Computing Human Faces for Human Viewers: Automated Animation in Photographs and Paintings

Volker Blanz
Institute for Vision and Graphics
University of Siegen
Siegen, Germany
blanz@informatik.uni-siegen.de

ABSTRACT

This paper describes a system for animating and modifying faces in images. It combines an algorithm for 3D face reconstruction from single images with a learning-based approach for 3D animation and face modification. Modifications include changes of facial attributes, such as body weight, masculine or feminine look, or overall head shape, as well as cut-and-paste exchange of faces. Unlike traditional photo retouche, this technique can be applied across changes in pose and lighting. Bridging the gap between photorealistic image processing and 3D graphics, the system provides tools for interacting with existing image material, such as photographs or paintings. The core of the approach is a statistical analysis of a dataset of 3D faces, and an analysis-by-synthesis loop that simulates the process of image formation for high-level image processing.

Categories and Subject Descriptors

I.3.7 [**Computer Graphics**]: Animation

General Terms

Algorithms

1. INTRODUCTION

Unlike many other fields in computer science, progress in computer graphics involves adaptations to human users in a variety of different ways: First, the image data produced in graphics has to consider and exploit the laws of human perception. For example, research in tone mapping aims at producing images at low dynamic range in intensity, which can be displayed on standard computer screens, and still reproduce the visual appearance of natural scenes that have a large dynamic range due to extreme lighting conditions. Many algorithms in tone mapping are inspired by psychophysical findings (such as [12]), and in turn, the only valid criterion for the quality of a tone mapping operator is human perception [16].

ICMI'06, November 2–4, 2006, Banff, Alberta, Canada.

Computer graphics is being adapted to the human needs also in terms of the contents and style of the images, by producing material that is meaningful in a cultural context. While most of the computer graphics images and movies produced in the 1980s and 1990s were entirely virtual scenes, located in a physically and semantically void space, visual effects today are more and more embedded into a rich context: Computer graphics is mixed with shots of natural scenes, with either of the two dominating the picture, and existing, real characters, images and objects are reproduced and manipulated. Unlike the artificial characters of early computer animation, they may now be used as virtual stunt doubles in movies such as *The Matrix*. Scenes may be altered by adding or removing buildings or other objects in each frame. The benefit of the combination of 3D graphics with natural images is a photorealistic, highly complex visual appearance with meaningful content. However, it is still challenging to achieve the same visual standard for computer graphics elements when shown side by side with natural photo material.

In facial animation, we are beginning to see a level of quality that captures even subtle facial expressions or combinations of expressions. These give artists the tools to model the complex, sometimes conflicting emotions of their characters. An example of this can be seen in the character *Gollum* in the feature film *The Lord of the Rings*. A medium that conveys more and more emotional content in artistically sophisticated narratives, computer graphics is transforming from a machine-centered to a human-centered medium, with the content not limited but enhanced by technology.

In contrast to the strive for photorealistic images, non-photo-realistic rendering has developed algorithms that give artistic styles to images and movies, some of them simulating painterly styles such as oil on canvas or watercolor, pen and ink drawings, cartoon drawings and engravings ([8, 10]). Non-photorealistic rendering, therefore, bridges the gap between electronic and traditional art, and brings a variety of interesting connotations to otherwise synthetic images. Non-photorealistic rendering may also be a powerful tool for visualization, and for making image content more comprehensible to human viewers [6].

Finally, the progress in user interfaces for content creation in computer graphics adapts this tool more and more to the demands of the artist, and it remains an interesting challenge to make the increasingly powerful algorithms in graphics easily available to users with little or no technical background. Moreover, the efficiency of the tools provided

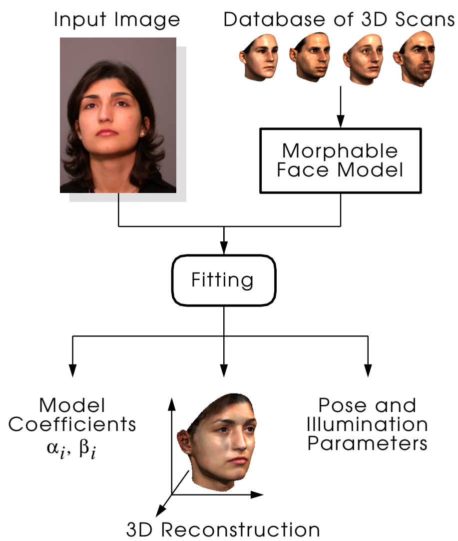

Figure 1: Fitting the Morphable Model to an image produces not only a 3D reconstruction, but also model coefficients α_i, β_i, and an estimate of head orientation, position and illumination.

by computer graphics systems may help to increase the creative power of those who use it significantly. An important component is the level of abstraction of the interaction tools and of the internal representations of computer graphics systems: As an example, consider an artist who wishes to make a character in an image more skinny. In an image-based system, such as software for digital image processing, the artist would need to shift the facial silhouette of the cheeks with copy-and-paste tools, or even paint the new silhouette with a digital brush. In 3D computer graphics, the face would be represented by a polygon mesh, and the artist would select and shift groups of vertices to change the 3D shape of the cheeks. On the highest level of abstraction, however, the artist would like to have an interactive tool, such as a slider, that controls the skinniness or fatness of a face directly, so the user would only select a face in the image and use that slider.

In this paper, we describe a system that implements this paradigm of high-level control. The system works on faces in any given photo or painting at any pose and illumination. Applied on existing image material, it provides a tool that interacts with material that is meaningful in a complex cultural background, such as paintings. As a tool that animates, modifies or exchanges faces, it addresses perhaps the most relevant content of visual media, the human face.

1.1 System Overview

Modification of faces in given images at any pose and illumination is a non-trivial problem that involves information about the 3D geometry of the face. For example, making a face more fat or skinny changes the silhouette and the shading of the face, which calls for a direct or indirect (implicit) representation of effects in 3D space, such as perspective projection, occlusion and interaction of light with matter. In our approach, we chose an explicit representation of the 3D geometry of the face.

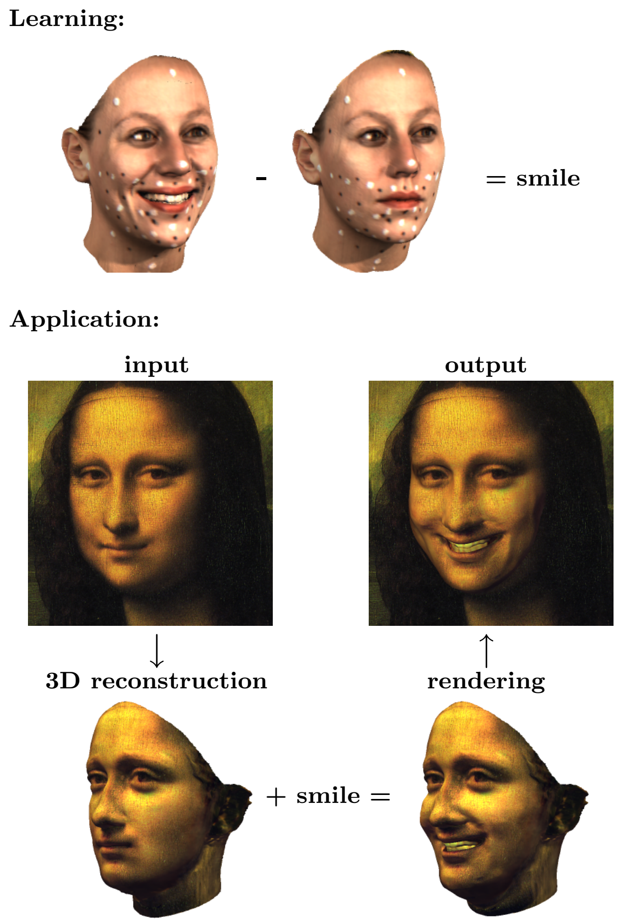

Figure 2: In the vector space of faces, facial expressions are transferred by computing the difference between two scans of the same person (top row), and adding this to a neutral 3D face. To modify Leonardo's Mona Lisa (second row), we reconstruct her 3D face (third row), add the expression, and render the new surface into the painting (second row, right).

In order to interact with faces in 3D space, given an image, we have to solve the difficult, ill-posed problem of 3D shape reconstruction from single images. This problem can only be solved by including prior knowledge about the possible 3D solutions. In 3D reconstruction from architecture, such prior knowledge may be the fact that many lines are parallel or orthogonal in 3D [9]. For human faces, the set of 3D solutions may be restricted by exploiting the statistics of face shapes. The core of our work, therefore, is a *3D Morphable Model* of faces [4] that captures the natural variations observed in human faces. This model is learned automatically from a dataset of 3D scans of faces [4]. Figure 1 illustrates the process of 3D shape reconstruction. The algorithm for shape reconstruction is fully automated, but it has to be initialized by manually labelling a set of between 6 and 15 feature points in the image. As a side effect, the algorithm computes the 3D pose and illumination parameters of the face in the scene, which can be used when the modified 3D face is drawn back into the input image.

For 3D face manipulation, we describe three modes of interaction:

- Facial animation,
- High-level control of facial attributes, such as gender, body weight and nose shape,
- Exchanging faces in images.

In all of these interactions, a 3D model of the face is reconstructed from the input image, modified in 3D, and drawn back into the original image (Figure 2.) Both the animation and the modification of high-level attributes are learning-based: changes in 3D geometry and in texture are learned from datasets of 3D scans, unlike the labor-intensive manual surface editing procedures performed by animators and modelers in production studios today.

In the following sections, we describe the Morphable Model (Section 2), summarize the algorithm for 3D shape reconstruction (Section 3), describe the approach to facial animation (Section 4), discuss our technique for exchanging faces in images (Section 5), and present a method for learning and changing attributes in faces (Section 6).

2. A MORPHABLE MODEL OF 3D FACES

The Morphable Model of 3D faces [14, 4] is a vector space of 3D shapes and textures spanned by a set of examples. Derived from textured *Cyberware* (TM) laser scans of 200 individuals, the Morphable Model captures the variations and the common properties found within this set. Shape and texture vectors are defined such that any linear combination of examples $\mathbf{S}_i$, $\mathbf{T}_i$,

$$\mathbf{S} = \sum_{i=1}^{m} a_i \mathbf{S}_i, \quad \mathbf{T} = \sum_{i=1}^{m} b_i \mathbf{T}_i. \tag{1}$$

is a realistic face if $\mathbf{S}$, $\mathbf{T}$ are within a few standard deviations from their averages. Each vector $\mathbf{S}_i$ is the 3D shape of a polygon mesh, stored in terms of x, y, z-coordinates of all vertices $j \in \{1, \ldots, n\}$, $n = 75972$:

$$\mathbf{S}_i = (x_1, y_1, z_1, x_2, \ldots, x_n, y_n, z_n)^T. \tag{2}$$

In the same way, we form texture vectors from the red, green, and blue values of all vertices' surface colors:

$$\mathbf{T}_i = (R_1, G_1, B_1, R_2, \ldots, R_n, G_n, B_n)^T. \tag{3}$$

Such a definition of shape and texture vectors is only meaningful if the vector components of all vectors $\mathbf{S}_i$, $\mathbf{T}_i$ have point-to-point correspondence: Let $x_{i,j}$,$y_{i,j}$,$z_{i,j}$, be the coordinates of vertex j of scan i. Then, for all scans i, this has to be the same point, such as the tip of the nose or the corner of the mouth. Using an algorithm derived from optical flow, we compute dense correspondence for all $n = 75972$ vertices automatically [4].

Finally, we perform a Principal Component Analysis (PCA, see [7]) to estimate the probability distributions of faces around their averages $\bar{\mathbf{s}}$ and $\bar{\mathbf{t}}$. This gives us a set of m orthogonal principal components $\mathbf{s}_i$, $\mathbf{t}_i$, and the standard deviations $\sigma_{S,i}$ and $\sigma_{T,i}$ of the dataset along these axes. We can now replace the basis vectors $\mathbf{S}_i$, $\mathbf{T}_i$ in Equation (1) by $\mathbf{s}_i$, $\mathbf{t}_i$:

$$\mathbf{S} = \bar{\mathbf{s}} + \sum_{i=1}^{m} \alpha_i \cdot \mathbf{s}_i, \qquad \mathbf{T} = \bar{\mathbf{t}} + \sum_{i=1}^{m} \beta_i \cdot \mathbf{t}_i. \tag{4}$$

In the following, we use the $m = 149$ most relevant principal components only, since the other components tend to contain noise and other non class-specific variations.

3. ESTIMATION OF 3D SHAPE, TEXTURE, POSE AND LIGHTING

From a given set of model parameters α and β (4), we can compute a color image $\mathbf{I}_{model}(x, y)$ by standard computer graphics procedures, including rigid transformation, perspective projection, computation of surface normals, Phong illumination, and rasterization. The image depends on a number of rendering parameters ρ. In our system, these are 22 variables:

- 3D rotation (3 angles)
- 3D translation (3 dimensions)
- focal length of the camera (1 variable)
- angle of directed light (2 angles)
- intensity of directed light (3 colors)
- intensity of ambient light (3 colors)
- color contrast (1 variable)
- gain in each color channel (3 variables)
- offset in each color channel (3 variables).

All parameters are estimated simultaneously in an analysis-by-synthesis loop. The main goal of the analysis is to find the parameters α, β, ρ that make the synthetic image $\mathbf{I}_{model}$ as similar as possible to the original image $\mathbf{I}_{input}$ in terms of pixel-wise image difference in the red, green and blue channel:

$$E_I = \sum_x \sum_y \sum_{c \in \{r,g,b\}} (I_{c,input}(x, y) - I_{c,model}(x, y))^2. \tag{5}$$

All scene parameters are recovered automatically, starting from a frontal pose in the center of the image, and at frontal illumination. To initialize the optimization process, we use a set of between 6 and 15 feature point coordinates [5]: The manually defined 2D feature points $(q_{x,j}, q_{y,j})$ and the image positions (p_{x,k_j}, p_{y,k_j}) of the corresponding vertices k_j define a function

$$E_F = \sum_j \| \begin{pmatrix} q_{x,j} \\ q_{x,j} \end{pmatrix} - \begin{pmatrix} p_{x,k_j} \\ p_{y,k_j} \end{pmatrix} \|^2. \tag{6}$$

that is minimized along with the image difference E_I in the first iterations.

In order to avoid overfitting effects that are well-known from regression and other statistical problems (see [7]), we add regularization terms to the cost function that penalize solutions that are far from the average in terms of shape, texture, or the rendering parameters:

$$E_{reg} = \sum_i \frac{\alpha_i^2}{\sigma_{S,i}^2} + \sum_i \frac{\beta_i^2}{\sigma_{T,i}^2} + \sum_i \frac{(\rho_i - \bar{\rho}_i)^2}{\sigma_{R,i}^2}. \tag{7}$$

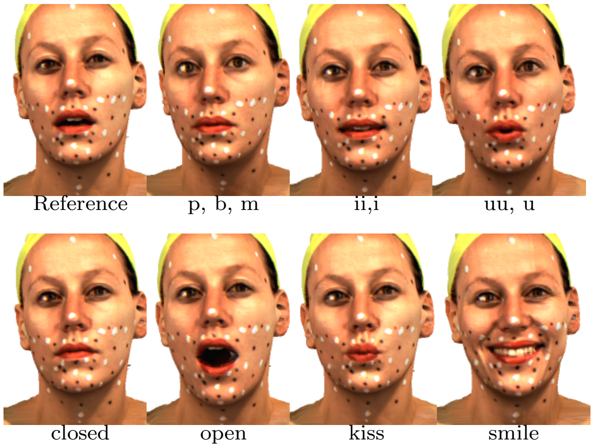

Figure 3: Examples from the dataset of 35 static 3D laser scans forming the vector space of mouth shapes and facial expressions. 17 scans show different visemes, others show the mouth opening gradually.

$\overline{\boldsymbol{\rho}}_i$ are the standard starting values for ρ_i, and $\sigma_{R,i}$ are ad–hoc estimates of their standard deviations. The full cost function

$$E = \frac{1}{\sigma_I^2} E_I + \frac{1}{\sigma_F^2} E_F + E_{reg} \tag{8}$$

can be derived from a maximum-a-posteriori approach that maximizes the posterior probability of α, β and ρ, given $\mathbf{I}_{input}$ and the feature points [4, 5]. The optimization is performed with a Stochastic Newton Algorithm [5]. The fitting process takes 3 minutes on a 3.4GHz Xeon processor.

The linear combination of textures $\mathbf{T}_i$ cannot reproduce all local characteristics of the novel face, such as moles or scars. We extract the person's true texture at high resolution from the image by an illumination-corrected texture extraction algorithm [4]. At the boundary between the extracted texture and the predicted regions, we produce a smooth transition based on a reliability criterion for texture extraction that depends on the angle between the viewing direction and the surface normal. Structures that are visible on one and occluded on the other side of the face can be reflected, due to facial symmetry.

4. FACIAL ANIMATION IN IMAGES

The Morphable Model can not only represent variations across the faces of different persons, but also changes in shape and texture of an individual face due to facial expressions. Morphing between such face vectors generates smooth, continuous transitions as they occur over time t when a face moves:

$$\mathbf{S}(t) = (1 - \lambda(t)) \cdot \mathbf{S}_{expression1} + \lambda(t) \cdot \mathbf{S}_{expression2}, \tag{9}$$

with a scalar function $\lambda(t)$ that controls the rate at which the face transforms from expression 1 to expression 2.

Unlike previous approaches to facial animation, such as parameterized models that are designed by artists (for an overview, see [11]) or models that simulate the physical properties of muscles and tissue [13], our technique relies entirely on observations of facial expressions on human faces. An automated algorithm learns how points on the surface move as a person acts or speaks, and these movements can be transferred to novel faces.

In order to learn the degrees of freedom of faces in facial expressions and speech from data, we recorded a set of 35 static laser scans of one person (Figure 3). 17 of the scans show different visemes, which are the basic mouth shapes that occur in human speech. Mouth movements and expressions learned from these scans can then be transferred to new individuals by a simple vector operation (Figure 2):

$$\Delta\mathbf{S}_{expression} = \mathbf{S}_{expression,person1} - \mathbf{S}_{neutral,person1} \tag{10}$$

$$\mathbf{S}_{expression,person2} = \mathbf{S}_{neutral,person2} + \Delta\mathbf{S}_{expression}. \tag{11}$$

This simple linear approach assumes that the deformations of faces are identical for all individuals, which is only an approximation of the individual expressions observed in real faces. A full investigation and a statistical analysis of the individual differences in expressions requires a large database of different persons' expressions [15]. One of the challenges in this approach is to extrapolate to novel, unknown faces, and to avoid overfitting and other statistical problems. Our direct transfer of expressions, therefore, is a safe guess at the price of missing some of the idiosyncrasies.

To convert the scan data into shape and texture vectors of the Morphable Model, it is essential to compute dense point-to-point correspondence between different expressions in the same way as between scans of individuals (Section 2). However, the large differences in geometry between open and closed mouth scans, and the fact that surface regions such as the teeth are visible in some mouth poses and occluded in others, make this problem significantly more difficult, and requires additional techniques as described in [2]. Unlike the face, which is morphed in a non-rigid way during animation, the teeth are rigid surface elements located behind the lips in 3D space. Their position is fixed with respect to the head (upper teeth) or the chin (lower teeth). Taken from a single, open mouth scan of the subject shown in Figure 3, these teeth can be inserted into novel 3D faces by a simple scale and translation operation, based on the positions of the corners of the mouth [2]. Figure 4 shows a set of examples. In paintings, the strokes of the brush are captured by the high-resolution texture extracted from the image.

5. EXCHANGING FACES IN IMAGES

The Morphable Model and the fitting procedure described in Section 3 achieve a full separation of parameters that are characteristic for an individual, i.e. shape and texture, from scene parameters that are specific for a given image, such as pose and illumination. Therefore, it is straight forward to exchange faces in images by replacing the shape and texture parameters α β or vectors $\mathbf{S}$ and $\mathbf{T}$, while keeping the scene parameters unchanged [3]. This process is summarized in Figure 7.

The Morphable Model only covers the face, including forehead, ears and neck, but not the back of the head and the shoulder. We leave these unchanged by using the original image as a background to the novel face. Alpha blending along the cutting edges of the facial surface produces a smooth transition between original and modified regions. In contrast to cutting edges, the occluding contours along

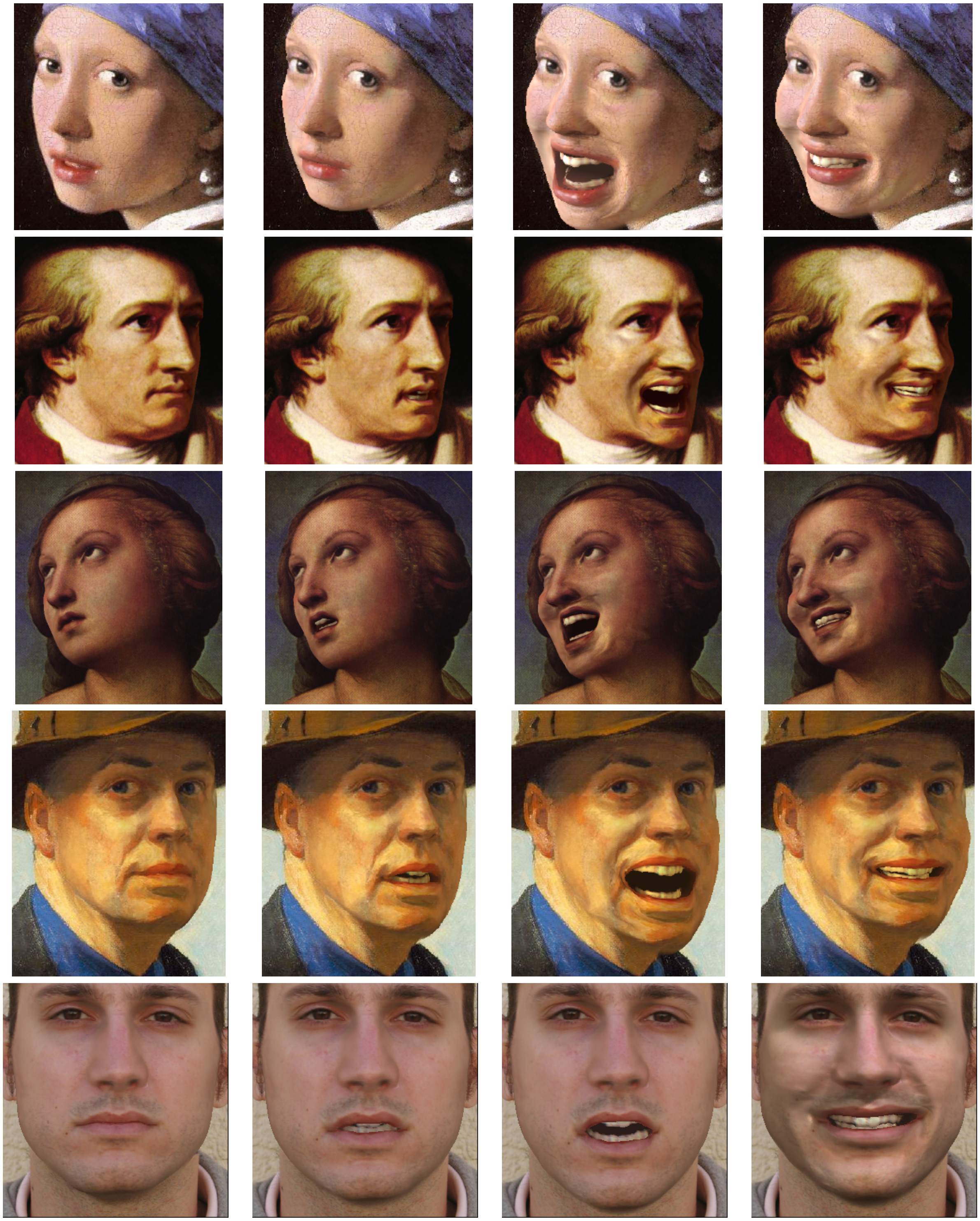

Figure 4: Reconstructed from the original images (left column), 3D shape can be modified automatically to form different mouth configurations. The paintings are Vermeer's "Girl with a Pearl Earring", Tischbein's Goethe, Raphael's St. Catherine, and Edward Hopper's self-portrait. The bottom left image is a digital photograph. The wrinkles are not caused by texture, but entirely due to illuminated surface deformations. In the bottom-right image, they are emphasized by more directed illumination. Teeth are transferred from 3D scans (Figure 3). The open mouth in Vermeer's painting was closed by our algorithm automatically by projecting the reconstructed shape vector on the subspace of neutral faces (top row, second image).

Figure 5: In a virtual try-on for hairstyles, the customer (top, left) provides a photograph, and selects target images of hairstyles (top, right). Then, the system creates synthetic images of the customers face pasted into these hairstyles.

the silhouette of the face are not blended with the background, but drawn as sharp edges. When the novel face is smaller or more skinny than the original face, the original silhouette would be visible next to the contour of the 3D model. In order to prevent this from happening, we have proposed a simple algorithm that removes the original silhouette line from the background image. In that image, the algorithm reflects pixels from outside of the face across the silhouette to the inside (Figure 7, see [3] for details.) This can be done automatically, because the original contour line is known from the projection of the reconstructed 3D model of the original face.

In some images, strands of hair or other objects occlude part of the facial surface. For these cases, we have proposed a simple compositing algorithm. Based on pixel luminance and with some additional, manual interaction, an alpha map for the foreground structures is created, and with this alpha map, the original image is composited on top of the new face (Figure 7.)

The algorithm has a number of applications in image processing, for example in consumer software for digital image processing, and in projects such as a virtual try-on for hairstyles (Figure 5.) If applied to image sequences, it could also be interesting for video post-processing.

6. LEARNING-BASED MODIFICATION OF HIGH-LEVEL ATTRIBUTES

Interactive control of perceptually meaningful features of faces, such as skinny versus obese appearance, involves global changes of face shape and texture, unlike the local changes in pixel values or vertex positions in 2D image processing and 3D mesh editing, respectively. One way of providing such tools would be to collect a set of deformation patterns, sim-

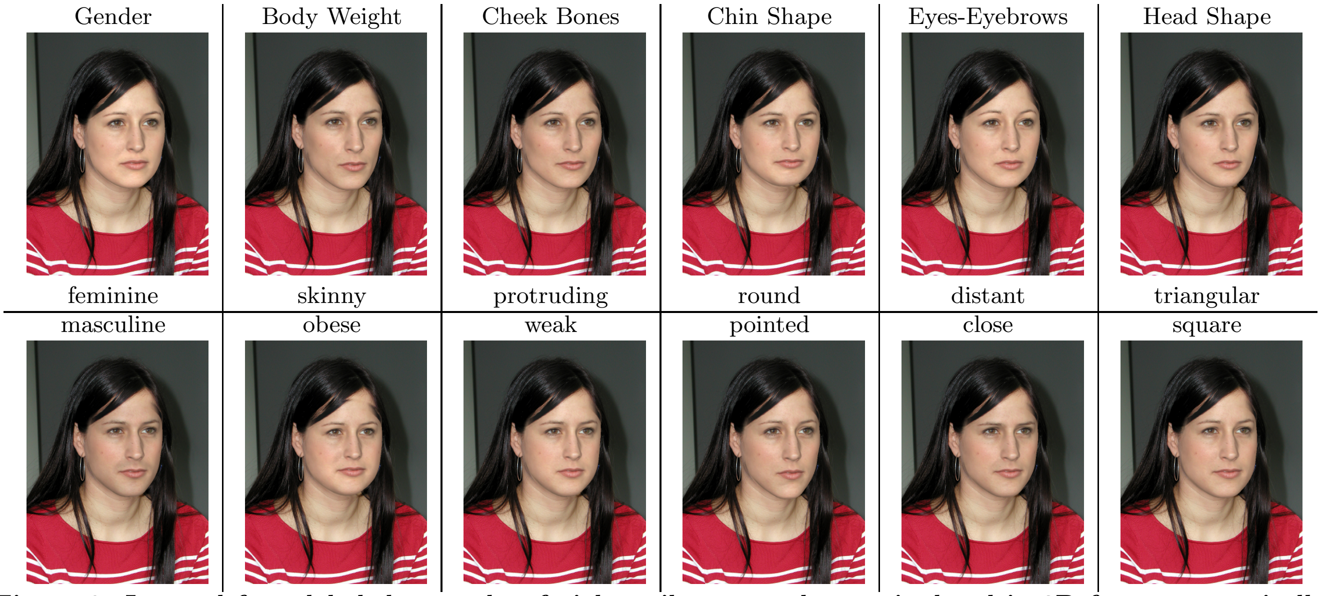

Figure 6: Learned from labeled examples, facial attributes can be manipulated in 3D faces automatically, and in combination with 3D shape reconstructions, manipulations can be performed within given portraits at any pose and illumination. Attributes in regions such as the eyes are treated as global shape vectors, which reveals correlations between different regions of the face observed in the data. For example, the distance between eyes and eyebrows turns out to be correlated with chin shape (fifth column). This correlation can be suppressed by setting vector components of $\boldsymbol{a}$ to zero outside of the region of interest, which makes the changes local.

ilar to the facial expressions in Section 4, that are manually defined, and apply those to novel faces automatically within the framework of the Morphable Model. However, modeling such deformations would be a challenging task that requires careful observation of human anatomy, and artistic skill to implement these observations on a prototype face. Instead, we propose an approach that only involves human *ratings* of attributes, such as skinniness, rather than any kind of *description*. Rating is a much easier task for humans, and it can be performed even on the most subjective attributes, such as attractiveness of faces. Based on ratings of a set of example data (shape vectors or texture vectors of faces), we compute a vector that can be added to or subtracted from a given face to change the attribute, while all other attributes and the individual characteristics of the face are left unchanged.

Let $\boldsymbol{x}_i$, $i = 1...m$, be a set of sample vectors (shape or texture), and $b_i \in \mathbb{R}$ be the ratings of a given attribute for these 3D faces. b_i can be either given as ground truth, e.g. for gender, or based on subjective ratings by the user. Our approach is to estimate a function f that assigns attribute values to faces, and then follow the gradient of this function in order to achieve a given change in attribute at a minimal, most plausible change in appearance [4, 1]. Given the limited set of data, we choose a linear regression for f, and minimize the least squares error

$$E = \sum_{i=1}^{m} (f(\boldsymbol{x}_i) - b_i)^2. \tag{12}$$

It can be shown that the gradient $\boldsymbol{a}$ of the optimal function f depends on an appropriate definition of the distance measure in face space. A perceptually meaningful distance measure is given by the probability distribution of faces in terms of PCA [1]. Then, the optimal vector for changing attributes turns out to be a simple weighted sum of the example data (for details, see [1]):

$$\boldsymbol{a} = \frac{1}{m} \sum_{i=1}^{m} b_i \boldsymbol{x}_i. \tag{13}$$

Adding multiples $\boldsymbol{x} \mapsto \boldsymbol{x} + \lambda \boldsymbol{a}$, with $\lambda \in \mathbb{R}$, will change facial attributes in the desired way and leave all characteristics that are uncorrelated with the attribute unchanged. We would like to point out that the linear attribute function f is only a first order approximation of the correct, non-linear function. Still, our results indicate that for many attributes, this approximation provides an effective tool for face modeling (Figure 6.)

7. CONCLUSION

We have presented a framework for high-level manipulation of faces in existing image material. Our approach exploits prior knowledge about classes of objects that is learned automatically from a dataset of examples. We have shown results only for the class of human faces, but it can be extended to other relevant classes of objects, such as full human bodies and animals or perhaps even vehicles or buildings. The approach is very general in terms of the imaging conditions due to the internal 3D representation of the object class. We believe that the link between image-based and 3D-based image processing is a promising strategy to pursue in the future. The overall system, as it has been described in this paper, is an example of how computer graphics is adapted to humans in terms of content, users and viewers.

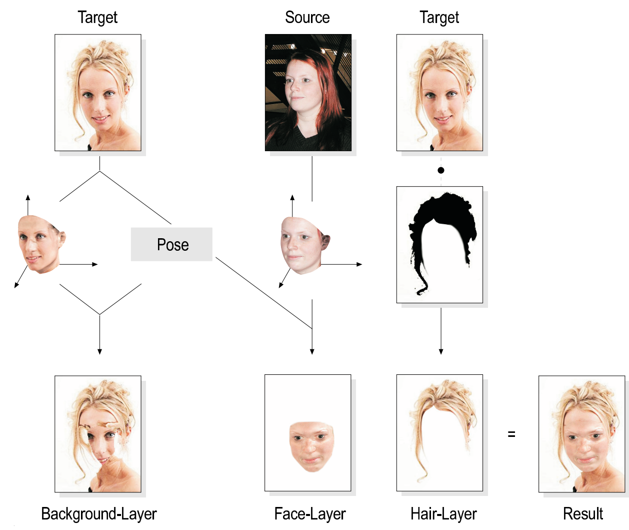

Figure 7: For transferring a face from a source image (top, center) to a target (top, left), the algorithm builds a composite of three layers (bottom row): An estimate of 3D shape and scene parameters ("pose") from the target image helps to remove the contour of the original face in the background (left). The scene parameters of the target are also used to render the 3D face reconstructed from the source image (center column). A semi-automatic segmentation of hair defines the transparency mask for the hairstyle in the top layer (right column).

8. ACKNOWLEDGMENTS

The author would like to thank Thomas Vetter, Curzio Basso, Barbara Knappmeyer, Niko Troje, Heinrich Bülthoff, Tomaso Poggio, Kristina Scherbaum and Hans-Peter Seidel for their input to the projects summarized in this paper.

9. REFERENCES

[1] V. Blanz, I. Albrecht, J. Haber, and H.-P. Seidel. Creating face models from vague mental images. In E. Gröller and L. Szirmay-Kalos, editors, *Computer Graphics Forum, Vol. 25, No. 3 EUROGRAPHICS 2006*, Vienna, Austria, 2006.

[2] V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating faces in images and video. In P. Brunet and D. Fellner, editors, *Computer Graphics Forum, Vol. 22, No. 3 EUROGRAPHICS 2003*, pages 641–650, Granada, Spain, 2003.

[3] V. Blanz, K. Scherbaum, T. Vetter, and H.-P. Seidel. Exchanging faces in images. In M.-P. Cani and M. Slater, editors, *Computer Graphics Forum, Vol. 23, No. 3 EUROGRAPHICS 2004*, pages 669–676, Grenoble, France, 2004.

[4] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Computer Graphics Proc. SIGGRAPH'99*, pages 187–194, 1999.

[5] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(9):1063– 1074, 2003.

[6] M. Burns, J. Klawe, S. Rusinkiewicz, A. Finkelstein, and D. DeCarlo. Line drawings from volume data. In *Computer Graphics Proceedings SIGGRAPH 2005*, pages 512–518, 2005.

[7] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley & Sons, New York, 2nd edition, 2001.

[8] A. Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In *Computer Graphics Proceedings SIGGRAPH 1998*, pages 453–460, 1998.

[9] D. Liebowitz, A. Criminisi, and A. Zisserman. Creating architectural models from images. In *Proc. of EuroGraphics*, volume 18, pages 39–50, 1999.

[10] V. Ostromoukhov. Digital facial engraving. In *Computer Graphics Proceedings SIGGRAPH 1999*, pages 417–424, 1999.

[11] F. I. Parke and K. Waters. *Computer Facial Animation*. A K Peters, Wellesley, Massachusetts, 1996.

[12] S. N. Pattanaik, J. Tumblin, H. Yee, and D. P. Greenberg. Time-dependent visual adaptation for fast realistic image display. In *Computer Graphics Proceedings SIGGRAPH 2000*, pages 47–54, 2000.

[13] D. Terzopoulos and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(6):569–579, 1993.

[14] T. Vetter and T. Poggio. Linear object classes and image synthesis from a single example image. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):733–742, 1997.

[15] D. Vlasic, M. Brand, H. Pfister, and J. Popovic. Face transfer with multilinear models. In *Computer Graphics Proc. SIGGRAPH'05*, pages 426–433, 2005.

[16] A. Yoshida, V. Blanz, K. Myszkowski, and H.-P. Seidel. Perceptual evaluation of tone mapping operators with real-world sceness. In B. E. Rogowitz, T. N. Pappas, and S. J. Daly, editors, *Human Vision and Electronic Imaging X, IS&T/SPIE's 17th Annual Symposium on Electronic Imaging (2005)*, volume 5666 of *SPIE Proceedings Series*, pages 192–203, San Jose, USA, January 2005.