorithm for automatic face validation based on anchor points detection and distance between a generic face model and a query model, which let us to exclude non-frontal faces and non-face objects from a query. In order to ascertain the accuracy, the algorithm was tested on more than 1500 objects including faces and non-faces. The results prove that the method is stable and can reject with high accuracy non-face objects from the query.

The rest of this paper is organized as follows. Section II overviews the related work. Section III describes our generic model face validation algorithm. Description of a test data set is given in section IV. Experimental results and conclusion ends this paper.

2 Related Work

Face validation problem on 3D models is mainly decomposed to the problem of anchor points localization. To localize main points on the face researchers are using different tools and methods like geometrical analysis [2] or differential geometry [3].

To detect automatically face on a scanned scene, Mian et al. in [2] proposed a slice searching algorithm for the nose tip point. Searching for nose tip candidates is performed on each slice of the model. Authors center a circle on multiple horizontal intervals on the slice and inscribe a triangle using center of the circle and points of intersection of the slice and the circle. The point with maximum triangle altitude becomes the nose tip candidate on the slice. The point which has maximum confidence is taken as the nose tip of the face. Authors tested the algorithm on the FRGCv2 dataset, which is 2.5D face dataset. In this article the problem has been simplified to face detection on the face models without any non-face examples. Moreover considering only horizontal slices makes the algorithm dependent to rotations.

Curvature based approach has been proposed by Colombo et al. in [4], where authors describe how a face can be detected based on the curvatures analysis. Approach used in the article is based on HK-Classification which can portion a face surface to regions of convex and concave shapes. Their algorithm in comparison with previous one has a validation stage. In the validation stage authors are comparing segment from a face, which has been cropped based on anchor point candidates (the nose tip and the eyes corners). Such validation stage based on range image is exposed to holes and spikes in the model.

Researchers mainly search and detect a face on 3D models based on face anchor points, therefore rest of related work will be devoted to the anchor point localization on 3D face

models.
Many methods for landmark validation are not invariant to different type of rotations. Lu et al. in [5] proposed an approach which is based on the assumption that the nose tip has the highest Z value. Based on that, they rotate the face and in each rotation search for the largest value in the Z-axis. Verification stage, based on the vertical profile in the nose tip point gives them correct result which is not sufficient for the rotation along the Z-axis.
In this paper we propose generic face model validation and curvature based face main points searching algorithm. First stage in the algorithm is convex and concave regions searching and main point candidates localization based on maximum Gaussian curvature value in each region. Validation step relay on a scaled face generic model fitting to each combination of main points candidates. Such method is invariant to the rotation on all axes, invariant to the resolution and also do not needs help from texture images.

3 Differential geometry tools and generic face model for face validation

Our algorithm for automatic face models validation is based on a generic face model fitting to adequate face regions. Searching of the appropriate convex and concave regions is performed by Mean and Gaussian curvatures analysis on each vertex. Classification based on signs of the curvatures assigns the vertex to the convex or the concave regions. Searching is performed using large face area for curvatures calculation to estimate the most marked out regions (fig. 1). Having those regions and based on the generic face model (described in section 3.1) face can be validated. More details about the curvatures calculation can be found in [3].

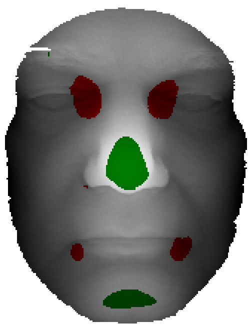

Figure 1 – *Main convex and concave regions on the face model (red - concave regions, green - convex regions).*

Points candidates extraction. Having Gaussian and Mean curvatures values calculated on each vertex, convex and concave regions extraction can be performed. To validate a query model, most marked out regions were chosen. Based on HK-Classification such regions can be easily extracted.
Models, face or non-face, can be more complicated having plenty of convex and concave regions. To reduce number of regions and to select only the most marked out, HK-Classification thresholding process can be performed. To localize correct regions which belong to the eyes and the nose Gaussian curvature was thresholded (figure 1 shows result for the main regions extraction from a face model).

Having those regions, main point searching can be performed in each region separately. To localize the nose tip and the inner eyes corners in each region simply maximum Gaussian curvature value was localized. Max Gaussian curvature value corresponds to the most convex/concave point in the region. Point with the maximum Gaussian curvature in **the convex regions will become the nose tip candidate**, while point with the maximum Gaussian curvature in **the concave regions will become the inner eye point candidate**.

3.1 Face generic model building

Our generic face model (figure 2) was built based on 40 models selected from the IV2 [1] dataset. The generic model is composed from 9 main face points (fig. 2) which positions were calculated based on selected 2.5D facial models. The models were firstly manually landmarked with 9 feature points. Next, all models were translated and rotated to a frontal position having the nose tip as the origin. Fusion of all models relay on mean main point position calculation in 3D space. The generic model was further normalized so that the distance between the two eye inner corners was equal 1 mm.

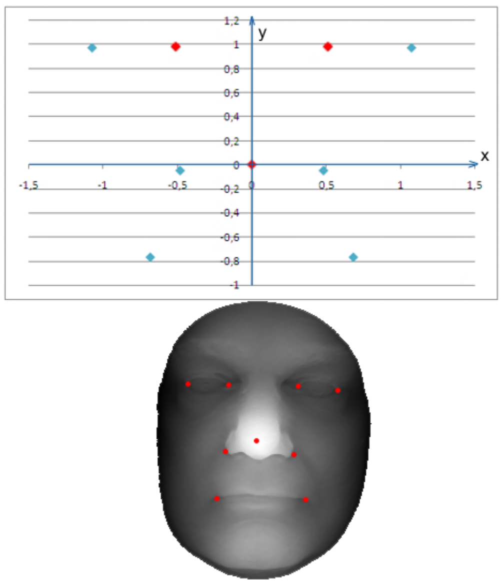

Figure 2 – *Generic model made based on 40 models from IV2 data set (x,y projection, red points - main three points - inner corners of the eyes and the nose tip).*

1. IV2 - French biometric data base created in cooperation of few laboratories (http ://lsc.univ-evry.fr/techno/iv2/PageWeb-IV2.html).

3.2 Face validation

The main part of the whole algorithm is based on the generic face model fitting to the query model. Well fitting of the generic model to the query model, means that the distance (equation 1) between points of the generic model and the closest points of the query model is small and less than some threshold.
The distance can be calculated based on equation :

$$dist = \sqrt{\sum_{i=0}^{n}(GP_i - CPQM(GP_i))^2}, \qquad (1)$$

where n is a number of points in the generic model (in our case 9), GP_i is a point in the generic model, $CPQM(GP_i)$ is the closest point on the query model to the point of the generic model.
To calculate distance between the generic model and the query model, first of all a correspondence between those two has to be established. The correspondence can be established based on the extracted concave and convex points from the query model, **related to the nose tip candidates and the inner eyes corners candidates on the face models**, (section 3) and the same points from the generic model (figure 3).
Query model can have numerous of convex and concave regions, where from each region in the previous section we have extracted the most convex and the most concave point. Having unknown number of the convex and the concave points (related on face models to the nose and the eyes) candidates and without any prior knowledge about the query model all combinations of points candidates have to be considered.
Heaving two sets of points, the nose tip candidates and the eyes inner corner candidates (section 3), each combination of three points (two concave points and one convex point) is considered to calculate translation and rotation between the generic face model and the query model. Figure 3 shows some correspondences between the Generic Model and a Query Model. Rotation and translation between two sets of points with known correlation can be calculated using Singular Value Decomposition algorithm [6, 7, 8], which is a matrix decomposition algorithm, used iteratively in the Iterative Closest Point algorithm. SVD let us to find fine translation and rotation between objects in correspondence based on their covariance matrix.
Having translation and rotation for each selected combination of points, generic model can be moved over the query model surface. To deal with scale changes, generic model was scaled based on the distance between the concave point candidates.
Now when scaled generic model is over the surface and anchored in the concave and the convex points, the distance can be calculated based on equation 1. This algorithm has to be repeated for all combinations of main point's candidates and the smallest distance from all distances between the generic model and the query model surface which is less than some threshold can validate face and also pick up correct anchor points on the model which will be the nose tip and the inner corners of the eyes.
Tests made on face models lead that the sum of distances between generic model and face model cannot be more than 70 mm which means that each point of generic model have to be in the distance less than 7.7 mm.

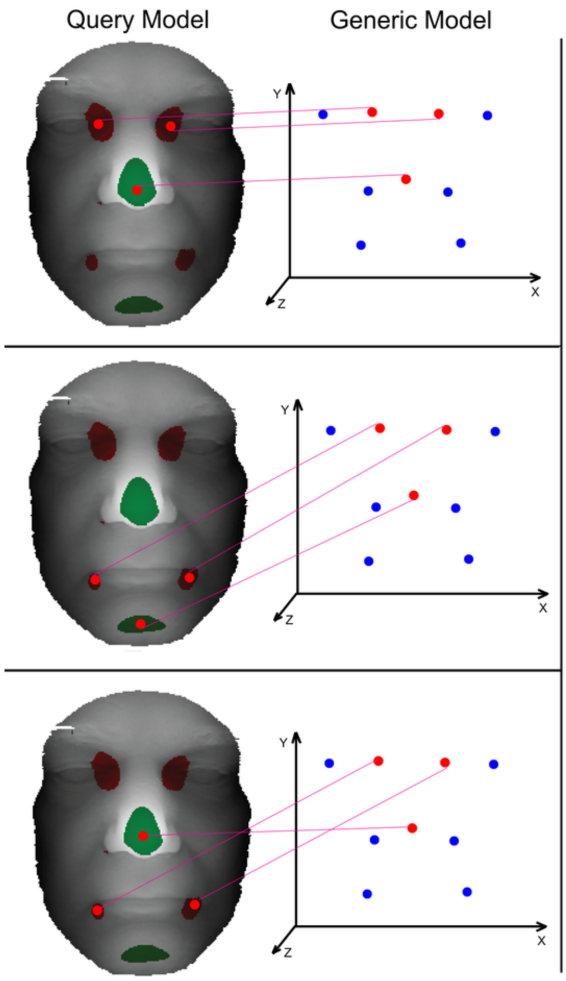

Figure 3 – *Few examples of different correspondence combinations of main points from the generic face model and the points candidates from the query model.*

4 Data set characteristic

The aim of this article is to deal with the problem of models validation for recognition purposes. In real world subjects might be non-cooperative, which means that can move during scanning process. This kind of situations causes many problems during acquisition and recognition.
The main goal building the test dataset was to simulate non-cooperative behavior of the subject. The whole test dataset contains three different datasets/subsets and can be divided based on their origin.

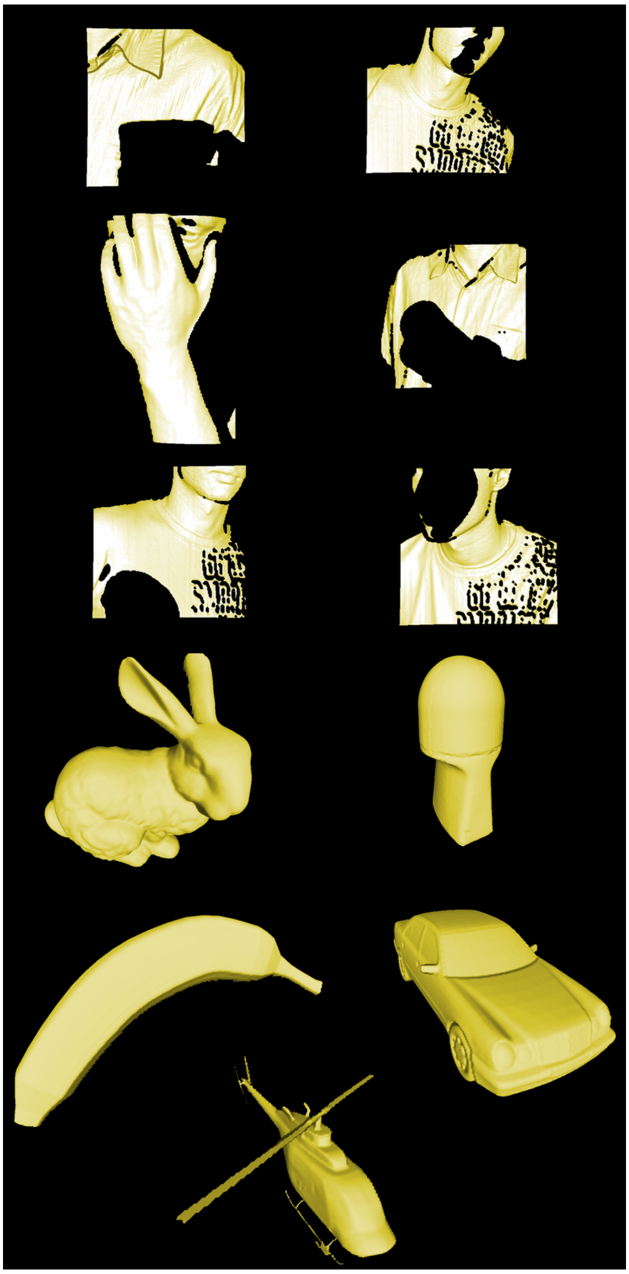

Figure 4 – *Examples of wrong query models.*

Figure 5 – *Examples of correct query models.*

The first part of the test dataset is a dataset called "Unsupervised conditions" (tab. 1). It is our own data set, in which uncontrolled conditions were simulated. Data set contains 77 non-face models scanned during subject movement which causes scanning of some clothes or part of the face (upper part of figure 4) and correctly scanned faces (25 models) with some rotations and partial occlusions. All models were scanned using non-contact 3D digitizer Minolta Vi-300 with resolution 400x400 points (fig. 7).
To increase number of non-face models dataset has been supplemented by adding some models from Stuttgart Range Image Database [9] (tab. 1) like a bunny or a car (lower part of figure 4). Stuttgart Range Image Database contains a collection of synthetic range images taken from high-resolution polygonal models available on the web. Whole data set contains 42 models where each model has 258 range scans which give 10836 range models. For our purposes only part of this dataset was added to the experiments.
To validate algorithm ability to accept face models, test dataset has been supplemented by adding 933 models from FRGCv1.0 dataset [10]. This dataset is a frontal position face dataset which can be used to prove that algorithm is able to pass face models to the recognition pipe.

Un-supervised conditions	count
part of face/shoulders	77
Face-frontal	12
Face-occlusion	9
Face-rotation	4
SUM	102
FRGC v1.0	count
Frontal faces	933
Stuttgart Range Image Database	count
agfa	66
auto	66
banana	66
bunny	66
copter	66
bunny	66
deo	66
duck	66
SUM	528
SUM of all	**1563**

Table 1 – *Test data sets characteristic.*

5 Experiments

Based on division of the test dataset (tab. 1), algorithm ability to reject non-face models and to keep face models for future recognition purposes was performed.
I order to assess propriety of face models validation algorithm, whole test data set was processed. The threshold dividing test set to face and non-face models was set to 70 mm.
Test 1 : ”Un-supervised conditions”.
The first test was made on our own models scanned in the laboratory (fig. 7) (models simulate un-supervised conditions during acquisition). This test had to prove that algorithm is able to reject non-face models (part of a face or shoulders) from a recognition pipe while face models with some rotations or small occlusions should pass the conditions.
During this test all **face models** (25) were accepted : the minimum distance (equation 1) between the Generic Model and a face model was between 10.39 mm and 54.73 mm, much less than the face /non-face threshold. In case of **non-face models**, only one non-face model per 77 passed the distance condition and was accepted as a face model, rest of them were rejected as a non-face ones. The correct distance between this model and the Generic Model was 53.9 mm.
Test 2 : ”FRGC v1.0”.
The aim of the second test was to ascertain that the algorithm is able to labeled face models as correct ones. To ensure large variation of faces FRGCv1.0 dataset was chosen with 933 2.5D face models with frontal positions. All from 933 face models pass the test of a face validation with correct distances to the Generic Model between 9.14 mm and 60.40 mm.

Test 3 : ”Stuttgart Range Image Database”.
The last test was made to ascertain algorithm ability to reject non-face models. Test was made on the subset of ”Stuttgart Range Image Database” with 528 2.5D non-face models. During this test all non-face models were rejected as not correct ones with distances to the Generic Face Model between 81.46 and 359.10

All tests results can be seen in figure 6, where data has been organized to show crossing part of face and non-face models, distance of face models to the Face Generic Model has been sorted in descending order while for non-face models in ascending order. Using 958 face models and 605 non-face models for validation purposes only one model did not pass the conditions and has been incorrectly accepted as a face model, all face models have passed validation conditions and was labeled as face models.

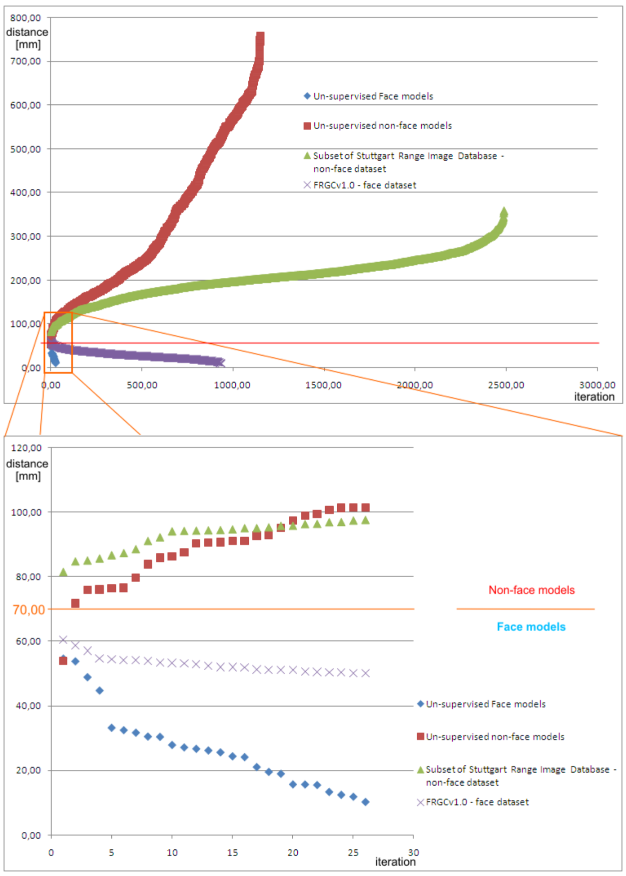

Figure 6 – *Results of face models validation organized to show crossing part between face and non-face models (vertical axis shows distance between the Generic Model anchored in selected convex and concave points and a query model, horizontal axis shows iteration, plot has been divided to different subsets in the data set).*

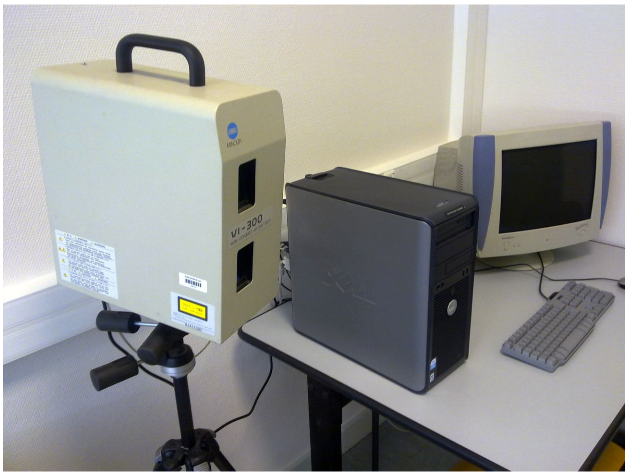

Figure 7 – *Scanning environment to simulate "Unsupervised conditions".*

6 Conclusion and future work

In this paper we presented an automatic algorithm for 3D face validation purposes. Recently 3D face recognition has been perceive as a major face recognition solution, while 3D face validation or searching on 3D scans is omitted.

Our solution for a query model validation/labeling as a face or a non-face model is based on main convex and concave points searching on the query model. Those points correspond to the inner corners of the eyes and the nose tip on face models. Based on those points correspondence between the Generic Face Model can be set up. The validation process is based on distance measurement between the query model surface and the Face Generic Model moved over the face surface and anchored in many combinations of the main convex and concave points from the query model. The smallest distance between the Generic Face Model anchored in one of main query model point's combination and the query model surface gives the measurement score. If measurement score is less than face/non-face threshold, query model is considered as a face model otherwise model is labeled as a non-face model and can be rejected from the recognition pipeline.

Presented results prove, that the Generic Face Model validation algorithm is stable (1/1563 models has been incorrectly labeled) and accepts with high accuracy face models (all, 958 face models were labeled as a face).

In our future work, we are moving to partial face models validation, to give more information to recognition algorithms where a big advantage will be to know, what part of model is missing.

7 Acknowledgment

This work was partially carried out within the French FAR3D project supported by ANR under the grant ANR-07-SESU-004 FAR3D.

Références

[1] Timothy C. Faltemier, Kevin W. Bowyer, et Patrick J. Flynn. Rotated profile signatures for robust 3d feature detection. *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, 2008.

[2] Ajmal Mian, Mohammed Bennamoun, et Robyn Owens. An efficient multimodal 2d-3d hybrid approach to automatic face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(11) :1927–1943, 2007.

[3] Przemyslaw Szeptycki, Mohsen Ardabilian, et Liming Chen. A coarse-to-fine curvature analysis-based rotation invariant 3d face landmarking. *International Conference on Biometrics : Theory, Applications and Systems*, pages 1–6, 2009.

[4] Alessandro Colombo, Claudio Cusano, et Raimondo Schettini. 3d face detection using curvature analysis. *Pattern Recognition*, 39(3) :444–455, 2006.

[5] Xiaoguang Lu et Anil K. Jain. Automatic feature extraction for multiview 3d face recognition. *Automatic Face and Gesture Recognition*, pages 585–590, 2006.

[6] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *Transactions on Pattern Analysis and Machine Intelligence*, 13 :376–380, 1991.

[7] D.W. Eggert, A. Lorusso, et R.B. Fisher. Estimating 3-d rigid body transformations : a comparison of four major algorithms. *Machine Vision and Applications*, 9 :272–290, 1997.

[8] Gaojin Wen, Zhaoqi Wang, Shihong Xia, et Dengming Zhu. Least-squares fitting of multiple m-dimensional point sets. *The Visual Computer*, 22(6) :387–398, 2006.

[9] G. Hetzel, B. Leibe nad P. Levi, et B. Schiele. 3d object recognition from range images using local feature histograms. *CVPR*, 2001.

[10] P. Jonathon Phillips, Patrick J. Flynn, Todd Scruggs, Kevin W. Bowyer, Jin Chang, Kevin Hoffman, Joe Marques, Jaesik Min, et William Worek. Overview of the face recognition grand challenge. *Computer Vision and Pattern Recognition*, 1 :947–954, 2005.