Easy Rigging of Face by Automatic Registration and Transfer of Skinning Parameters

Ludovic Dutreve† Alexandre Meyer† Veronica Orvalho‡ Saïda Bouakaz†

†Université Lyon 1,
LIRIS, CNRS, UMR5205, France.

‡Instituto de Telecomunicações,
Faculdade de Cincias da Universidade do Porto, Portugal.

Abstract. Preparing a facial mesh to be animated requires a laborious manual rigging process. The rig specifies how the input animation data deforms the surface and allows artists to manipulate a character. We present a method that automatically rigs a facial mesh based on Radial Basis Functions (*RBF*) and linear blend skinning approach. Our approach transfers the skinning parameters (feature points and their envelopes, ie. point-vertex weights), of a reference facial mesh (source) - already rigged - to the chosen facial mesh (target) by computing an automatic registration between the two meshes. There is no need to manually mark the correspondence between the source and target mesh. As a result, inexperienced artists can automatically rig facial meshes and start right away animating their 3D characters, driven for instance by motion capture data.

1 Introduction and Related Work

Modeling 3D faces is becoming more and more automatic and common with systems based on photos [1] or on user-friendly interactions [2]. Consequently, many applications in the area of games or virtual reality offer to novices the capacity to generate or customize a 3D facial avatar. Indeed, avatar with better ability to depict his owner in the virtual community will provide a better immersion. However, setup a facial mesh in order to animate it requires fastidious manual rigging to specify how the input animation data deforms the surface. For instance, the first stage of a rigging process, which is to manually place some feature points (FP) on the face, takes around 2 minutes [3]. Thus, in the same way that [4] does for the body, we aspire to a system that automates the face rigging task, to make animations more accessible for children, educators, researchers, and other non-expert animators.

For the real-time facial mesh deformation, we have chosen to focus on the traditional Linear Blend Skinning[1] (LBS) [5, 6]. Indeed, with Blendshape [7] which consists on morph between key meshes, they remain the most popular methods used in practice in real-time 3D engines. Although both methods provide realistic animations, it seemed more tractable for our purpose to deal with skinning parameters than key shapes, and more pertinent to stay compatible with recent mesh deformation based on skinning [8] for a potential extension. Skinning binds

[1] also known as skinning or enveloping or Skeletal Subspace Deformation(SSD)

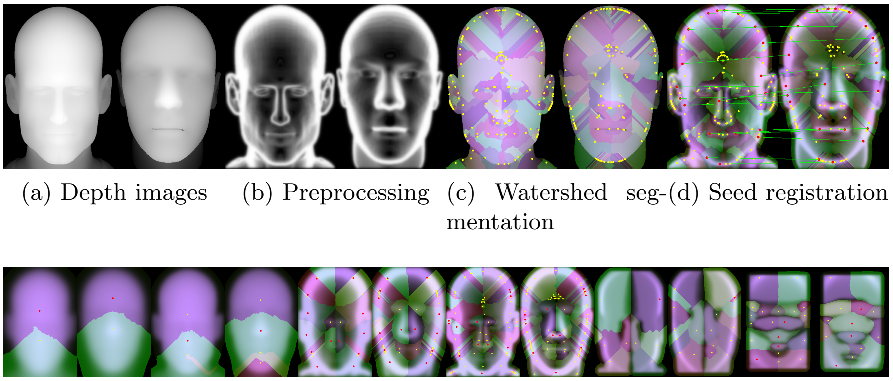
controllers to vertices in order to deform the mesh according to the controllers deformations. Usually, these controllers are bones or skeleton for the body and feature points (FP) or landmarks for the face[2]. In most case, a vertex may be influenced by several FP by using a convex combination of weights.

In this paper, we propose a method which takes as input a mesh of face (target) and provides as output skinning parameters, *i.e.* a FP set and the associated point-vertex influences (weights). These skinning parameters are transferred without human intervention from a reference mesh (source) already rigged. Comparing to *procedural* approaches mostly used in automatic body rigging approaches like [4], transfer-based approaches has the advantage, in the specific case of face, to reuse fine manual work designed specially to tackle subtle facial animation. Moreover, previous transfer approach for face [10] has validated the transfer based concept. Comparing to [10], our main contribution is the fully automatic registration which provides a landmark correspondence between the two meshes (See Section 2). Once the 3D target mesh is rigged with our method, it may be directly animated by any FP-based animations coming for instance from a motion capture system based on webcam as illustrated in our results (See Figure 4). Or, it may be used by any 3D engines and major 3D Softwares.

An important aspect of the automatic transfer of facial animation parameters is the registration which tries to find a transformation that optimally fits points from the source surface to the target surface. Registration of range images of the same object [11] is often tackled by variant of the Iterative Closest Points algorithm [12–14]. Although we represent the frontal face view with a depth image which is similar to a range image, our problem has to be classified in the area of registering two instances of objects belonging to the same family: variation of faces morphologies. Our case is also more general than the class of non-rigid registrations problems aiming at fitting a template [15, 7, 16, 14] model to scanned or motion capture surface. These methods are often based on variational approaches minimizing surfaces differences from morphable shape model computed from a database.They provide good results when faces are close which is not necessary our case. And, for the sake of generality, we preferred working on an approach without *a priori* on the surfaces which allows to register two cartoon faces as illustrated in Figure 2d and 2e.

Overview A reference facial mesh is previously rigged for skinning animation. Our method automatically registers the reference mesh (source) to the facial mesh to rig (target) in order to transfer the skinning parameters (FP and point-vertex influences) without human intervention. The result is a rigged facial mesh ready to be animated by skinning. The transfer is done in four phases. Firstly, the method we present addresses the correspondence issue using a new approach registering the result of a sequence of watershed segmentations described in Section 2. Then, these landmarks are used in a radial basis functions (*RBF*) learning process to transfer the FP from the source to the target and in the third phase (Section 3) to computes a dense registration. And finally, the point-vertex influences are transferred.

[2] as the ones defined by the Facial Definition Parameters of the MPEG4 standard [9].



(a) Depth images (b) Preprocessing (c) Watershed segmentation (d) Seed registration

(e) (f) (g) (h) (i) (j)

Fig. 1: Top row: one step of our landmark based registration. Bottom row: a sequence of the watershed segmentation/registration process with different preprocessing. (e) and (f): Gaussian filters with different radius. (g) and (h): a Laplacian filter followed by a Gaussian filter with different radius. (i) and (j): vertical and horizontal anti-symmetric filters.

2 Surface Correspondence

The first issue of our transfer approach is the computation of the landmark correspondence on both meshes, which may be tricky if the anthropometric lengths are slightly different. To find this landmark correspondence, our approach runs a sequence of preprocessing, watershed segmentation and region-based registration on the frontal depth images of the faces.

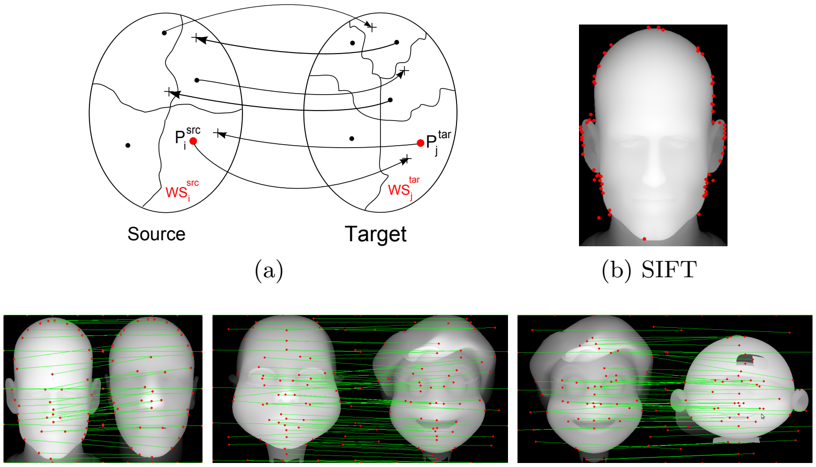
We have chosen to register the two faces using a depth images representation for several reasons. We start from the observation that a frontal part of a face is 2.5D, so the depth image representation should keep the important information for registration. Depth image allows to be easily independent on the mesh resolution which is essential for the usefulness of the method. A low polygon facial model may be registered with a high polygon model. Comparing to complex remeshing, computing the depth map representation of a face can be done easily using an orthogonal projection (as for instance the Z-buffer) and the reprojection of a 2D pixel position on the mesh allows to return in 3D. And finally, working on regular 2D grids (images) is more convenient and more efficient in computation time than working directly with the mesh.

One Step of Watershed Segmentation At each step, to compute new landmark correspondences, we perform a watershed segmentation where the regions are registered. The depth image is segmented by the watershed algorithm [17] interpreting it as a topographic surface and gradually immerses it in a water container [17]. After the segmentation, the surface will be divided into its catchment basins meaning every pixels is assigned to a region as illustrated in Figure 1c. The local minimum point (seed) of each basin is the representant of the basin. Since the segmentations are computed using the global image information, our method can localize the landmark correspondence more accurately than local detection/grouping methods [18]. For instance, the SIFT descriptor algorithm selects landmarks only on the border which is less precise than our method as illustrated in Figure 2b and 2c. The assumption of our method is that objects of a same family will provide mostly similar segmentation which are more easy to register than directly the meshes.

At step 0, to initialize the sequence, we set landmarks around the respective bounding box of each face. In this paragraph, we assume that k steps of the sequence of watershed segmentation/registration have already been done and have provided a set of landmarks. These landmarks are used as learning points for a RBF interpolation as described in [19] defining the transformation $RBF_k^{src->tar}$, just noted RBF, which transforms a point of the source image to the target image. By inverting the landmarks, we compute $RBF_k^{tar->src}$ noted RBF^{-1}. We consider the next step of the sequence, the $k+1^{th}$ watershed segmentation/registration. The watershed segmentation computes on the two depth images, two sets of regions with their associated seeds (See Figure 1c). A seed of the source segmentation will be selected as landmark if it can be associated with a seed of the target segmentation. The correspondence criteria between two watershed is described in Figure 2a. Two watersheds WS_i^{src} and WS_j^{tar} match if the seed P_i^{src} (resp. P_i^{tar}) falls into the watershed WS_j^{tar} (resp. WS_i^{src}) after being transformed by the RBF (resp. RBF^{-1}) defined by the previous steps. *i.e.* if $RBF(P_i^{src}) \in WS_j^{tar}$ and $RBF^{-1}(P_j^{tar}) \in WS_i^{src}$ we add P_i^{src} and P_j^{tar} to the set of landmarks defining the correspondence. For instance, in Figure 2a, the bottom right watershed regions match, top ones not. At the end of this step, we recompute the RBF with the added landmarks for the next step. This criteria is fast to compute and has the advantage to quickly registers similar regions. According to our tests, it is enough efficient to not considering an other criteria more complicated to compute.

Sequence of Segmentation/Registration as a Descriptor Problem The watershed segmentation applied on depth image combined with our registration criteria has the advantage to provide a global approach which can be easily iterated with different preprocessing. Indeed, in this Section we propose to take advantage of various descriptors to catch different regions of a face, for instance regions of high curvature or of high symmetry, *etc.*. Nevertheless, this general framework may be refined by any kind of surface descriptors and is related to the classical issue of choosing pertinent descriptors for image or surface registration [20, 11]. The choose of descriptors order have to be from coarse to fine.

The first preprocessing we apply is a Gaussian filter on the depth images to remove the issue of small regions during the segmentation. This will register the global appearance (local maxima) of the face, mainly the nose and often the forehead (See Figure 1e). In second, we apply a Laplacian filter followed by a Gaussian filter with different radius. It generally registers edges like borders of the face or of the nose (See Figure 1g). Finally, since a face included many symmetric regions, we compute two descriptors of symmetry, horizontal and vertical, as illustrated in Figure 1i. In practice, results of Figure 3 have been computed by these preprocessing also illustrated in the bottom row of Figure 1. But, reader may define other interesting descriptors for different class of shape.



(a) (b) SIFT

(c) (d) (e)

Fig. 2: In (a), the criteria to match two segmented regions. In (b), the SIFT descriptors algorithm [21] selects landmarks only on the border. In (c), (d) and (e), our sequence of segmentations/registrations provides a set of 2D pair of landmarks. (d) and (e) illustrate the registration on cartoon faces.

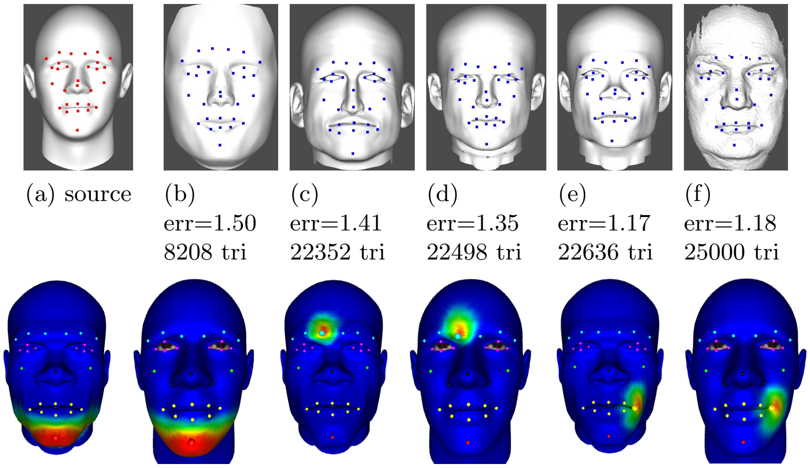
3 Transfer of Feature Points and Point-Vertex Influences

The sequence of watershed segmentations/registrations previously described provides as output two sets of landmarks on the mesh defining a correspondence as illustrated in the bottom row of Figure 2. We use these landmarks to define a RBF interpolation $RBF_{final}^{src->tar}$ to transfer each FP position of the source face to the target face as illustrated in the top row of Figure 3.

Rigging the target mesh with common skinning techniques requires to attach each vertex to one or more FP with influence values. As for the FP, we perform this task by transferring this information from the source mesh by defining a dense registration in a similar way than in [22, 10]. To compute this dense registration, each vertex V of the target mesh is transformed by the RBF interpolation: $V' = RBF_{final}^{tar->src}(V)$. Since V' does not fall necessary exactly on the source mesh, it is projected[3] on the source mesh to falls into a point P of a

[3] We apply a spherical projection but our tests with a cylindrical projection do not lead to significant differences.

triangle T_{src} which defines our dense registration. Notice that P is a 3D position on the triangle and is not necessary a vertex. We compute the FP influences of V by interpolating the ones of the 3 vertices of the triangle T_{src}. We have tested the Inverse Distance Weighting interpolation and the barycentric interpolation which both provide similar and enough accurate results in our case. The bottom row of the Figure 3 shows an example of influences transfer for some FP.



(a) source

(b) err=1.50 8208 tri

(c) err=1.41 22352 tri

(d) err=1.35 22498 tri

(e) err=1.17 22636 tri

(f) err=1.18 25000 tri

Fig. 3: Top row: examples of transfer of the set of FP . Bottom row: examples of transfer of the point-vertex influences from the reference face (the color scale of the weights starts from blue (0.) to red (1.)).

4 Results and Conclusion

Figure 3 shows a sample of our facial mesh database used for our experiments which includes 15 face variations with a large scale of characteristics: 3D meshes count from about thousand triangles to several tens of thousand triangles; bounding box sizes are also quite variable; some meshes include only the face, others include the whole head; and some models are computer generated, others are scanned faces (like the last on the right). The only constraint we ask to the models is that the face has to be manually set to look in the Z axis direction to be able to compute the depth image of the frontal view. We also tested our registration on cartoon faces as illustrated in Figure 2d and 2e. To measure the quality of the landmark correspondence, we have manually defined twenty FP on representative positions for animation for each 3D mesh (See Figure 3a). We define an error criteria by summing the distance between each FP of the transferred rig and the manually set ones normalized by the diagonal of the bounding box to be able to compare results on different mesh size. Notice that the result of the automatic FP transfer may be manually edited if the user feels the need, without changing the rest of the method. Our results in Figure 3, 4 and on the provided video are not manually adjusted.

Computing the depth image representation of the mesh is instantaneous using the GPU/Z-buffer of the frontal view. The landmark correspondence determination is computationally dependant on the depth image resolution and on the number of steps in the sequence of segmentations/registrations. In our experiments, we used depth images of 512×512 with the sequence of preprocessings/segmentations/registrations described in Figure 1 and in Section 2. It takes about 20s on a Athlon X2 3800+ with 2GB of RAM. The dense registration is computationally dependent on the number of triangles of the two meshes and takes for instance about 30s for a mesh with 12000 triangles. Once the dense registration is done, the time of skinning parameters transfer is immediate. And finally to illustrate our results, Figure 4 and the provided video show an animation applied on two faces, the first was rigged manually while the second was automatically rigged by our method in a minute without human intervention. The animation is provided on the fly by a motion capture system using a simple webcam and the Lucas-Kanade [23] marker tracker. Notice that any other motion capture system may be used and that the markers on the face may be different than the FPusing retargeting adaptation [24, **?**].

These results prove the viability of such an automatic transfer approach based on depth image representation of the mesh which makes the implementation easier, lets use efficiently several descriptors with an approach based on segmentation, and allows to easily deal with mesh of different resolutions. Nevertheless, fully automatic transfer of rigging from a generic reference human face to a large range of faces like cartoons or animals is still delicate because of the landmark correspondence issue. A solution would be to rig several types of reference faces (human, cartoon, animal, *etc.*) as preprocess and rig a new face by transfer of parameters from the nearest reference face.

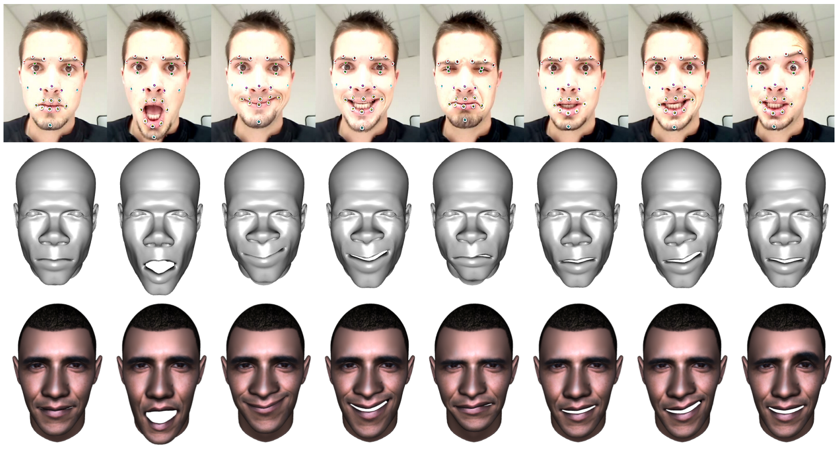


Fig. 4: Top: motion capture animation. Middle: reference model previously rigged. Bottom: the same animation applied on a face automatically rigged by our method, rigging is transferred from the reference without intervention.

References

1. Geoegios Stylianou and Andreas Lanitis. Image based 3d face reconstruction: a survey. *International Journal of Image and Graphics*, 2009.
2. Nikolaos Ersotelos and Feng Dong. Building highly realistic facial modeling and animation: a survey. *The Visual Computer*, 24(1):13–30, 2007.
3. Softimage. Pick the landmarks with face robot. *http://facerobot.wiki.softimage.com/index.php/Stage_3:_Pick_the_Landmarks*, 2009.
4. Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. *ACM Trans. Graph.*, 26(3):72, 2007.
5. N. Magnenat-Thalmann, R. Laperrire, , and D. Thalmann. Joint-dependent local deformations for hand animation and object grasping. In *In Proceedings on Graphics interface*, 1988.
6. J.P. Lewis, M. Cordner, and N. Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *ACM SIGGRAPH'00*.
7. Z. Deng and J. Y. Noh. *Computer Facial Animation: A Survey*, chapter 1. Springer, Ulrich Neumann, Zhigang Deng (eds.), 2007.
8. W. Feng, B. Kim, and Y. Yu. Real-time data driven deformation using kernel canonical correlation analysis. In *ACM SIGGRAPH*, 2008.
9. J.L. Zhong. Flexible face animation using mpeg-4/snhc parameter streams. In *IEEE International Conference on Image Processing*, pages II: 924–928, 1998.
10. V. Costa Orvalho, E. Zacur, and A. Susin. Transferring the rig and animations from a character to different face models. *Computer Graphics Forum*, 2008.
11. J. Salvi, C. Matabosch, D. Fofi, and J. Forest. A review of recent range image registration methods with accuracy evaluation. *Image Vision Comput.*, 2007.
12. Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.
13. Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Third Int. Conf. on 3D Digital Imaging and Modeling (3DIM)*, June 2001.
14. P. Eisert D. Schneider. Algorithms for automatic and robust registration of 3d head scans. *Journal of Virtual Reality and Broadcasting 2009*, 2009.
15. Brett Allen, Brian Curless, and Zoran Popović. The space of human body shapes: reconstruction and parameterization from range scans. In *ACM SIGGRAPH*, 2003.
16. T. Weise, H. Li, L. Van Gool, and M. Pauly. Face/off: live facial puppetry. In *Symposium on Computer Animation*, 2009.
17. Alan P. Mangan and Ross T. Whitaker. Partitioning 3d surface meshes using watershed segmentation. *IEEE Trans. on Visu. and Computer Graphics*, 1999.
18. H. Arora Varsha Hedau and N. Ahuja. Matching images under unstable segmentation. In *IEEE Computer Vision and Pattern Recognition*, 2008.
19. L. Dutreve, A. Meyer, and S. Bouakaz. Feature Points Based Facial Animation Retargeting. In *ACM Symp. on Virtual Reality Software and Technology*, 2008.
20. Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 2003.
21. David G. Lowe. Distinctive image features from scale-invariant keypoints,. *International Journal of Computer Vision*, 2004.
22. J. Y. Noh and U. Neumann. Expression cloning. In *ACM SIGGRAPH*, 2001.
23. Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. *Int. Conference on Artificial Intelligence*, 1981.
24. Frederic Pighin and J. P. Lewis. Facial motion retargeting. In *ACM SIGGRAPH 2006 Courses*, page 2, New York, NY, USA, 2006. ACM.