EUROGRAPHICS 2004 / M.-P. Cani and M. Slater
(Guest Editors)

Volume 23 (2004), Number 3

Exchanging Faces in Images

Volker Blanz[1], Kristina Scherbaum[1], Thomas Vetter[2] and Hans-Peter Seidel[1]

[1] Max-Planck-Institut für Informatik, Saarbrücken, Germany
[2] University of Basel, Departement Informatik, Basel, Switzerland

Abstract

Pasting somebody's face into an existing image with traditional photo retouching and digital image processing tools has only been possible if both images show the face from the same viewpoint and with the same illumination. However, this is rarely the case for given pairs of images. We present a system that exchanges faces across large differences in viewpoint and illumination. It is based on an algorithm that estimates 3D shape and texture along with all relevant scene parameters, such as pose and lighting, from single images. Manual interaction is reduced to clicking on a set of about 7 feature points, and marking the hairline in the target image. The system can be used for image processing, virtual try-on of hairstyles, and face recognition. By separating face identity from imaging conditions, our approach provides an abstract representation of images and a novel, high-level tool for image manipulation.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Animation

1. Introduction

The transfer of faces between given images that are taken from different viewpoints and at different illuminations is a novel type of image manipulation. Manipulations in images are often performed with traditional photographic retouche or with digital processing software. However, these methods cannot transfer faces if the images are taken from different viewpoints. Differences in illumination make the task even more difficult and can only be compensated convincingly by skilled artists. In the image domain, 3D rotations give rise to very complex transformations that are difficult to model [BP96]. Rotations around large angles involve occlusions and make previously unseen surface regions visible. We therefore propose a more flexible approach that represents the face and the image parameters in 3D.

The 3D shape of faces has been estimated from single or multiple images in several approaches in face recognition, modeling and animation [LMT98, FM99, BV99, ZC00, GBK01, SK01]. A system that renders the same face back into the original image or video after some modifications has been presented for facial animation [BBPV03]. We extend that approach to solve the more general problem of pasting a face into another person's image.

Embedding 3D objects into the scene context of images or video is still a challenging task that has not been fully explored. Most work in 3D Computer Graphics, especially in the field of face modeling and animation, renders 3D objects only in isolation, or as a part of an entirely virtual scene. Rendering them into scenes defined by 2D data requires a very precise estimate of the 3D scene geometry, such as the camera parameters and the objects' position and orientation in space [DTM96, LCZ99]. Moreover, the illumination of the the synthetic objects has to be consistent with the scene context. Even though rendering is image synthesis, finding the appropriate rendering parameters is a difficult problem in image analysis and computer vision. We solve this problem in an analysis-by-synthesis technique. Unlike other methods for estimation of scene geometry [Fau93], our algorithm can be applied to unknown faces shown in single images.

Our system is based on an algorithm that estimates a textured 3D face model from a single image, along with all relevant scene parameters, such as 3D orientation, position, focal length of the camera, and the direction and intensity of illumination [BV03]. Given a set of about 7 feature points that are manually defined by the user in an interactive interface, the algorithm automatically fits a Morphable Model of 3D faces to the image and optimizes all model parameters. The Morphable Model captures the range of possible shapes

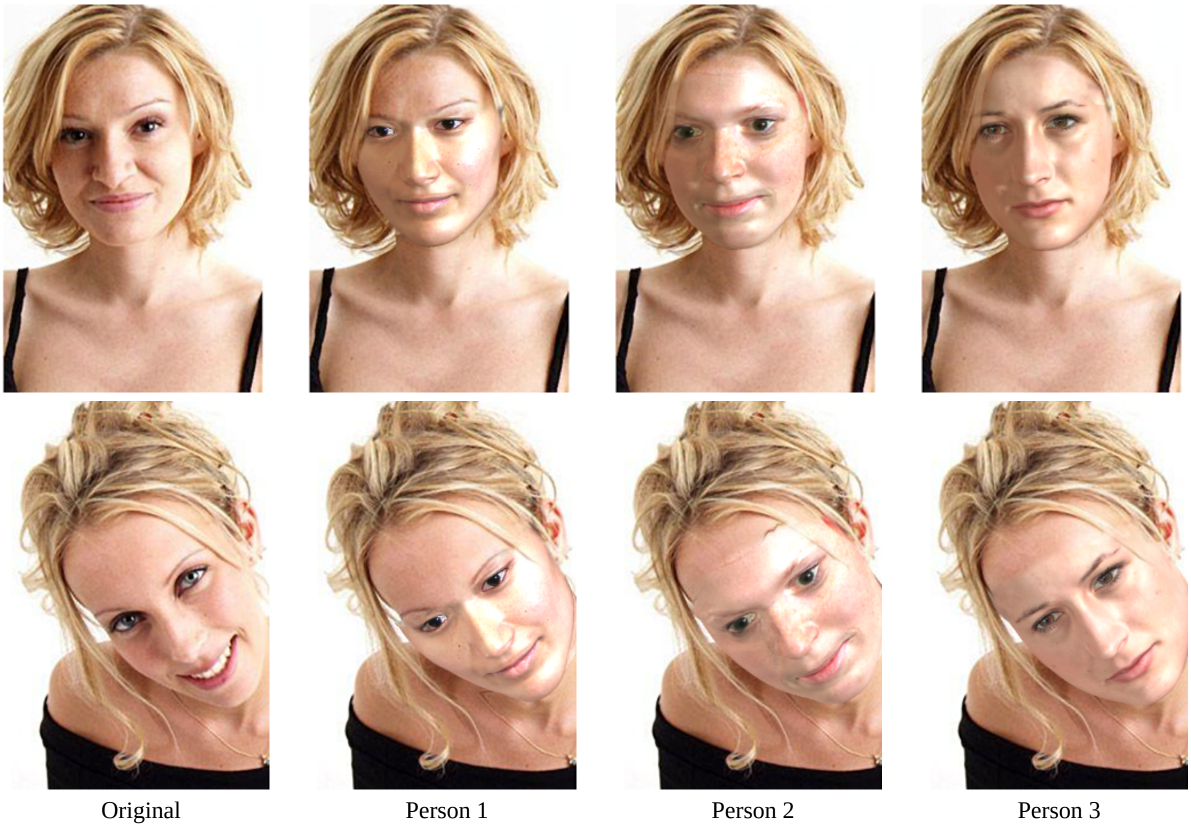


Figure 1: *Exchanging Identity: After processing the original target images (left), which includes manual clicking of features, automated 3D reconstruction and manual segmentation of foreground hair, any other face can be inserted automatically.*

and textures observed in a dataset of 3D scans. For transferring a face from a source image into a target image, we apply the fitting algorithm to both. In the source image, we are interested in the reconstructed 3D face model, and from the target image we mainly use the scene parameters.

Two properties of our framework are crucial for exchanging faces: First, the algorithm for 3D face reconstruction uses a strict and explicit separation between face identity (shape and texture) and scene parameters. Second, the Morphable Model of 3D faces is consistently aligned and illuminated. Therefore, reconstructed faces are interchangeable, and they are guaranteed to fit into the novel scene.

The face model used in this paper does not include hair. Rendering hair has been an active field in computer graphics, and impressive results have been achieved (for an overview, see [MTHK00]). In our application, however, it would be necessary to reconstruct hair from images, which is still an unsolved problem. We therefore restrict our system to exchange only the facial surface from the source to the target image, while the hairstyle of the target is retained. In fact, this procedure is very adequate for at least two applications: virtual try-on for hairstyles, where the customer can see his or her own face in a photograph of a hairstyle, and face recognition, where hairstyle is a non-diagnostic feature that should be ignored in order to avoid being misled by disguises.

In Section 2 and 3, we briefly summarize the concept of a Morphable Model and the algorithm for 3D reconstruction. In Section 4 and 5, we present the algorithm for exchanging faces in images. Section 6 introduces a system for virtual try-on of hairstyles, and Section 7 discusses results that have been achieved with this framework in face recognition.

2. A Morphable Models of 3D Faces

The Morphable Model of 3D faces[VP97, BV99] is a vector space of 3D shapes and textures spanned by a set of examples. Derived from 200 textured *Cyberware* (TM) laser scans, 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

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

is a realistic face if $\mathbf{S}$, $\mathbf{T}$ are within a few standard deviations from their averages. In the conversion of the laser scans into

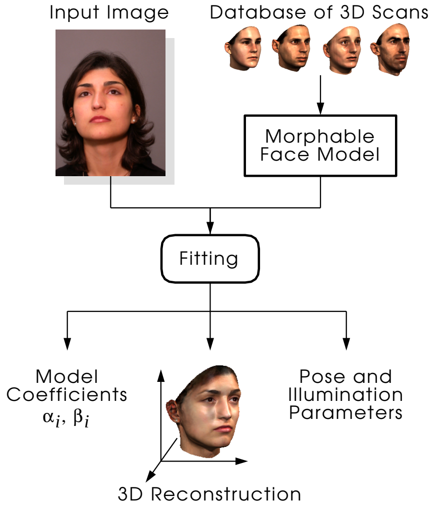


Figure 2: *Fitting the Morphable Model to an image produces not only a 3D reconstruction, but also model coefficients α_i, β_i, and an estimate of head orientation, position and illumination.*

shape and texture vectors $\mathbf{S}_i$, $\mathbf{T}_i$, it is essential to establish dense point-to-point correspondence of all scans with a reference face to make sure that vector dimensions in $\mathbf{S}$, $\mathbf{T}$ describe the same point, such as the tip of the nose, in all faces. Dense correspondence is computed automatically with an algorithm derived from optical flow [BV99].

Each vector $\mathbf{S}_i$ is the 3D shape, stored in terms of x, y, z-coordinates of all vertices $k \in \{1, \ldots, n\}$, $n = 75972$ of a 3D mesh:

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

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

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

Finally, we perform a Principal Component Analysis (PCA, see [DHS01]) to estimate the probability distributions of faces around their averages $\bar{\mathbf{s}}$ and $\bar{\mathbf{t}}$, and we replace the basis vectors $\mathbf{S}_i$, $\mathbf{T}_i$ in Equation (1) by an orthogonal set of m eigenvectors $\mathbf{s}_i$, $\mathbf{t}_i$:

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

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

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

Unlike earlier systems [BV99], the starting values of ρ are no longer estimated by the user: 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 feature point coordinates [BV03]: 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

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

that is added to 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 [DHS01]), 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. The full cost function is

$$E = \frac{1}{\sigma_I^2} E_I + \frac{1}{\sigma_F^2} E_F + \sum_i \frac{\alpha_i^2}{\sigma_{S,i}^2} + \sum_i \frac{\beta_i^2}{\sigma_{T,i}^2} + \sum_i \frac{(\rho_i - \bar{\rho}_i)^2}{\sigma_{R,i}^2}. \tag{7}$$

The standard deviations $\sigma_{S,i}$ and $\sigma_{T,i}$ are known from PCA of shapes and textures. $\bar{\rho}_i$ are the standard starting values for ρ_i, and $\sigma_{R,i}$ are ad–hoc estimates of their standard deviations.

The cost function (7) can be derived from a Bayesian approach that maximizes the posterior probability of α, β and ρ, given $\mathbf{I}_{input}$ and the feature points [BV99, BV03]. E_I is related to independent Gaussian noise with a standard deviation σ_I in the pixel values, and the regularization terms are

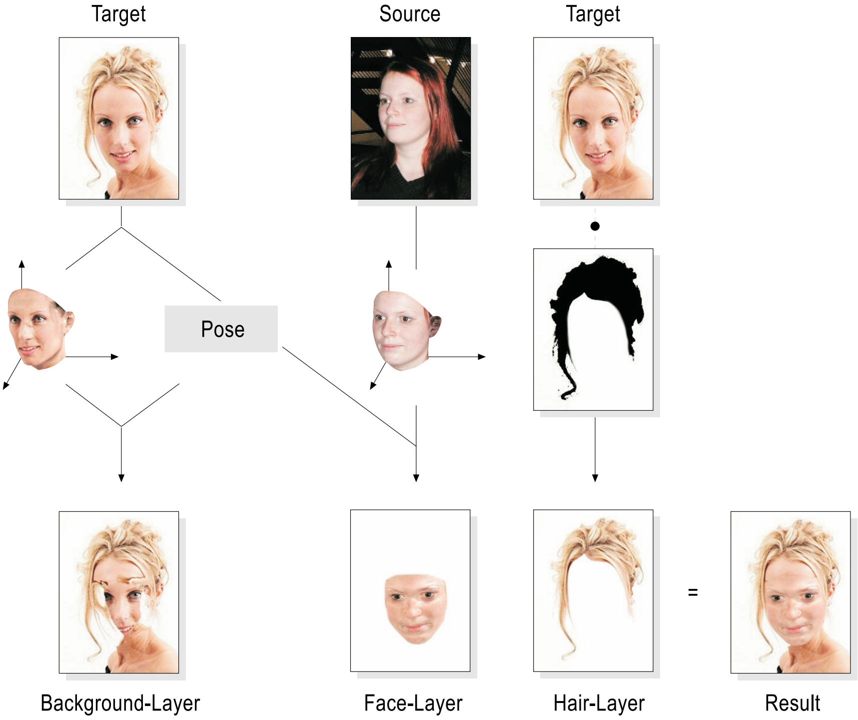


Figure 3: *For transferring a face from a source image (top, center) to a target (top, left), the algorithm builds a composite of three layers (bottom row): An estimate of 3D shape and scene parameters ("pose") from the target image helps to remove the contour of the original face in the background (left). 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).*

derived from the prior probabilities. The system performs an optimal tradeoff between minimizing E_I and achieving a plausible result.

The optimization is performed with a Stochastic Newton Algorithm [BV03]. The fitting process takes 4.5 minutes on a 2GHz Pentium 4 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 from the image, wherever it is visible, by an illumination-corrected texture extraction algorithm [BV99]. 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. Due to facial symmetry, we reflect texture elements that are visible on one and occluded on the other side of the face.

4. Exchanging Faces

The main step in exchanging faces is to render the face that was reconstructed from the source image with the rendering parameters that were estimated from the target image. Our approach makes sure that the new face fits precisely into the target image in terms of geometry and illumination without any further adaptation. In other words, 3D shapes and textures are perfectly interchangeable. In the following, we explain why this is guaranteed by the specific properties of the Morphable Model and the reconstruction algorithm (Section 3).

Having applied the fitting algorithm to both the source and the target image, we have estimates of 3D shape, texture and rendering parameters for both. A sufficient criterion for a successful exchange is that both 3D shapes are aligned to each other in 3D, and both textures have similar illumination.

When the Morphable Model was formed, the 3D faces were aligned with respect to 3D rotation and translation automatically with 3D Absolute Orientation [HS92]. As a result, all linear combinations will be aligned consistently. The heads are scaled in their natural size, so the dataset reflects the natural variation in head size, and reconstructed heads may differ in size. However, we have experienced little variation, which indicates that the scaling parameters in the fitting procedure (focal length of the camera and viewing distance) compensate for most differences. As a consequence, we do not need to normalize heads in size after reconstruction.

Most textures in the dataset are recorded with the same illumination setup, so linear combinations reflect the natural variation of skin complexion. The illumination setup was predominantly ambient, reducing the shading effects on the skin. Reconstructing texture from images is locally ambiguous since dark pixels in the image might be due to dark skin or dim illumination. The global error function (7), however, includes regions such as the eyes that are bright in all faces, and therefore we found the reconstructed textures to be close to the persons' true skin complexions. The prior probability involved in the optimization produces a plausible tradeoff between the variations in skin color and illumination.

The final step of the reconstruction algorithm is illumination-corrected texture extraction. By inverting the effect of illumination, this removes specular highlights and shading, so the texture can be re-illuminated in the new image. The final textures are still interchangeable: When the illumination was estimated, textures were limited to linear combinations of examples. The estimated texture and illumination produce pixel values $\mathbf{I}_{model}$ that will, in general, be similar to $\mathbf{I}_{input}$. Inverting the illumination on $\mathbf{I}_{input}$ will therefore produce a result that is close to the span of examples in terms of color and brightness.

5. Compositing

The final image is composited from three layers (Figure 3): The scene around the face is left unchanged, so we can use the original target image as a background for rendering. The new, exchanged face is rendered in front of the original face as a second layer. Strands of hair that are in front of the face have to be added in a third layer. In the next sections, we describe how these layers are prepared.

5.1. Background Layer

The background image shows the scene of the target image, and the original persons' face, hair and body. The original face will be covered by the novel face in the next layer. However, the novel face may be smaller than the original, leaving parts of the original visible and producing double contours.

We solve this problem by a background continuation method [BBPV03] that is based on a reflection of pixels beyond the original contour into the face area. The original contour is known, since we applied the reconstruction algorithm to the target image. In removing the original facial contour, we infer the background behind the original face from surrounding pixels.

5.2. Face Layer

Consisting of the face, the ears, most part of the forehead and the upper half of the neck, the Morphable Face Model can replace most of the skin in the image by the new person's data. The silhouette of this region can be categorized in three types:

1. Occluding contours, such as the profile line in side views, are defined by the 3D shape of the surface. They are rendered as sharp edges.
2. Boundaries of hair regions that occlude the skin, for example on the forehead or the cheeks along strands of hair, are produced by the hair layer described in the next section.
3. Mesh boundaries at the neck and the forehead where the skin would, in fact, continue. These boundaries should not be visible in the images. We produce a smooth transition between the face layer and the background, using alpha blending with a mask that was predefined on the reference shape of the Morphable Model.

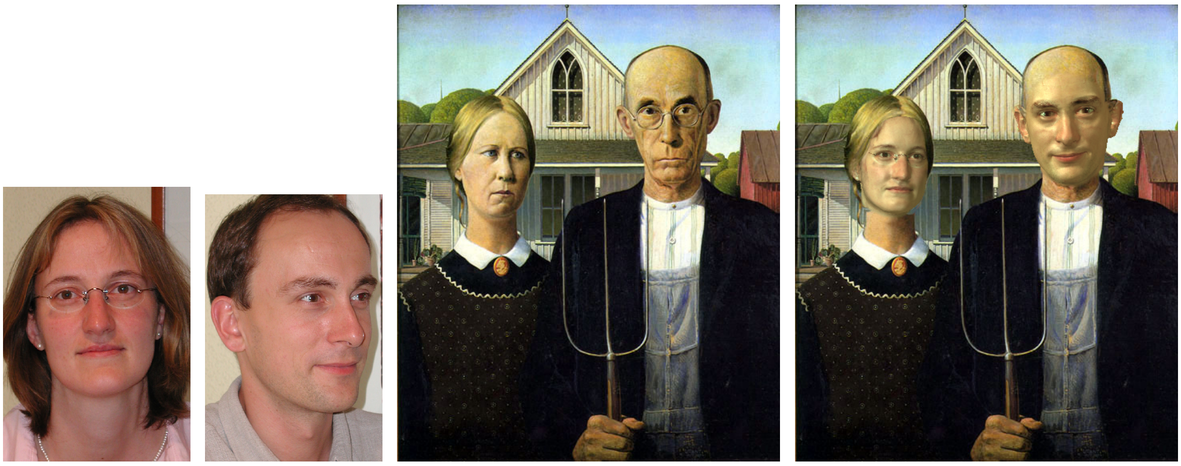
The face layer contains the upper part of the neck only, since the flexibility of the neck is not captured in the Morphable Model.

In some source images, skin is partly covered by hair. This hair would be mapped on the face as a texture, and rendered into the novel images. The user can either mask these regions in the reconstruction phase or in rendering. Masks are defined on the surface parameterization, so masks for typical situations, such as the forehead covered by the fringe or the ears covered by long hair, can be predefined and reused across individuals.

5.3. Hair Layer

If hair covers part of the face in the target image, we define a hair layer that will be drawn in front of the face. The hair layer of a particular target image can be used for all faces rendered into this image. The hair layer is identical to the original target image, but has alpha values for transparency assigned to each pixel. Automated classification of pixels into skin and hair is a difficult task. We currently perform this classification manually by defining alpha values for opacity.

For dark hair, a semi-automatic technique defines the opacity: We automatically make dark pixels opaque (hair), and bright regions transparent (skin). This simple mapping from luminance to alpha values gives a precise result along strands of hair. The alpha values require some manual postprocessing: Our criterion is not valid on the entire face, since the hair layer should be transparent at the eyebrows, and opaque at specular reflections of hair. After our algorithm has pre-segmented the boundary, this is easy to correct by the user. For blond hair, our segmentation often detects strands of hair only partially, based on the dark pixels on skin or hair caused by self-shadowing. In this case the detected regions serve as a starting point for manual segmentation.



Person 1 Person 2 Original Result

Figure 4: *Faces can be rendered into any image, such as a painting (Grant Wood: American Gothic).*

Figure 5: *The persons from Figure 4 are rendered into a movie poster.*

6. Virtual Try-On for Hairstyles

Current systems that give users a preview of their own face with different hairstyles are entirely based on 2D technology. More specifically, they are restricted to frontal views of faces. After some geometrical alignment, faces and hairstyles are composited into novel images. Due to the complex three-dimensional structure of hairstyles, it would be highly informative for users to see their face in side-views as well. Moreover, frontal views of the customer's face or the new hairstyle may be unavailable, for example if the user wishes to try a hairstyle shown in a magazine. Since most photographs are non-frontal, current systems are likely to fail. Our method provides a simple, but general and robust solution (Figures 1 and 6).

For virtual try-on, there are a two possible scenarios. In both cases, the user or customer provides an image of his or her face, which can be an arbitrary snapshot, and after labeling a set of features, obtains a 3D reconstruction. For the target image showing the hairstyle, there two alternatives:

In one setting, the user provides a photograph of a specific hairstyle, for example from a magazine. The manual processing involves labeling the feature points, and segmenting the hair for the foreground level. The face is then automatically rendered into the target image.

The second scenario requires no manual interaction during try-on after the user's face has been reconstructed. It involves a set of preprocessed hairstyles that are stored in a database. The manual processing of the hair layer for compositing has to be performed only once. Subsequently, any face can be rendered and composited with this hairstyle in a fully automated way. During try-on, the user can select a hairstyle and get the synthetic image in a fraction of a second. Users can inspect hairstyles from different viewpoints if the original hairstyles in the database were photographed from several directions. Unlike synthetically rendered hair, the hairstyle is left unchanged in the image, and will therefore always look photo-realistic.

7. Front Views from Non-Frontal Images for Face Recognition

The method described in this paper provides a powerful component for automated face recognition: An earlier version of our program has been part of the Face Recognition Vendor Test 2002 [PGM*03], which investigated a combination of 3D-based preprocessing and 10 leading commercial face recognition systems. Since all these systems use image-based methods for comparing images, and they are designed mainly for frontal views of faces, their performance drops significantly when non-frontal images are involved in the testing [PGM*03]. The systems were given a set of *gallery* images showing 87 persons in a frontal pose, and they were

Figure 6: *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.*

tested with *probe* images of the same persons with their heads rotated up, down, left or right by about 30° to 45°.

Our algorithm pre-processed the non-frontal images and generated frontal views of each probe. All frontal views were rendered into a standard image showing a person with short hair. The purpose of the standard image is to produce complete portraits with hair, shoulders and a natural background, similar to those that the systems were designed for.

In a verification task where the systems have to decide if an images shows a specific person from the gallery or not, the best verification rate observed was approximately 55% without and 85% with our system [PGM*03]. 9 out of 10 systems showed an increase of performance in this order of magnitude with the front views generated by our algorithm.

The results of the Face Recognition Vendor Test 2002 show that our algorithm can contribute to computer vision by producing synthetic images transformed into standard imaging conditions. These synthetic images reduce the variation in the input data for image-based recognition systems.

8. Conclusion

We have introduced an abstract representation of images in terms of facial identity and scene parameters specific to the image. This is achieved by explicitly separating 3D shape, texture, scene geometry and illumination parameters in image analysis. The fact that we can exchange faces in images validates our implementation of this abstract representation.

Exchanging faces across changes in viewpoint and illumination is a novel way of processing images on a high level, and has not been possible with image-based methods. With existing 3D technology, the task required the skill of an artist and many hours of manual work. Our method requires only a few simple manual processing steps, and performs the non-trivial 3D registration automatically. Compensating for all relevant imaging parameters, it is applicable to any given pair of photographs, which makes the program interesting for a wide range of applications.

Our contribution to the Face Recognition Vendor Test 2002 demonstrates that methods from Computer Graphics can be useful components in Computer Vision. Transferring technology from Computer Graphics to Vision has rarely been an issue so far, while the opposite direction of transfer has become more and more popular and fruitful in recent years. Based on analysis-by-synthesis, our approach is well suited to serve as a connection between both fields. Moreover, by mapping from 2D images to a 3D representation and back, our algorithm combines the benefits of image-based methods, such as photo-realism, with the versatility of 3D graphics.

In future work, we plan to include an algorithm for detecting facial features, and develop methods for the challenging task of hair segmentation. With these two extensions, our system would be fully automated. Finally, we are planning to exchange faces in video sequences, which involves tracking head motions. Combined with the separate manipulation of speech and head motion [BBPV03], our method then forms a complete editing tool for video post-processing.

Acknowledgements : We would like to thank all those who left their portraits to the mercy of our manipulative experiments: Lisa Jochem, Dorothee Kissel, Sandra Kruschewski, Conny Liegl, Meike Michaeli, Ewelina Orrasa, Laura Scholly, Nadja Siebertz, Carsten and Blanka Wagner, Jennifer Westcott and Akiko Yoshida. The database of 200 laser scans was collected by Niko Troje and Tordis Philipps at Max-Planck-Institut für Biologische Kybernetik, Tübingen in the group of Heinrich Bülthoff.

References

[BBPV03] BLANZ V., BASSO C., POGGIO T., VETTER T.: Reanimating faces in images and video. In *Computer Graphics Forum, Vol. 22, No. 3 EUROGRAPHICS 2003* (Granada, Spain, 2003), Brunet P., Fellner D., (Eds.), pp. 641–650.

[BP96] BEYMER D., POGGIO T.: Image representations for visual learning. *Science 272* (1996), 1905–1909.

[BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3D faces. In *Computer Graphics Proc. SIGGRAPH'99* (Los Angeles, 1999), pp. 187–194.

[BV03] BLANZ V., VETTER T.: Face recognition based on fitting a 3d morphable model. *IEEE Trans. on Pattern Analysis and Machine Intell. 25*, 9 (2003), 1063– 1074.

[DHS01] DUDA R., HART P., STORK D.: *Pattern Classification*, 2nd ed. John Wiley & Sons, New York, 2001.

[DTM96] DEBEVEC P. E., TAYLOR C. J., MALIK J.: Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. *Computer Graphics 30*, Annual Conference Series (1996), 11–20.

[Fau93] FAUGERAS O.: *Three-Dimensional Computer Vision*. Cambridge, MA: MIT Press, 1993.

[FM99] FUA P., MICCIO C.: Animated heads from ordinary images: A least squares approach. *Computer Vision and Image Understanding 75*, 3 (1999), 247–259.

[GBK01] GEORGHIADES A., BELHUMEUR P., KRIEGMAN D.: From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. on Pattern Analysis and Machine Intelligence 23*, 6 (2001), 643–660.

[HS92] HARALICK R., SHAPIRO L.: *Computer and robot vision*, vol. 2. Addison-Wesley, Reading, Ma, 1992.

[LCZ99] LIEBOWITZ D., CRIMINISI A., ZISSERMAN A.: Creating architectural models from images. In *Proc. of EuroGraphics* (1999), vol. 18, pp. 39–50.

[LMT98] LEE W., MAGNENAT-THALMANN N.: Head modeling from pictures and morphing in 3D with image metamorphosis based on triangulation. In *Proc. Captech98 (Modelling and Motion Capture Techniques for Virtual Environments)* (Geneva, 1998), Springer LNAI LNCS Press, pp. 254–267.

[MTHK00] MAGNENAT-THALMANN N., HADAP S., KALRA P.: State of the art in hair simulation. In *Int. Workshop on Human Modeling and Animation, Seoul, Korea* (2000), pp. 3–9.

[PGM*03] PHILLIPS P., GROTHER P., MICHAELS R., BLACKBURN D., TABASSI E., BONE M.: *Face Recognition Vendor Test 2002: Evaluation Report*. NISTIR 6965, Nat. Inst. of Standards and Technology, 2003.

[SK01] SIM T., KANADE T.: *Illuminating the Face*. Tech. Rep. CMU-RI-TR-01-31, The Robotics Institute, Carnegie Mellon University, Sept. 2001.

[VP97] VETTER T., POGGIO T.: Linear object classes and image synthesis from a single example image. *IEEE Trans. on Pattern Analysis and Machine Intelligence 19*, 7 (1997), 733–742.

[ZC00] ZHAO W., CHELLAPPA R.: SFS based view synthesis for robust face recognition. In *Int. Conf. on Autom. Face and Gesture Recognition* (2000), pp. 285–292.