Improving Facial Depth Data by Exemplar-based Comparisons

Leandro Dihl[1]
leandro.dihl@isr.uc.pt
Leandro Cunha[1]
lmvcruz@isr.uc.pt
Nuno Gonçalves[12]
nunogon@deec.uc.pt

[1] Institute of Systems and Robotics
University of Coimbra
Coimbra, Portugal

[2] Portuguese Mint and
Official Printing Office
Lisboa, Portugal

Abstract

3D face models are widely used for several purposes, such as biometric systems, face verification, facial expression recognition, 3D visualization, etc. They can be captured by using different types of devices, like plenoptic cameras, structured light cameras, time of flight, etc. Nevertheless, the model generated by all of these consumers devices are very noisy. In this work, we present a filtering method for meshes of faces preserving their intrinsic features. It is based in an exemplar-based neighborhood matching where all models are in a frontal position avoiding rotation and perspective drawbacks. Moreover, the model is invariant to depth translation and scale. The obtained results showed that this method is robust and promising.

1 Introduction

In the last years, it has been increased the number of applications using faces captured from consumer 2.5D cameras. A relevant issue is in terms of personal recognition and security. Despite the 2.5D face acquiring technologies have improved in last years, the resultant outputs are still noisy. Deformed and noisy meshes, holes, and errors are common. These problems impact on 3D visualization processes, face verification models, person recognition, pose detection and facial expression recovery. The outputs of these devices need to be improved in order to guarantee reliable and satisfactory results by preserving their geometrical structure.

In this sense, we are developing a specific filter for face meshes, that is, a content-aware filter based on copies where it is proposed to correct each point of a given mesh by comparing the local neighborhood using a set of copies. We take advantage of the previous knowledge of the models (in this case the faces) to improve this comparison. In this way, one begins by detecting points of features and creating point correctors for the given regions, and then using only the corresponding regions to correct a point of the filtered mesh.

2 Related Work

Many methods have been proposed for denoising and smoothing meshes improving the output from 3D scanners or 3D cameras. They normally work locally and iteratively on the data structure of the mesh surface. But the main deficiency of these methods is that they are designed for generic structures not taking into account the intrinsic characteristics of their models. Yagou et al. [7] use the mean and a median filter applied to the normal vector of triangles. Here in this method, the normal vector of a triangle was modified according to the normal vectors of its neighbors and then the vertices positions are updated. The works proposed by [[1], [8], [2]] also perform mesh denoising by filtering the face normals, using a general formulation of the bilateral filter with some variants. Due to its simplicity and feature-preserving capability, the bilateral filter has been used in numerous applications in image processing.

3 Model-Exemplars Neighborhood Comparison

Given a facial RGBD model, the filtering method replaces the depth of each point through a comparison between its neighborhood and the neighborhood of the respective points into previously provided exemplars. figure 2 illustrates the pipeline.

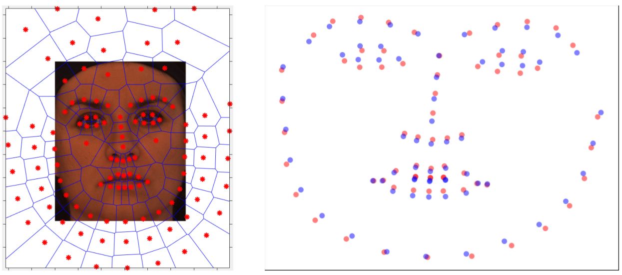
The proposed method is based on a model covering, created from a set of Facial Feature Points (FFP)(Section 3.1). For filtering purpose, it is important to guarantee a proper comparability between neighborhoods of points of the model and points from the exemplars. We achieve this goal by an alignment of the facial features points followed by a resampling of the model by the same frequency of the exemplars (Section 3.2).

The proper frequency sampling allows the neighborhood comparison of different models (including models from different types of sensors). Moreover, the feature points allow to subdivide the face area into subregions and constrain the filtering comparisons to the respective region. This constraint implies a proper correction (the neighborhood will be corrected by another of the same part of the face) and a search acceleration.

In the normalized depth, the filtering phase is applied to the resampled model (Section 3.2). The resampling guarantees a model resolution invariance, the covering alignment faces translation, rotation and scaling invariance, and the neighborhood normalization guarantees depth translation and scaling invariance.

3.1 Face Covering and Alignment

As mentioned previously, the covering is based on FFP. This phase is compound into the following steps: detection, insertion, and removal of FFP, the Voronoi diagram creation of the FFP, and finally the definition of search regions.



(a) (b)

Figure 1: The Figure (a) shows the face covering method with the divided regions on the face. The (b) shows the alignment of the faces.

The subdivision process shall meet three conditions:

1. All faces must have FFP at correspondent places;
2. The union of regions must cover the whole face;
3. The area of each region must be inversely proportional to the average of expected noise.

These conditions are achieved by an extension of the set of FFP proposed by Kazemi and Sullivan [4]. After the points detector, new feature points are inserted to fill the regions where the detector does not work, such as the center of the cheeks and the forehead. These points are generated using geometric measures from the points obtained initially. At this step, unnecessary points such as the central contour of the lips of the mouth are removed to prevent the creation of excessive search regions.

The second step of face coverage is the creation of the Voronoi diagram, which corresponds to a special type of decomposition of a given space, for example, a metric space, determined by the distance between the FFP. Figure 2(a) illustrates the Voronoi diagram obtained from the FFP considered in the previous steps.

Finally, the last step corresponds to the definition of the R search regions. These regions are defined by the polygons generated by the Voronoi diagram. Each region is enlarged in order to avoid distortions in the filtering step.

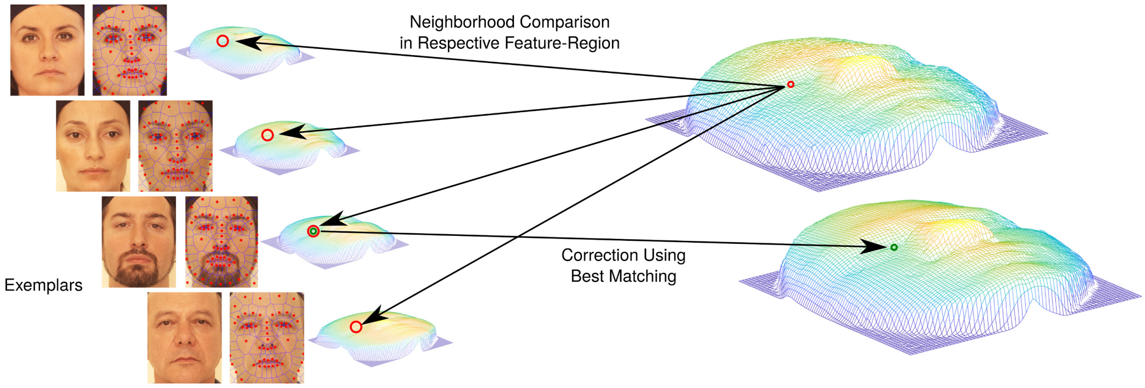


Figure 2: The pipeline of the proposed method.

To guarantee a proper sampling frequency of all exemplars and models, we perform an alignment [5] of their FFP. First, we choose one exemplar (anyone) to be the base, and we align all others according to this one. After the alignment, we scale each face with respect to the base one using the alignment affine transformation [5]. Finally, we can resample the other face by the same frequency sampling of the base one. The Figure 2(b) shows the alignment of the faces.

3.2 Normalized Depth-Neighborhood Comparison

For each point p in the mesh, we create its neighborhood as a $k \times k$ (k is odd) matrix in which the central value is the depth of p and the other values are the depth of the respective neighbor. The comparison between two neighborhoods is performed using Euclidean distance. However, even when we are comparing two geometrically similar neighborhoods, it is possible to have a big difference. To obtain a proper measure of geometric similarity, we normalize all neighborhood as follows:

$$\Phi(x,y) = \frac{\psi(x,y) - \mu}{\sigma} \tag{1}$$

where μ is its mean and σ is its variance.

The purpose of the normalization step is handling the models obtained with different setups, as distance and scale making the approach more robust. In this way, μ creates a translational invariance and σ creates an invariance to scale (both in the depth).

We denote M_p by the point in the model in the position p and E^e_q by the point q in the exemplary e. The neighborhood of each point M_p is compared to the neighborhoods at the exemplars, and the respective depth is replaced by the depth of the respective best matching. The search for the most similar neighborhood on exemplars is performed by a Nearest Neighbor method, implemented using a Kd-Tree, in conjunction with PCA for dimensionality reduction [3]. The neighborhood comparison metric of a point M_p and a point E^e_q is given by:

$$dist\left(M_p, E^e_q\right) = \sum_{i=1}^{k}\sum_{j=1}^{k}\left(M_p(i,j) - \left(E^e_q(i,j)\right)\right)^2 \tag{2}$$

4 Results

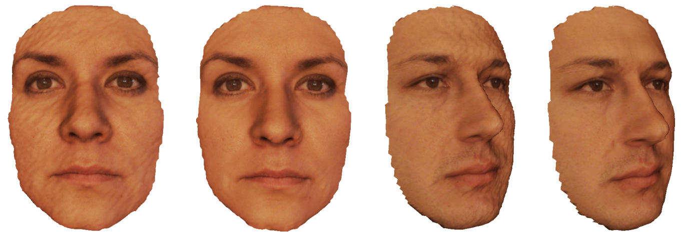
The initials results were generated using a set of exemplars based on the Bosphorus Database[6]. This database is intended for research on 3D and 2D human face processing tasks including expression recognition, facial action unit detection, facial action unit intensity estimation, face recognition under adverse conditions, etc.

In order to achieve the results, we performed the experiments applying noise to the set of the different models. The Figure 3 shows the visual results obtained. The (a) and (c) are the noised meshes and (b) and (d) are the filtered images.

5 Conclusions and Future Work

We presented a content-aware model to filtering of the face meshes based on exemplars. The initial results seemed robust and promising. A direction for future work is to define a validation metric for the results. It could be based on the measure of the variance of each point. Indeed, the proposed method reduces the overall variance. In addition, it is possible to use the symmetry of faces for creating this quality measure. Furthermore, we can define a metric by a comparison of the filtered model with a facial template deformed by the FFPs.

Finally, our filtering approach is based on a division of the model in regions in which all points have an intrinsic geometric similarity. We presented how to define these regions for the specific case of faces with the use of FFPs. For other types of models, it is necessary to use a feature detector that satisfies the three conditions presented in Section 3.1. A future work direction is to define general descriptors that can be used for general purpose filtering.



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

Figure 3: This Figure shows some visual results obtained with the proposed method. (a) and (c) are the noised meshes and (b) and (d) are the filtered images

References

[1] Mario Botsch, Mark Pauly, Leif Kobbelt, Pierre Alliez, Bruno Lévy, Stephan Bischoff, and Christian Rössl. Geometric modeling based on polygonal meshes. In *ACM SIGGRAPH 2007 Courses*, SIGGRAPH '07, New York, NY, USA, 2007. ACM.

[2] Hojin Cho, Hyunjoon Lee, Henry Kang, and Seungyong Lee. Bilateral texture filtering. *ACM Trans. Graph.*, 33(4):128:1–128:8, 2014.

[3] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. Image analogies. In *SIGGRAPH*, 2001.

[4] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1867–1874, June 2014.

[5] A. Myronenko and X. Song. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, Dec 2010.

[6] Arman Savran and Blent Sankur. Non-rigid registration based model-free 3d facial expression recognition. *Comput. Vis. Image Underst.*, 162(C):146–165, September 2017. ISSN 1077-3142.

[7] H. Yagou, Y. Ohtake, and A. Belyaev. Mesh smoothing via mean and median filtering applied to face normals. In *Geometric Modeling and Processing. Theory and Applications. GMP 2002. Proceedings*, pages 124–131, July 2002.

[8] Y. Zheng, H. Fu, O. K. Au, and C. Tai. Bilateral normal filtering for mesh denoising. *IEEE Transactions on Visualization and Computer Graphics*, 17(10):1521–1530, Oct 2011.