COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2011; **22**:169–176
Published online 12 April 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cav.395

SPECIAL ISSUE PAPER

Easy acquisition and real-time animation of facial wrinkles

Ludovic Dutreve, Alexandre Meyer* and Sada Bouakaz

LIRIS – Université de Lyon, Bâtiment Nautibus (710), 43, Boulevard du 11 Novembre 1918, 69622 Villeurbanne Cedex, France

ABSTRACT

Facial animation details like wrinkles or bulges are very useful for the analysis and the interpretation of facial emotions and expressions. However, outfitting a virtual face with expression details for real-time applications is a difficult task. In this paper, we propose a mono-camera acquisition technique of facial animation details and a technique which add a wrinkle map layer (fine-scale animation) to a skinning layer (large-scale animation) for real-time rendering of a virtual 3D face. The acquisition is based on ratio image computed from two pictures of a same face, with and without expression. The real-time dynamic wrinkles technique is based on a small set of reference poses. These two methods offer an easy and low-cost way to capture facial animation details and use it for real-time facial animation. Copyright © 2011 John Wiley & Sons, Ltd.

KEYWORDS
facial animation; wrinkle acquisition; fine-scale animation; RBF interpolation

***Correspondence**
Alexandre Meyer, LIRIS – Université de Lyon, Bâtiment Nautibus (710), 43, Boulevard du 11 Novembre 1918, 69622 Villeurbanne Cedex, France.
E-mail: alexandre.meyer@liris.cnrs.fr

1. INTRODUCTION

Facial animation details like wrinkles or bulges are very useful for the analysis and the interpretation of facial emotions and expressions [1]. However, outfitting a virtual face with expression details for real-time applications is a difficult task. The main reason is the difficulty to capture or synthesise wrinkle phenomena which human used to see every day, especially in a real-time context.

Since many works have been proposed for large-scale animations and deformations [2,3], only a few of them deal with real-time animation of small-scale details on face. We denote small-scale details in an *animation* context, i.e. wrinkles and bulges appearing while muscles contractions, instead of the micro-structures of the skin independent to facial expressions. An interesting approach was proposed by Oat [4], which consists on blending wrinkle maps to render animated details on a human face. This technique provides good results at an interactive time using common bump-mapping technique for the rendering. However, it requires manual tuning of blending coefficients, leading to manual efforts for each new animation, and requires wrinkle maps which may be difficult to create.

Capturing these facial animation details is still a challenging issue. On one hand, current capture methods focus on capturing the whole 3D geometry of a face with heavy active system based on lasers, structured lights or gradient-based illumination, with multiple cameras [5–7]. Even if recent researches propose passive systems [8,9], stereo aspect is still the minimum, with often six or more high resolution cameras. On the other hand, many applications want to apply captured motions to other faces than the one filmed. Thus, they do not need 3D high resolution meshes of the actor from which desired features are hard to extract and to transfer.

In this paper, we propose two techniques to both capture details and apply them on a real-time facial animation. To avoid complex motion capture system, we propose an easy and low-cost technique to capture facial details. On several frontal photos of different facial expressions, we capture corresponding normal maps with a few manual effort of landmarking on pictures. By considering only frontal single-camera views, we address the problem of normal reconstruction using the shading information in an inverse problem, where classical facial capture approaches use stereo. Our dynamic wrinkle technique consists on adding a wrinkle map layer to a skinning layer. The skinning layer is a common bone-based large-scale animation technique on which we add the fine-scale details thanks to a local-area-based wrinkle map layer. In order to apply wrinkle map, we use a small set of reference poses. A reference pose is a pair of a large-scale deformation (a skeleton pose) and its associated small-scale details (a wrinkle map). During the animation, the current skeleton pose is compared with the reference poses and wrinkle maps coefficients are automatically computed with the use a nonlinear function. Notice that comparison is done at a bone level resulting in local blends and independence between areas of the face.

2. RELATED WORK

Since our approach is based on single camera for the normal maps acquisition, we focus on single image shape reconstruction, and specially on inverse shading approaches which have to be classified in the class of Shape from Shading (SfS) problems. The SfS problem has been widely studied in the computer vision area [10]. It is known as difficult because of its ill-posedness [11]. Consequently, few approaches have been tried on real photographs [12]. By considering the normal-map reconstruction instead of the 3D shape we relax a bit the difficulty of the problem [13]. Moreover, normal maps are well suited to real-time rendering and to our wrinkles-synthesis-from-example method. Wu *et al.* [13] propose an interesting interactive normal reconstruction from a single image. By aiming at reconstructing only local features such as wrinkles, we can limit the phase of user interaction which is mainly dedicated to give global information. Considering photo of face gives an a priori on the shape form which helps to solve the ill-posedness aspect and allows to reconstruct the large scale aspect of a face [14]. To our knowledge, state-of-the-art on SfS (understanding mono-camera) does not show convincing results on skin wrinkles acquisition. Thus, they are often procedurally synthesised [15].

While some techniques were proposed to generate wrinkles on arbitrary surface [16,17], few approaches focused on real-time facial applications. Larboulette and Cani [18] proposed a technique to simulate dynamic wrinkles. The user defines wrinkling areas by drawing a perpendicular segment of wrinkles (wrinkling area), following by the choice of a 2D discrete control curve (wrinkle template). The control curve conserves its length while the mesh deformation, generating amplitude variations. Wrinkles are obtained by mesh subdivision and displacement along the wrinkle curve. Many methods need a high resolution mesh or a *on-the-fly* mesh subdivision scheme to generate or animate wrinkles [19,20]. Due to real-time constraints and the resource sharing, these techniques may be difficult to use in an efficient way. GPU computing allows to render efficiently fine details using bump maps for interactive rendering. Oat presented in Ref. [4] a GPU technique to easily blend wrinkle maps applied to a mesh. Maps are subdivided in regions, for each region, coefficients allow to blend the wrinkle maps. This technique requires few computational and storage costs, three normal maps are used for a neutral, a stretched and a compressed expressions. Furthermore it is easily implemented and added to an existing animation framework. The main drawback of this method is that it requires manual tuning of the wrinkle maps coefficients for each region. Similarly, de Melo and Gratch [21] synchronise a normal map with a pseudo-muscular model to avoid manual coefficient tuning. Normal maps are obtained by a manual image manipulation step. They directly use luminance information on a single photography and manually fit details to the character's texture. Our details extraction approach is more formal and requires less manual intervention. Our animated details method aims at providing a real-time dynamic wrinkling system for skinned face which generates automatically wrinkle maps coefficients and does not require a mask map.

3. ACQUISITION OF THE SKIN DETAILS NORMAL MAPS

In this section, we detail how we capture the skin details normal maps of several facial expressions. Since we do not need to capture the entire geometry of the face, our approach uses the illumination variation between two pictures of a same face with different expressions. Let $\mathcal{I}$ be the image of the neutral pose, and $\mathcal{I}'$ the image of the expression pose where the details we want to extract are. We seek to compute the normal map which will be applied during the real-time facial animation to produce details.

3.1. Normal Map Generation

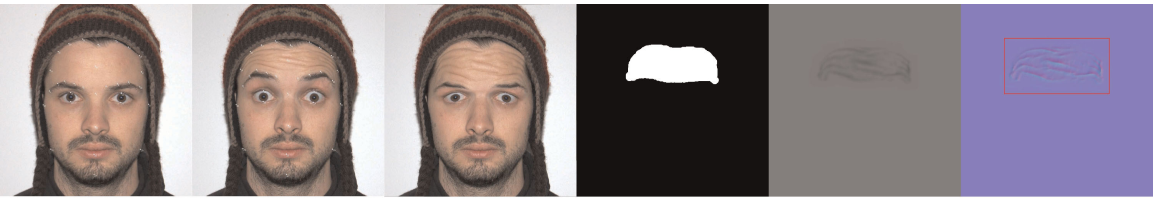
The first step of our method consists of computing the ratio image to extract light variation [22]. In order to work on pixel at correct positions, we start to deform $\mathcal{I}'$ into $\mathcal{I}'_{\mathrm{d}}$ to fit $\mathcal{I}$. This allows us to obtain a mapping between each pixel, and so, to know the difference of illumination at each point of the face surface. The deformed image $\mathcal{I}'_{\mathrm{d}}$ is obtained using a Radial Basis Functions (RBF) scattered data interpolation [23] between two sets of landmarks manually picked on the two images $\mathcal{I}$ and $\mathcal{I}'$. The ratio image $\mathcal{R}$ is computed by dividing $\mathcal{I}$ to $\mathcal{I}'_{\mathrm{d}}$:

$$\mathcal{R} = \frac{\mathcal{I}'_{\mathrm{d}}}{\mathcal{I}} \tag{1}$$

Then, we perform a blurring filter to $\mathcal{R}$ to reduce acquisition noise. To avoid artefacts due to eventual registration errors, we give the possibility to define regions of interest with a simple painting tool. Furthermore, these masks help to avoid defining a large number of landmarks to obtain a perfect mapping, even in the areas that the user does not need to extract features. Figure 1 shows an example of pictures of different expressions with the associated ratio image.

Let C_{d} the intrinsic colour of the skin, C_{l} the colour of the light, $\vec{N} = (N_x, N_y, N_z)^t$ (resp. $\vec{N}' = (N'_x, N'_y, N'_z)^t$) the normal at the surface of the skin of the neutral image (resp. the interpolated expression image[1]), and $\vec{L}$ the

[1] Notice that we work into the tangent coordinate system of the neutral image.



(a) neutral expression (b) expression with details (c) interpolated expression (d) mask (e) ratio image (f) normal map

Figure 1. Ratio image computation using two pictures of a same face with two expressions and a mask to specify where details we want to capture are. The last image is the normal map obtained from the ratio image and its gradients.

light-direction vector. If we assume that skin is a diffuse surface, under the Lambertian model, $I = C_d \times C_l \times \vec{N} \times \vec{L}$ and $I'_d = C_d \times C_l \times \vec{N}' \times \vec{L}$. We will retrieve $\vec{N}'$ in two step: the normal tilt $\vec{N}_z$ and the normal orientation $(\vec{N}_x, \vec{N}_y)$. Equation (1) gives:

$$\mathcal{R} = \frac{\mathcal{I}'_d}{\mathcal{I}} = \frac{C_d \times C_l \times \vec{N}' \times \vec{L}}{C_d \times C_l \times \vec{N} \times \vec{L}} = \frac{\vec{N}' \times \vec{L}}{\vec{N} \times \vec{L}} \qquad (2)$$

Using the flash light of the camera, we can approximate the light direction to $\vec{L} = (0, 0, -1)$, given $\mathcal{R} = N'_z / N_z$. We make the approximation that regions of interest are locally plane at the neutral pose. By default, we assume that these planes are parallel to the camera plane, resulting in $N_z = 1$, but the user can define different values for N_z if considered necessary, to refine the result.

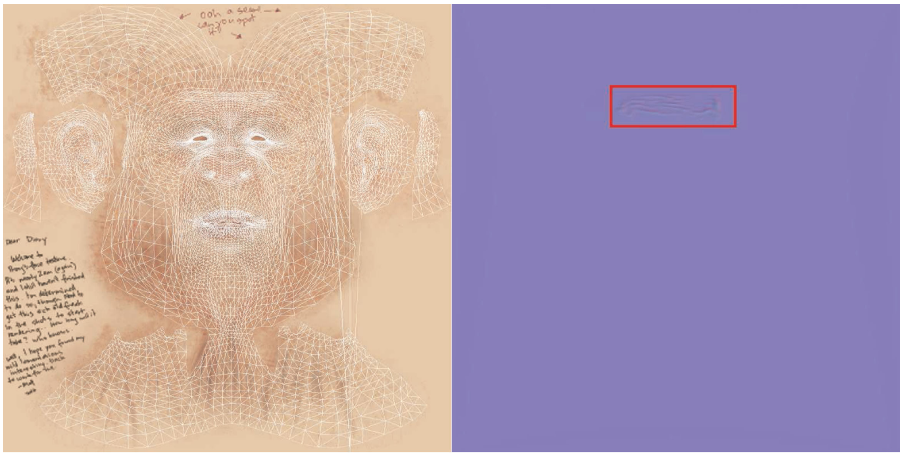
Equation (2) shows that the ratio image intensity is dependant to the orientation of the surface. More $\mathcal{R}$ is dark, more N'_z is small and more the wrinkles are deep. Notice that this use of the ratio image avoids the painful problem of finding the intrinsic colour from a photo. We should now compute the normal orientation $(\vec{N}_x, \vec{N}_y)$ for a given pixel of the wrinkles at each pixel. For this, N'_z is not enough. We assume that the gradient of the image ratio is a good approximation of the normal orientation, which is mainly similar to the assumption presented in Ref. [24,25]. Let G_x and G_y be the 2D normalised gradient of $\mathcal{R}$ for the two axes $\vec{x}$ and $\vec{y}$. We want to find N'_x and N'_y such that $N'_x 2 + N'_y 2 + N'_z 2 = 1$, and $N'_x = \alpha G_x$ and $N'_y = \alpha G_y$ with α a normalisation coefficient. We obtain the equations:

$$\begin{aligned} N'_x &= G_x \times \sqrt{1 - N'_z 2} \\ N'_y &= G_y \times \sqrt{1 - N'_z 2} \end{aligned} \qquad (3)$$

We then convert $\vec{N}'$ into a normal map norm. Figure 1 illustrates a normal map obtained with a ratio image.

3.2. 3D Character Adaptation

The last step we perform for details acquisition is the interpolation between the capture space to character texture space. It aims at ranging the computed normal map to the character. We use the same method as for the interpolation between the neutral and the expression images, two sets of landmarks are manually defined on both normal map and character's texture map, and a RBF-based interpolation deformed the first to the second (see Figure 2).



(a) texture map (b) new normal map

Figure 2. Interpolation of the extracted normal map (Figure 1f) to fit the 3D character texture space (a).

DOI: 10.1002/cav

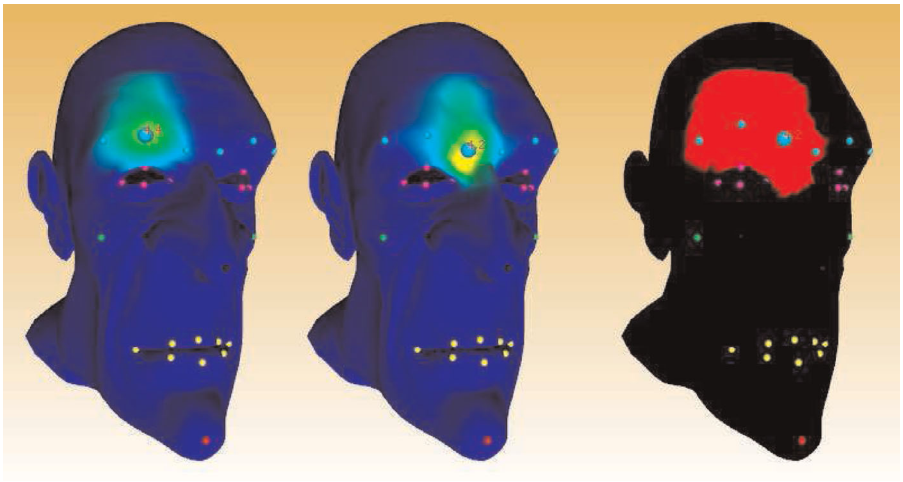


Figure 3. The two left images show the skinning influences of two right eyebrow bones. The third image shows the influence of a reference pose for each vertex attached to these bones.

4. REAL-TIME FINE DETAILS ANIMATION

Our goal is to add a wrinkle map layer (fine-scale animation layer) to a bone-based facial animation (large-scale animation). We use the wrinkles data captured at the acquisition step to generate on the fly visual dynamic fine details such as wrinkles or bulges on the face surface. In this section we explain how we use a small set of reference poses in real-time and at arbitrary poses.

4.1. Reference Poses

A reference pose is a pair of a large-scale deformation (a skeleton pose) and its associated small-scale details (stored in the form of a normal map we call wrinkle map). To create a reference pose, we deform the facial skeleton to obtain the expression on which we want the wrinkles appear. This expression should be strongly pronounced to provide better results. Then, we associate to the pose a wrinkle map extracted in the acquisition step. Since influences work with bones or group of bones, it is possible to define a pose where wrinkles appear in various areas of the face. Having details in different areas will not cause that all of these details appear at the same time at an arbitrary frame. For example, details around the mouth would appear independently with forehead details, even if they are described in a same reference pose.

4.2. Pose Evaluation

While runtime, the first step is to compute the influence of each reference pose on each bone. This consists in finding how the bone position at an arbitrary frame looks like its position at the reference poses.[2] Computing these influences at the bone level instead of a full pose level allows to determine regions of interest. This offers the possibility to apply different reference poses at a same time. Resulting in the need of less reference poses (actually, only two may be sufficient for face: a stretched and a compressed expression).

We define the influence α_{jfk} of the pose Π_k for the bone P_j at the position $\vec{P}_{jf}$ for the arbitrary pose Π_f by the following equation:

$$\alpha_{jfk} = \min\left(1, \max\left(0, \frac{\vec{AB}\cdot\vec{AC}}{\|\vec{AC}\|}\right)\right) \qquad (4)$$

where $\vec{AB} = (\vec{P}_{jf} - \vec{P}_{j0})$, $\vec{AC} = (\vec{P}_{jk} - \vec{P}_{j0})$, $(\cdot)$ denotes the dot product and $||\ldots||$ denotes the Euclidean distance. α_{jfk} is the size of the segment between the orthogonal projection of $\vec{P}_{jf}$ on the segment $[\vec{P}_{jk}, \vec{P}_{j0}]$ normalised by the size of the segment $[\vec{P}_{jk}, \vec{P}_{j0}]$.

4.3. Bones Masks

Once we know the reference poses influences for each bone, we could use them for the per-pixel lighting. The main idea is to use skinning weights to compute the influence of reference poses for each vertex, and by the interpolation done during the rasterisation phase, for each fragment (Figure 3). Since wrinkles and expressive details are greatly related to the face deformations, we can deduce that these details are related with bones transformations too. So we associate bones influence with reference poses influences. A mesh vertex v_i with a skinning weight of w_{ij}

[2] Notice that we deal with bones positions in the head root bone coordinates system, so we could assume that head rigid transformation will not cause problems while pose evaluation.

with the bone P_i is influenced by the reference pose Π_k at the arbitrary pose Π_f such as:

$$\beta_{ifk} = \sum_{i=1}^{n} \lambda_{ij} \times \theta(\alpha_{jfk}) \tag{5}$$

with

- $\lambda_{ij} = 1$ if $w_{ij} > 0$ and $\lambda_{ij} = 0$ else;
- $\theta(x)$ the modulation function described in the next section.

4.4. Modulation Function

Wrinkles do not appear on face linearly according to bones displacements. Indeed, two bones moving from the neutral position to the compressed one will create wrinkles more intensively at this end of the movement than at the beginning. For this purpose, we introduce during the synthesis phase a nonlinear function θ such as $\theta : [0, 1] \rightarrow [0, 1]$ which rules this phenomena by modifying the distance used during the normal map blending. To propose a function based on real wrinkles apparition, we run our normal map acquisition algorithm on each frame of a video sequence where a face start from a neutral pose to an expressive one. It provides for each pixel and for a given bones configuration, a vector of normal map displacement. The length of this vector indicates the intensity of the wrinkles. Since the wrinkle synthesis is done on the GPU, it would have been inefficient to store for each pixel a set of discrete values for each bones configurations. We preferred fit an analytic curve:

$$\theta(x) = \frac{\cos(\pi(1 + x)) + 1}{2} \tag{6}$$

Figure 4 shows the segment curve we capture and the curve we propose. The cosine function seems to be an interesting base if we apply to it a translation and a normalisation to check our constraints. We try to depict a smooth and progressive acceleration of winkles amplitude. We give priority to highest per cent of deformation, when wrinkles are deepest, whereas the distance between the two curves, which seems high at the begin of the deformation, produces only a weak visual impact. Notice that user can define a new function and/or capture his own data to better fit his needs if necessary.

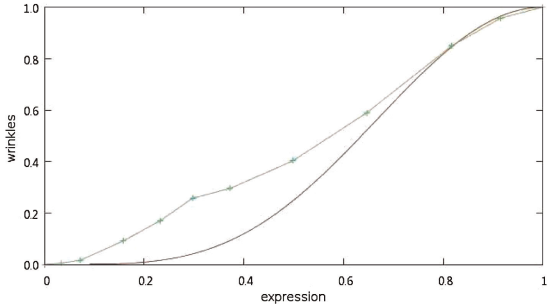


Figure 4. In green: the capture curve, in red: the function we propose for wrinkle appearance by the reference pose ratio.

4.5. Details Blending

The final step of our method is to apply wrinkle maps to our mesh using coefficients θ (β_{ifk}) computed at the previous step. Since normal maps only contain deformations associated with the expression, we should apply a finest blending to avoid loss of static details located on the neutral normal map (i.e. details such as pores, scares and other static fine details). For example, with a simple averaging, a fragment influenced by 100% of a wrinkle map will be drawn without using the neutral map, resulting in the fact that details of the neutral map will not appear. Let $\vec{n}_f$ the final normal, n^i the ρ normals provided by the wrinkle maps, their coefficients w^i and $\vec{n}$ the normal provided by the default normal map, we first compute weighted normals η^i of the wrinkle maps and then compute the final normal:

$$\eta^i = (0, 0, 1)^t + w^i \times \left(n^i - (0, 0, 1)^t\right) \tag{7}$$

$$\vec{w}_f = \begin{pmatrix} n_x + \sum_{i=1}^{o} \rho\eta_x^i \\ n_y + \sum_{i=1}^{\rho} \eta_y^i \\ n_z \times \prod_{i=1}^{\rho} \eta_z^i \end{pmatrix} \tag{8}$$

The addition of the two first coordinates makes a simple averaging between the direction of the two normals, given the desired direction. The z components are multiplied, this leads to increase details obtained from the two normals. More z is small, and more the surface will be bumpy, multiplication allows to add the neutral surface variation to the wrinkled surface.

5. RESULTS AND DISCUSSION

Our acquisition technique provides good results with a few effort and material. Despite the lighting model approximation, which does not take into account the specular reflection of the skin, we obtain very interesting visual results. Our method provides an alternative to others facial motion capture systems if the goal is to obtain only animation fine-scale details and not the full geometry of the face. Figure 5 shows an example of use of a captured normal map by our monocular acquisition technique.

Our 3D test models *Proog*[3] is about 10 000 triangles. The faces is rigged with 21 bones. Animation runs at more than 100 fps on a common personnel computer with an Athlon 64 3800+ CPU, 3Go RAM and a NVidia Geforce GTX 275 GPU. Figure 6 shows the neutral and the two reference poses used for demonstration. Figure 7 shows the six basis expressions of *Proog* with and without our dynamic wrinkles and Figure 5 shows the possibility to

[3] © Copyright 2006, Blender Foundation/NetherlandsMedia Art Institute/www.elephantsdream.org.

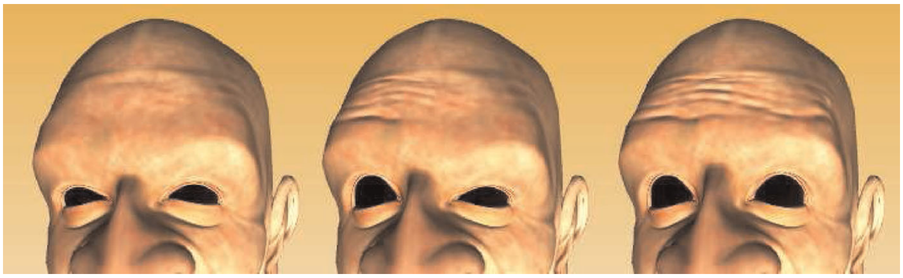


Figure 5. Example of acquired normal map applied on our target character.

apply dynamic details on different area of the face even if details are given on a same wrinkle map.

Large scale skinning-based deformation are not modified. The addition of our method to an existing implementation is easy. No additional rendering pass is required. Our choice to use skinning weights as bones masks offers many advantages. They allow us to relate large scale and small scale deformations, and so, we do not need additional mask textures. They ensure that vertices influenced by reference poses are vertices which are displaced accordingly with bones too.

Although our technique easily provides wrinkles on facial animation, several steps are important to obtain good results. First, a good rigging is primary since we directly use skinning weights as bones masks, and so, it defines how each vertex will be influenced by the different reference poses. Second, the reference poses should consist on an orthogonal set of skeleton poses as much as possible, to avoid an over fitting. Notice that blending reference poses in a same area is possible but it becomes a problem if similar bones displacements lead to different fine-details. Finally, detail maps quality greatly influences the visual results. Problems occur when capturing faces with beard where mouth details can not be obtain because of the incoherence in the ratio computation. Our approach is more suited to soft skin subjects.

6. CONCLUSION

We have presented two techniques. The first aims at capturing facial animation details from two pictures of a same face with and without expression. It requires few manual work and does not need specific material. A simple camera with its flash light was used. The second is a technique which use reference poses to generate non-linearly in real-time wrinkles and fine-details appearing with an arbitrary skinned face animation. In addition to providing interesting visual results, the requirements that we considered necessary and/or important have been met. Our dynamic animation wrinkles runs in real-time, the use of per-pixel lighting allows us to dispense with high-resolution meshes or costly subdivision techniques. Furthermore, it is based on widely used techniques such as skinning and bump mapping. Its implementation does not present technical difficulties and does not modify usual

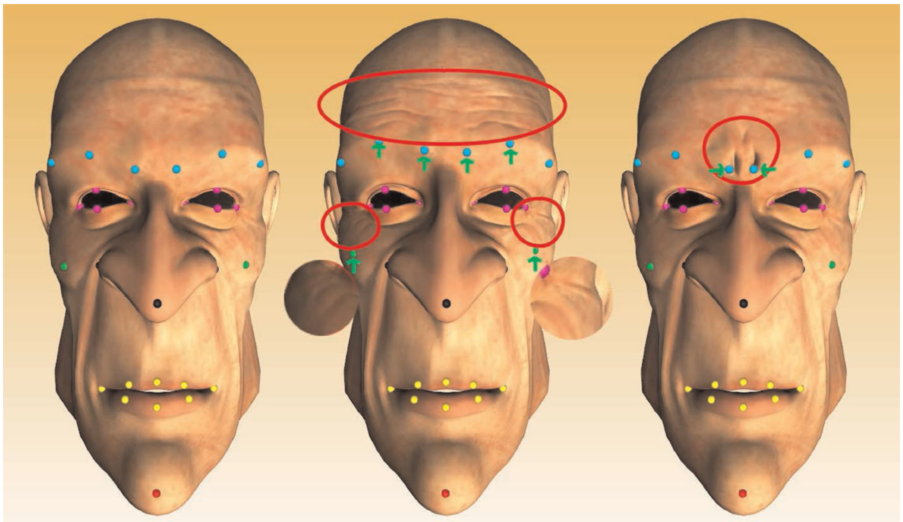


Figure 6. The neutral and the two reference poses.

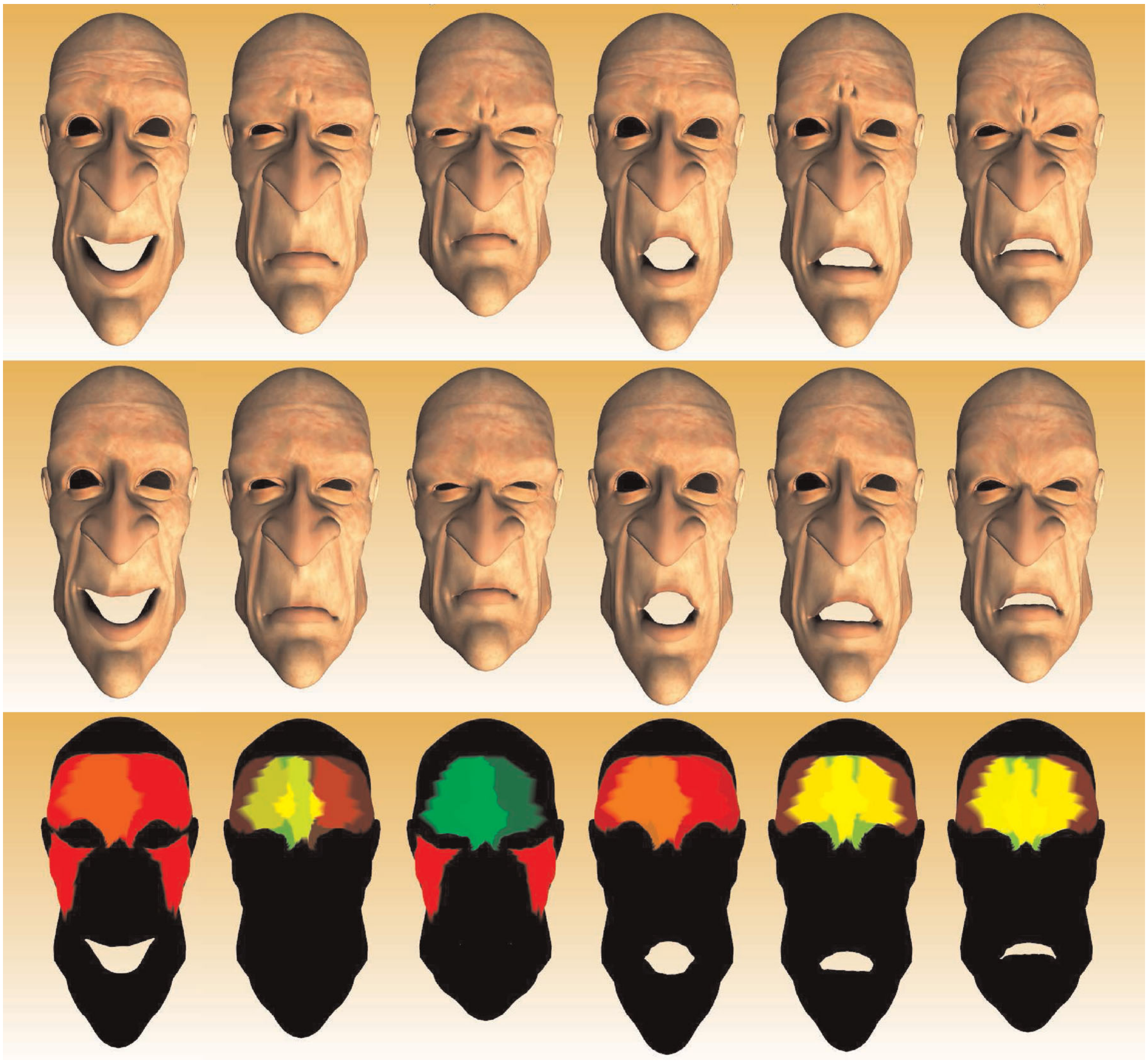


Figure 7. The first row shows the six basis expressions (joy, sad, disgust, surprise, fear and angry) of our character with captured dynamic wrinkles. The second row shows the same expressions without dynamic wrinkles. The last row shows the reference poses influences on the mesh (red for the first reference pose and green for the second).

animation and rendering pipeline. In future works, it may be interesting to consider specular reflection of the skin in our acquisition step. We will also try to improve our image-based details extraction to a video-based one, allowing to apply winkles in real-time from a captured face. Regarding the 3D aspect, last advances in graphics hardware provide local tesselation ability, it should be easy to adapt our bump-map method to a local mesh refinement with very similar input textures.

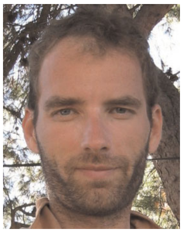
REFERENCES

1. Courgeon M, Buisine S, Martin J-C. Impact of expressive wrinkles on perception of a virtual character's facial expressions of emotions. In *IVA '09: Proceedings of the 9th International Conference on Intelligent Virtual Agents*, 2009.
2. Ersotelos N, Dong F. Building highly realistic facial modeling and animation: a survey. *The Visual Computer* 2007; **24**(1): 13–30.
3. Gain J, Bechmann D. A survey of spatial deformation from a user-centered perspective. *ACM Transactions on Graphics* 2008; **27**(4): 107:1–107:21.
4. Oat C. Animated wrinkle maps. In ACM SIGGRAPH 2007 Courses, 2007.
5. Weise T, Leibe B, Van Gool L. Fast 3d scanning with automatic motion compensation. In *CVPR '07: Computer Vision and Pattern Recognition*, 2007.

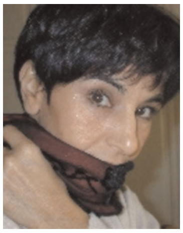
6. Alexander O, Rogers M, Lambeth W, Chiang M, Debevec P. The digital emily project: photoreal facial modeling and animation. In ACM SIGGRAPH Courses, 2009.
7. Wilson CA, Ghosh A, Peers P, Chiang J-Y, Busch J, Debevec P. Temporal upsampling of performance geometry using photometric alignment. *ACM Transactions on Graphics* 2010; **29**(2): 17:1–17:11.
8. Beeler T, Bickel B, Beardsley P, Sumner B, Gross M. High-quality single-shot capture of facial geometry. *ACM Transactions on Graphics* 2010; **29**(4): 40:1–40:9.
9. Bradley D, Heidrich W, Popa T, Sheffer A. High resolution passive facial performance capture. *ACM Transactions on Graphics* 2010; **29**(4): 41:1–41:10.
10. Durou J-D, Falcone M, Sagona M. Numerical methods for shape-from-shading: a new survey with benchmarks. *Computer Vision and Image Understanding* 2008; **109**(1): 22–43.
11. Prados E, Faugeras O. Shape from shading: a well-posed problem? In *CVPR '05: Computer Vision and Pattern Recognition*, 2005.
12. Vogel O, Valgaerts L, Breuß M, Weickert J. Making shape from shading work for real-world images. In *Proceedings of the 31st DAGM Symposium on Pattern Recognition* 2009; 191–200.
13. Wu T-P, Sun J., Tang C.-K., Shum H.Y. Interactive normal reconstruction from a single image. *ACM Transactions on Graphics* 2008; **27**(5): 119:1–119:9.
14. Smith WA, Hancock ER. Facial shape-from-shading and recognition using principal geodesic analysis and robust statistics. In *International Journal of Computer Vision* 2008; **76**(1): 71–91.
15. Bickel B, Botsch M, Angst R, Matusik W, Otaduy M, Pfister H, Gross M. Multi-scale capture of facial geometry and motion. In *ACM Transactions on Graphics* 2007; **26**(3).
16. Bando Y, Kuratate T, Nishita T. A simple method for modeling wrinkles on human skin. In *PG '02 Proceedings of the 10th Pacific Conference on Computer Graphics and Applications* 2002; 166–172.
17. Wang H, Hecht F, Ramamoorthi R, O'Brien J. Example-based wrinkle synthesis for clothing animation. *ACM Transactions on Graphics* 2010; **29**(4): 107:1–107:8.
18. Larboulette C, Cani M-P. Real-time dynamic wrinkles. *Computer Graphics International* 2004.
19. Lo Y-S, Lin I-C, Zhang W-X, Tai W-C, Chiou S-J. Capturing facial details by space-time shape-from-shading. *CGI '08: Proc. Computer Graphics International* 2008; pp. 118–125.
20. Na K, Jung M. Hierarchical retargetting of fine facial motions. *Computer Graphics Forum* 2004; **23**(3): 687–695.
21. de Melo C, Gratch J. Expression of emotions using wrinkles, blushing, sweating and tears. In *IVA '09: Proceedings of the 9th International Conference on Intelligent Virtual Agents*, 2009.
22. Liu Z, Shan Y, Zhang Z. Expressive expression mapping with ratio images. *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques* 2001; 271–276.
23. Siddiqui AM, Masood A, Saleem M. A locally constrained radial basis function for registration and warping of images. *Pattern Recognition Letters* 2009; **30**(4): 377–390.
24. Woodham RJ. Photometric method for determining surface orientation from multiple images. *Optical Engineering* 1980; **19**(1): 139–144.
25. Robles-Kelly A, Hancock ER. Shape-from-shading using the heat equation. *IEEE Transactions on Image Processing* 2007; **16**(1): 7–21.

AUTHORS' BIOGRAPHIES

Ludovic Dutreve is currently a final year PhD student in Université Lyon 1 (France) at LIRIS laboratory. He received his MSc (2007) in Université Lyon 1. His research interests include facial animation, character animation, and motion capture.



Alexandre Meyer received his PhD degree in Computer Science from Université Grenoble 1 in 2001, under the supervision of Fabrice Neyret in the iMAGIS/EVASION team. His work was on computer graphics rendering of trees in large scenes like landscape. From 2002 to 2003, he was a postdoctoral fellow at University College London in the Virtual Environments and Computer Graphics group where he worked with Prof. Celine Loscos on crowd simulation and rendering. Since 2004, he is lecturer at Université Lyon 1 and member of the LIRIS research lab. His current research concerns computer animation of characters.



Saida Bouakaz received her PhD from Joseph Fourier University in Grenoble, France. She is currently a professor in the Computer Science at Claude Bernard University, Lyon. Her research interests are in computer vision and graphics including motion capture and analysis, gesture recognition, facial animation.