Modeling shapes and textures from images: new frontiers

L. Van Gool[1,2], D. Vandermeulen[3], G. Kalberer[2], T. Tuytelaars[1], and A. Zalesny[2]
[1] ESAT/PSI/Visics, KULeuven, Belgium
[2] D-ITET/BIWI, ETH Zurich, Switzerland
[3] ESAT/PSI/MIC, KULeuven. Belgium

Abstract

Increasingly, models of the world are directly built from images. The paper discusses a number of recent developments that try to push the enveloppe of what image-based modeling can achieve. In particular, the analysis of 3D surface deformations is discussed for face animation, the extraction of matches under wide baseline conditions for 3D scene reconstruction, and the synthesis of viewpoint dependent textures for realistic object rendering.

1 Image-based modeling

During the last few years, low-cost and user-friendly solutions for 3D modeling have become available. Shape-from-video [13, 23, 12] extracts 3D shapes and their textures from video sequences as the only input. One-shot structured light techniques [33, 24] get such information from a single image, but need the projection of a special pattern. These techniques have the advantage that they are cheaper than traditional solutions like dedicated multi-camera rigs or laser scanners, as they only require off-the-shelf hardware. Moreover, they offer more flexibility in terms of portability and the range of object sizes they can handle.

This paper presents ongoing work on three extension of such systems.

Deformable shapes: The detailed capture of deformable 3D shapes is to a large extent still an open challenge. We discuss preliminary results for faces. Based on a one-shot, structured light method, 3D deformations are extracted. In particular, face dynamics during speech are acquired, analysed, and resynthesised for animation.

Wide-baseline matching: Shape-from-video requires large overlap between subsequent frames. Often, one would like to reconstruct from a small number of stills, taken from very different viewpoints. Based on local, viewpoint invariant features, wide-baseline matching ia made possible, and hence the viewpoints can be farther apart.

3D textures: Texture mapping is an old trick to hide the absence of geometric detail. A serious shortcoming of traditional texture mapping is that changing self-occlusions and shadows which result from changing viewpoint or illumination resp. cannot be simulated. Based on a series of views of a textured surface, a texture model is extracted, that captures viewpoint dependencies of the surface's appearance.

2 Face animation

Realistic face animation still is a challenge. Faces are the focus of attention for human observers, and even the smallest deviations from real speech are noticed. One has to deal with subtle effects, that leave strong impressions. Although recent computer animation movies have shown convincing results, there still is a lot of manual work involved.

When using 3D modeling for face animation, the synthesis can simulate the underlying anatomy of a face, or only generate the exterior, visible shape. If one can model the anatomy really well, one has very good control over the face, even for expressions that have not been observed before. With its many muscles, the face anatomy is very complicated, however. Work on emotional expressions by Pighin *et al.* [22] has demonstrated that realistic animation can also be achieved without such detailed knowledge. Their animations are based solely on observed, exterior face shape. These represent a kind of keyframes, between which a linear morph is applied.

Here a similar approach is presented – that is also only based on the extraction of exterior shapes – but for the more subtle case of speech animation. This is a harder problem than emotions as higher levels of geometric detail are required. Moreover, simple morphs between mouth positions do not capture the subtle co-articulation effects of fluent speech. Our work is not the first attempt (see e.g. [25]), but

seems to be more automated and based on data of higher spatiotemporal resolution.

2.1 Extracting example visemes

Animation of speech has much in common with speech synthesis. Rather than composing a sequence of phonemes according to co-articulation principles, animation generates sequences of *visemes*. These are the basic mouth deformations during speech. Whereas there is a consensus about the set of phonemes, there is less unanimity about the selection of visemes. There is no one-to-one relation between the 52 phonemes and the visemes, as different sounds may look the same and v.v. Realistic animation experiments have used any number from as few as 16 [9] up to about 50 visemes [26]. At least as important are the co-articulation principles that are used.

We based our selection of visemes on the work of Owens [21] for consonants. We use his consonant groups that yield the same visual impression when uttered, but do not consider all the possible instances of different, neighboring vocals that he mentions. In fact, we only consider two cases: rounded and widened, that represent the instances farthest from the neutral expression (lips closed and relaxed). For the visemes that correspond to vocals, we used those proposed by Montgomery and Jackson [18]. This leads to a total of 20 visemes: 12 representing the consonants, 7 representing the monophtongs, and one representing the neutral pose. This viseme selection differs from others proposed earlier. It contains more consonant visemes than most, mainly because the distinction between the rounded and widened shapes is made systematically. This selection seems to be a good compromise between the number of visemes and the realism that is obtained.

The face deformations corresponding to these visemes had to be analysed carefully. These deformations were extracted for faces of different age, race, and sex. Speech affects the entire facial structure below the eyes [19]. Therefore, we extracted 3D data for a complete face, but with emphasis on the area between the eyes and the chin. The 3D viseme extraction follows a number of steps, which are repeated for the different example faces.

The process starts with that every test subject says a sentence, that contains all the visemes at least once, but typically twice or more. This is captured in 3D using Eyetronics' ShapeSnatcher system [8] It projects a grid onto the face, and extracts the 3D shape and texture fromET a single image. By using a video camera, a quick succession of 3D snapshots can be acquired. We are especially interested in frames that represent the different visemes. These are the frames where the lips reach their extremal positions for that sound (Ezzat and Poggio [9] followed the same approach in 2D). The acquisition system yields the 3D coordinates of

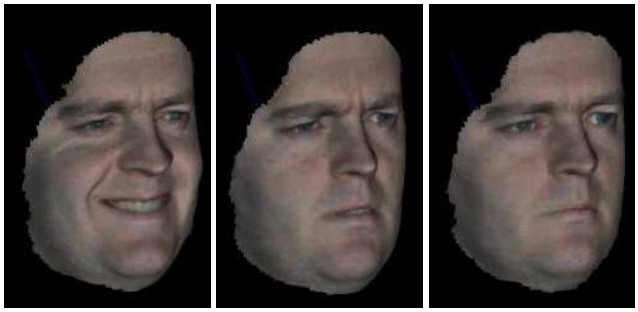

Figure 1. *3D Snapshots of a talking face, for one of the test subjects.*

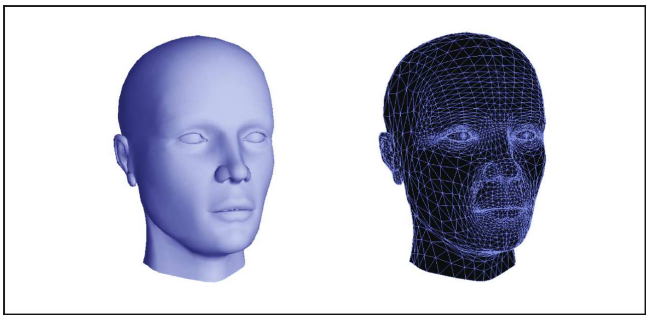

Figure 2. *Left: generic head model, Right: underlying mesh. This generic head model has been produced by Duran (the outer skin) and by Imagination in Motion (tongue and teeth), in the context of the European Mesh project.*

thousands of points for every frame. The output is a connected, triangulated and textured surface. Fig. 1 shows a few 3D snapshots obtained from such an acquisition session.

The problem is that the 3D points correspond to projected grid intersections, not corresponding, physical points on the face. Hence, the points for which 3D coordinates are given change from frame to frame. The next steps have to solve for the physical correspondences.

Physical correspondences are solved by mapping the 3D data onto a generic head mesh. This is a triangulated surface with 2268 vertices for the skin, supplemented with seperate meshes for the eyes, teeth, and tongue (another 8848, mainly for the teeth). Fig. 2 shows the generic head and its topology. This generic head model is fitted to the 3D data of the example face (i.e. 3D neutral face data of one of the test subjects) in a total of three steps. The first step in this fitting procedure deforms the generic head by a simple rotation, translation, and anisotropic scaling operation, to crudely align salient features like eye corners, nose tip,

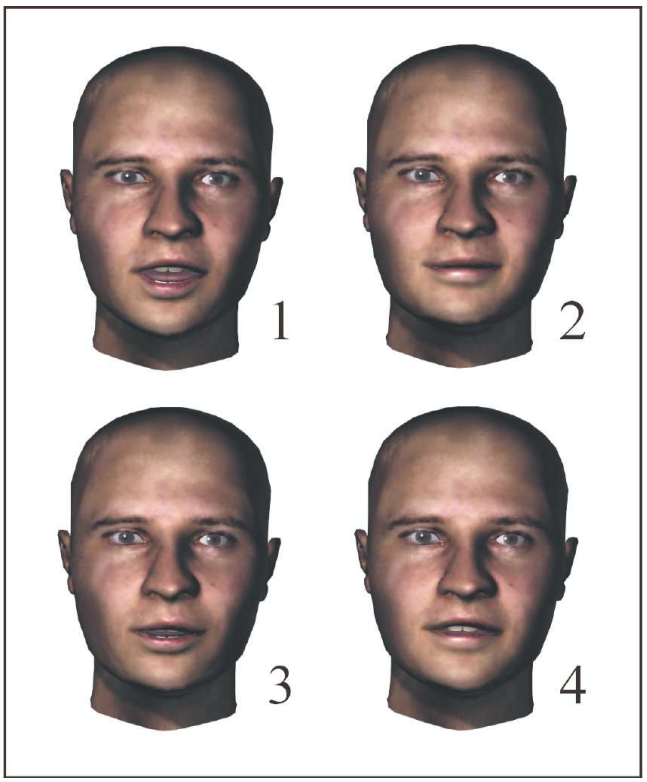

Figure 3. *Four visemes for one of the test subjects.*

etc., with those on the neutral shape of the example face. After this initial transformation, the salient features are better aligned through a piecewise constant, vertical stretch in 5 facial regions: from top-of-head to eyebrows, from eyebrows to eye corners, from eye corners to nose tip, from nose tip to mouth corners, and from mouth corners to bottom of the chin. The third step performs a local morph. This morphing maps the topology of the generic head precisely onto the example shape. In order to ease this mapping, the example faces had about 100 points marked as black dots.

These three steps are only applied once to the neutral face of a test person. From the initial, neutral frame the points are tracked throughout the video and the mesh adapts automatically to subsequenct 3D snapshots for non-neutral poses. The special facial features and the marked points were extracted in 3D from all frames, the mesh was deformed to keep these points aligned, and intermediate mesh points were positioned with the help of Radial Basis Functions [20] and projection onto the measured 3D surface.

In order to get the catalogue of 3D visemes for a single test person, the corresponding frames were selected from the video and their 3D meshes were averaged over different instances of the same viseme and stored. A number of visemes for one of the example faces is given in fig. 3. As a matter of fact, not the 3D meshes themselves were stored, but the difference with respect to the neutral one for the same person. These deformation fields of a single person still contain a lot of redundancy. This was investigated by applying a Principal Component Analysis. Over 99% of the variance in the deformation fields was found in the space spanned by the first 6 components. This space is referred to as the 'Viseme Space' of the person.

2.2 Bringing faces to life

The previous section described an approach to extract a set of visemes from talking faces, observed with the ShapeSnatcher system. This section describes how novel, static 3D face models, for which no such information is available, can be animated.

Such animation requires a number of steps:

personalising the visemes: a set of visemes, adapted to the physiognomy of the novel face is generated;

automatic, audio-based animation: from spoken text a time stamped sequence of visemes is generated, that drives the animation;

possibly modifications by the animator: ICA based tools allow the animator to modify the result.

A good animation requires visemes that are adapted to the shape or 'physiognomy' of the face at hand. One cannot simply copy or 'clone' the deformations that have been extracted from one of the example faces. The adapted visemes are created in a simple way, that in fact needs further validation. Faces can be efficiently represented as points in a so-called 'Face Space' [4]. These points represent their deviation from the average face along some principal modes. Hence, the novel face as well as the neutral, example faces correspond to such points. The example faces span a hyperplane in face space. By orthogonally projecting the novel face onto this plane, a linear combination in terms of the example faces is found, that comes closest to the novel face. This procedure is illustrated in fig. 4 and yields weights that can be applied to the visemes of the example faces to generate a set for the novel face.

Once the personalised viseme set has been produced, an Independent Component Analysis (ICA) is applied to them. The visemes are represented as points in their 6D IC space, coined 'Viseme Space'. Animation then amounts to navigating through this space, from viseme to viseme. This is where the issue of co-articulation pops up. Visemes exert mutual influences, i.e. the way in which we move our lips for a certain vocal or consonant is also dependent on the previous and subsequent sounds. This is similar to spline fitting, where surrounding points influence how close a point is approached, under what orientation the trajectory passes, etc. The audio track that drives the animation specifies the

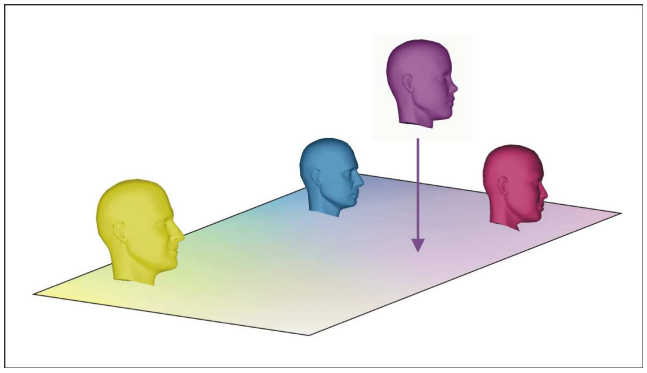

Figure 4. *Visemes are adapted to the face physiognomu by comparing it to faces for which visemes have been extracted (see text).*

visemes and their timing, but still has to be translated into a precise trajectory in Viseme Space. In our animation the trajectory is modeled as a NURBS, attracted with different strengths towards the different visemes along its path. These differences in strength reflect the variability in different viseme shapes. There is much more room for change when pronouncing 'd' than there is for 'm', for instance. A distinction is made between vocals and labial consonants on the one hand, and the remainder of the visemes on the other. The former impose their deformations much more strictly onto the animation than the latter, which can be pronounced with a lot of visual variation. In terms of the spline fitting this means that the animation trajectory will move precisely through the former visemes and will only be attracted towards the latter. The strength of this attraction differs between subclasses of the remaining visemes.

The foregoing animation from audio runs automatically. The animator can afterwards still change the influences of the different visemes, as well as the complete NURBS that define the trajectory in Viseme Space. We have used independent rather than principal components as they were found to provide a more intuitive basis.

3 Wide-baseline matching

Although 3D reconstructions can in principle be made from a limited number of stills, fully automated processing is only possible if the images have much overlap and are offered in the order of a continuous camera motion. The name 'shape-from-video' underlines this assumption. In order to automate similar reconstructions from stills that are taken from substantially different viewpoints, the computer should find correspondences under 'wide baseline' conditions. Consider the wide baseline image pair of fig. 5. The two images have been taken from very different viewing di-

Figure 5. *Two images of the same scene, but taken under very different viewing directions. The task is to find an initial set of features, that suffice to extract the epipolar geometry, which can then serve as a support for dense correspondence search.*

rections. Stereo and shape-from-video systems will most often not even get started in such cases, as correspondences are difficult to find.

The shape-from-video approach splits the correspondence problem into two stages. The first stage determines correspondences for a set of discrete features, usually corners. Based on these correspondences, the epipolar geometry for image pairs is determined. The second stage searches dense correspondences along the epipolar lines. Here, we propose a similar strategy for wide baseline matching. The focus of the discussion is on the first step, i.e. the initial matching of a discrete set of features.

3.1 The extraction of invariant neighbourhoods

When looking for initial features to match, we should focus on local structures. Otherwise, occlusions and changing backgrounds will cause problems, certainly under wide baseline conditions. Here, we look at small regions, constructed around or near interest points. If these regions are to be matched, they ought to cover the same part of the scene in the different views. Hence, they have to take on different shapes in the different images. The most important aspect of the strategy proposed here is that the region extraction works on the basis of individual images, i.e. without any knowledge about the other images. This property is key in avoiding a slow and combinatoric search for matches. In the proposed scheme regions are constructed in one go based on a single image, instead of by selecting a region in one image and then trying to find a match by deforming and relocating a region in the other image until some matching score surpasses a threshold. Here, the corresponding region in the second image is extracted independently, before one even attempts to match regions. The crux of the matter is that every step in the region extraction is invariant under the image variations one wants to be robust against. This is discussed in more detail next.

On the one hand the viewpoint may strongly change. Hence, the extraction has to survive affine deformations of the regions, not just in-plane rotations and translations. In fact, affine transformation also not fully cover the observed changes. This model will only suffice for regions that are sufficiently small and planar. We assume that a reasonable number of such regions will be found, an expectation borne out in practice. On the other hand, strong changes in illumination conditions may occur between the views. The chance of this happening will actually grow with the angle over which the camera rotates. The relative contributions of light sources will change more than in the frame-to-frame changes in a video. We model the effects of changing illumination by scaling the three colour bands (R, G, B) with different scale factors and by adding different offsets.

If we now want to construct regions that are in correspondence irrespective of these changes, every step in their construction ought to be invariant under both the geometric and photometric transformations just described. We have developed several such construction processes [28, 29] and the example regions in fig. 6 have been construced like that. As mentioned before, these constructions allow the computer to extract the regions in the two views completely independently. After they have been constructed, they can be matched efficiently on the basis of features that are extracted from the colour patterns that they enclose. These features again are invariant under both the geometric and the photometric transformations considered. To be a bit more precise, a feature vector of moment invariants is used. Recently, several additional constructions have been proposed by other researchers [3, 16]. Fig. 6 shows the regions that have been extracted for fig. 5. Only a restricted set of matching regions are shown, in order not to overload the figure. We refer to the regions as 'invariant neighbourhoods'.

3.2 Further wide baseline issues

The matching of invariant neighbourhoods is only the first step in the search for correspondences. Good 3D models require the selection of dense, pixelwise correspondences. Although the invariant neighbourhoods can provide us with the epipolar geometry, also the search along the epipolar lines for the dense correspondences requires adaptations. Disparities tend to get larger, a smaller part of the scene is visible to both cameras, and intensities of corresponding pixels vary more. In order to better cope with such problems, we propose a scheme that is based on the coupled evolution of Partial Differential Equations or the 'CODIM' scheme (COupled DIffusion Maps). This approach is described in more detail in a companion paper [27]. The point of departure of this method is optical flow. An important difference is that the search for correspondences is 'bi-local', in that spatio-temporal derivatives are taken at two different points in the two images. Disparities or motions are subdivided into a current estimate and a residue, which is reduced as the iterative process works its way towards the solution. This decomposition makes it possible to focus on the smaller residue, which is in better agreement with the linearisation that is behind optical flow.

If partial 3D reconstructions have already been produced from different photo sets, registration may better be done in 3D. The state-of-the-art in 3D registration is similar to that in 2D. Several, excellent methods have been proposed to precisely fit together partial, 3D reconstructions from initial positions that are close to the final solution [2, 5, 32]. This is very important, as it is usually easier to manually position the 3D patches more or less right, than it is to perform the fine docking by hand. Of course, it would be nicer if also the initial, crude positioning could be done by the computer,

Figure 6. *Invariant regions that were extracted for the images in fig. 5. Only regions are shown for which a corresponding partner in the other image has been found, but the regions in the two images have been extracted without knowledge about the other image.*

as this would render the whole registration automatic.

Again, invariants have proven instrumental in the development of methods that achieve such crude registration from arbitrary, initial 3D patch positions. They use special points or curves on the surface, which are characterised with invariants [10, 14]. A feature type that we have found to be particularly useful are bitangent curves. These curves are formed as follows. Suppose a plane touches the surface at two points (i.e. it is a 'bitangent plane'). Now one roles this plane over the surface so that it keeps in touch at two points. This yields pairs of bitangent curves. They are interesting, because they are invariant under Euclidean, affine, and even projective transformations. Moreover, the curve pairs can be given simple, invariant descriptions, especially in the case of Euclidean and affine transformations. These descriptions require only first derivatives [30].

4 Viewpoint-dependent texturez

Over recent years, image-based techniques for scene rendering have been developed (e.g the seminal lumigraph [11] work by Gortler et al.). These techniques can yield stunning visual quality, without ever getting at the underlying 3D shapes. Remarkably enough, this kind of image combination has first been exploited for the visualisation of complete objects and scenes. Texture mapping is, in fact, a much earlier use of image-based rendering. Since the early days of graphics, its task has been to conceal the lack of fine surface geometry. In contrast to the previously mentioned methods, a single, fixed texture image is used. Nevertheless, using the same texture for different viewing conditions has its limitations. The simple foreshortening and smooth surface shading of traditional texture mapping cannot mimic 3D effects such as variable self-occlusions and self-shadowing. This could be remedied by letting the mapped textures depend on viewpoint and illumination direction, but without resorting to a 3D surface reconstruction. Combining a geometry-based approach for the overall shapes of objects with an image-based rendering of the surface details seems to hold good promise for the realistic visualisation of scenes. The interactions with and between the objects is easier to implement, while the photo-realism of the scene rendering can be improved.

The first steps towards multiview texture analysis and synthesis have already been taken. Firstly, the changes in textures that occur under changing viewpoints and illuminations have been recorded systematically for a series of materials [7]. In order to get a handle on these effects, several authors have developed textures descriptions that either include such changes or are invariant. The outputs of Gaussian derivative filters for different viewing conditions have been clustered and used for material recognition and reproduction of the same piece of texture [15]. Chantler

and coworkers [17] have focused on the effects of changing illumination, both for analysis and synthesis, but do not choose a purely image-based approach. The approach of Cula and Dana [6] is image-based and is oriented towards texture classification. In some respect their system comes close to ours, but we focus on the synthesis of multiview textures. An important difference of these approaches with earlier work on bump maps and relief textures is that no 3D information is extracted.

4.1 The basic texture model

Our multiview texture model is an extension of a single view texture modeling technique. The latter extracts some carefully chosen statistics from an example texture during an analysis step. Synthesis then consists of constructing textures with similar statistics. The method has the advantage that it can handle both stochastic and structural textures. It does not copy any part of the example texture, thereby avoiding repetitions in the synthesized textures. It includes both short-range and long-range pixel interactions and therefore can pick up small- and large-scale effects. The texture model is also highly compact, only a couple of Kbytes. These advantages are preserved in the extended version that includes viewpoint dependency.

The single view modeling step extracts first- and second-order statistics from an example image. The first-order statistics correspond to the intensity histograms. The second-order statistics draw upon the cooccurrence principle: for pixel pairs at fixed relative positions the intensities are compared. The pixel pairs are called cliques and pairs with the same relative positions form a clique type. The clique is an ordered pair. Hence, a tail and head pixel can be distinguished. Instead of storing the complete joint probability distributions for the different clique types, our model only stores the histogram of the intensity differences between the head and tail pixels. It is not practical to collect these second-order statistics for all possible clique types. A selection of clique types is made that together contain sufficient information to generate a texture that is perceptually very similar. This selection is described elsewhere [34]. There it is also shown how this model can be used to synthesise a texture that is similar to the example texture. Suffice it here to say that the synthesis algorithm tries to generate a texture that has the same intensity difference statistics for the different clique types in the model. The same paper discusses the generalisation of these principles towards colour textures.

In summary, the basic texture model contains the information necessary to produce a texture for one viewpoint and illumination direction. This model consists of two types of information. On the one hand, there is the set of cliques types, which describe which interactions with neighbouring pixels are taken into account. This set will be referred at as the *neighbourhood system*. On the other hand, there is the intensity histogram of the textures, as well as the intensity difference histograms for the different clique types. These intensity statistics are referred to as the *statistical parameter set*. The choice of the neighbourhood system takes far more time than the extraction of the corresponding statistical parameter set.

4.2 Multiview texture models

The multiview texture model that we have proposed [34, 35], takes the single view model as its point of departure. The adaptations towards different viewing conditions are twofold. On the one hand, an affine deformation is applied to the neighbourhood system, in accordance with the change in viewing direction. Typically, the neighbourhood system is extracted once from a fronto-parallel view of the texture. This neighbourhood system is then affinely deformed in accordance with the slant and tilt angles under which other views are taken. Further refinements and changes in illumination are then taken into account through a complete update of the statistical parameter set for every novel view: all the histograms are extracted anew for the affinely deformed set of cliques, from the novel view taken under known conditions. This is repeated for all such examples, i.e. for all example images for different viewing and/or illumination directions. Complete knowledge about these directions is assumed. The advantage of this multiview modeling is that it hardly takes longer than the modeling for a single, overhead view. The neighbourhood system does not have to be selected again, as it is obtained through simple deformation. The extraction of the updated statistical parameter sets can be done very quickly.

Textures corresponding to unseen viewing conditions can only be generated in as far as sufficiently similar viewing conditions have been observed. The neighbourhood system can be adapted easily, but the statistical parameter set has to be obtained from interpolation or – and this may be a problem – extrapolation of those observed from similar cases. This interpolation or extrapolation is simplified by a PCA description of the histograms. This leads to very compact descriptions of these histograms, as weighted combinations of the principal components. Splines fitted to these weights yield the interpolated (or extrapolated) values. For this method to work well, sufficient examples must have been provided. An example of multiview texture is given in fig. 7. The top orange is a real one, the one below is a sphere clipped by the same silhouet, and oovered with multiview texture learned from the real one.

Acknowledgements: The authors gratefully acknowledge support from K.U.Leuven GOA project ‘VHS+’, ETH project ‘Visemes’, and European IST project ‘MESH’(with

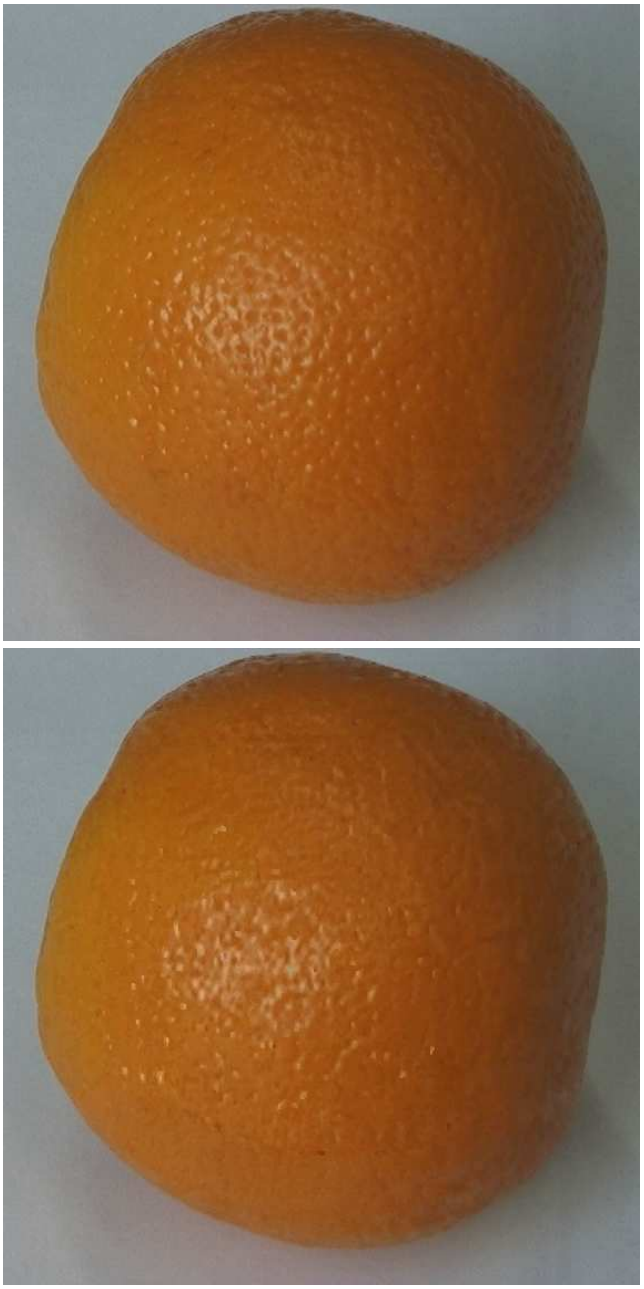

Figure 7. *The orange in the top image is real. The orange in the bottom image actually is a sphere covered with viewpoint dependent texture, and clipped with the outline of the top image. The result is far more realistic than what can be achieved through the mapping of viewpoint independent texture and simple shading.*

Duran, EPFL, Eyetronics, Un.Freiburg, Un.Geneva) and IST project 'Cimwos'. Several other people have contributed to the work described in this paper. In particular, the authors would like to thank Pascal Müller and Vittorio Ferrari.

References

[1] M. Armstrong, A. Zisserman, and P. Beardsley, Euclidean structure from uncalibrated images, 5th BMVC, 1994

[2] P. Besl, N. McKay, A method of registration of 3-D shapes, IEEE Trans. PAMI 12 (2) pp. 239-256, 1992

[3] A. Baumberg, Reliable Feature Matching across Widely Separated Views, Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp.774-781, 2000

[4] V. Blanz and T. Vetter, A morphable model for the synthesis of 3D faces, Proc. Siggraph, 1999

[5] Y. Chen and G. Medioni, Object modeling by registration of multiple range images, Proc. Int. Conf. on Robotics and Automation, pp. 2724-2729, 1991

[6] O. Cula and K. Dana, Compact representation of bidirectional texture functions, Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol.1, pp.1041-1047, 2001

[7] K. Dana, B. Van Ginneken, S. Nayar, and J. Koenderink, Reflectance and texture of real world surfaces, ACM Trans. on Graphics, vol.18, no.1, pp.1-34, 1999

[8] Eyetronics, http://www.eyetronics.com

[9] T. Ezzat and T. Poggio, Visual speech synthesis by morphing visemes, Int. J. on Computer Vision, vol.38, pp.45-57, 2000

[10] J. Feldmar and N. Ayache, Rigid, affine and locally affine registration of free-form surfaces, TR INRIA Epidaure, No. 2220, 1994

[11] S. Gortler, R, Grzeszczuk, R. Szeliski, and M. Cohen, The Lumigraph, Proc. Siggraph, pp.43–54, 1996

[12] R. Hartley and A. Zisserman, *Multiple View Geometry*, Cambridge University Press, 2000

[13] A. Heyden and K. Astrom, Euclidean reconstruction from image sequences with varying and unknown focal length and principal point, Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 1997

[14] A. Johnson and M. Hebert, Efficient multiple object recognition in cluttered3D scenes, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp.671-677, 1998

[15] T. Leung and J. Malik, Recognizing surfaces using three-dimensional textons, Proc. Int. Conf. Computer Vision, pp.1010-1017, 1999

[16] J. Matas, O. Chum, O., M. Urban, and T. Pajdla, Distinguished Regions for Wide-baseline Stereo, Technical Report Center for Machine Perception, Czech Technical University, Prague, CTU–CMP–2001–33, 2001

[17] G. McGunnigle and M. Chantler, Evaluating Kube and Pentland's fractal imaging model, IEEE Trans. on Image Processing, 10(4), pp.534-542, 2001

[18] A. Montgomery and P. Jackson, Physical characteristics of the lips underlying vowel lipreading performance, J. Acoustic Soc. Am., vol.73, pp.2134-2144, 1983

[19] K. Munhall and E. Vatikiotis-Bateson, The moving face during speech communication, in *Hearing by Eye*, eds. Campbell, Dodd, and Burnham, Vol.2, ch.6, pp.123-139, Psychology Press, 1998

[20] J.-Y. Noh and U. Neumann, Expression cloning, Proc. Siggraph, pp.277-288, 2001

[21] O. Owens and B. Blazek, Visemes observed by hearing-impaired and normal-hearing adult viewers, J. Speech and Hearing Res., vol.28, pp.381-393, 1985

[22] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. Salesin, Synthesising realistic facial expression from photographs, Proc. Siggraph, pp.75-84, 1998

[23] M. Pollefeys, R. Koch, and L. Van Gool, Self calibration and metric reconstruction in spite of varying and unknown internal camera parameters, Proc. Int. Conf. on Computer Vision, pp.90-96, 1998

[24] M. Proesmans and L. Van Gool, One-shot 3D-shape and texture acquisition of facial data, Proc. 1st Int. Conf. on Audio- and Video-based Biometric Person Authentication, pp. 411-418, Crans-Montana, Switzerland, march 1997

[25] L. Reveret, G. Bailly, and P. Badin, Mother, a new generation of talking heads providing a flexible articulatory control for videorealistic speech animation, Proc. ICSL, 2000

[26] K. Scott, D. Kagels, S. Watson, H. Rom, J. Wright, M. Lee, and K. Hussey, Synthesis of speaker facia movement to match selected speech sequences, Proc. Australian Conf. on Speech Science and Technology, vol.2, pp.620-625, 1994

[27] C. Strecha and L. Van Gool, PDE-based multi-view depth estimation, in these proceedings

[28] T. Tuytelaars, L. Van Gool, L. D'haene, and R. Koch. Matching Affinely Invariant Regions for Visual Servoing, Proc. Int. Conf. on Robotics and Automation, pp.1601-1606. 1999

[29] T. Tuytelaars and L. Van Gool, Wide Baseline Stereo based on Local, Affinely Invariant Regions, Proc. British Machine Vision Conf, pp.412-422, 2000

[30] J. Vanden Wyngaerd, L. Van Gool, R. Koch, and M. Proesmans, Invariant-based registration of surface patches, Proc. Int. Conf. on Computer Vision, pp.301-306, Kerkyra, Greece, 1999

[31] B. Van Ginneken, j. Koenderink, and K. Dana, Texture histograms as a function of irradiation and viewing direction, Int. J. of Computer Vision, vol.31, no.2/3, pp.169-184, 1999

[32] P. Viola and W. Wells, Alignment by maximisation of mutual information, Proc. Int. Conf. on Computer Vision, pp. 16-23, 1995

[33] P. Vuylsteke and A. Oosterlinck, Range Image Acquisition with a Single Binary-Encoded Light Pattern, IEEE PAMI 12(2), pp. 148-164, 1990

[34] A. Zalesny and L. Van Gool, A compact model for viewpoint dependent texture synthesis, in *Proc. SMILE 2000*, eds. Pollefeys, Van Gool, Zisserman, and Fitzgibbon, pp. 124-141, Springer LNCS 2018, 2001

[35] A. Zalesny and L. Van Gool, Multiview texture models, Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol.1, pp.615-622, Hauai, Hawaii, dec. 2001