A short version of this paper appeared in: International Journal of Computer Vision, 28:2 (1998) 103-116.

Synthesis of novel views from a single face image

THOMAS VETTER

vetter@mpik-tueb.mpg.de

Max-Planck-Institut für Biologische Kybernetik,Spemannstr. 38, 72076 Tübingen, Germany

Abstract. Images formed by a human face change with viewpoint. A new technique is described for synthesizing images of faces from new viewpoints, when only a single 2D image is available. A novel 2D image of a face can be computed without explicitly computing the 3D structure of the head. The technique draws on a single generic 3D model of a human head and on prior knowledge of faces based on example images of other faces seen in different poses. The example images are used to "learn" a pose-invariant shape and texture description of a new face. The 3D model is used to solve the correspondence problem between images showing faces in different poses.
The proposed method is interesting for view independent face recognition tasks as well as for image synthesis problems in areas like teleconferencing and virtualized reality.

Keywords: Image synthesis, face recognition, rotation invariance, flexible templates

1. Introduction

Given only a driver's license photograph of a person's face, can one infer how the face might look like from a different viewpoint? The three-dimensional structure of an object determines how its image changes with a variation in viewpoint. With viewpoint changes, some previously visible regions of the object become occluded, while other previously invisible regions become visible. Additionally, the arrangement or configuration of object regions that are visible in both views may change. Accordingly, to synthesize a novel view of an object, two problems must be addressed and resolved. First, the visible regions that the new view shares with the previous view must be redrawn at their new positions. Second, regions not previously visible must be generated or synthesized. It is obvious that this latter problem is unsolvable without prior assumptions. For human faces, which share a common structure, such prior knowledge can be obtained through extensive experience with other faces.
The most direct and general solution for the synthesis of novel views of a face from a single example image is the recovery of the three-dimensional structure of the face. This three-dimensional model can be rotated artificially and would give the correct image for the all points visible in the example image (i.e. the one from which the model was obtained). However, without additional assumptions, the minimal number of images necessary to reconstruct a face using localized points is three (Huang and Lee, 1989). Even with the additional assumption that a face is bilaterally symmetric at least two images are required (Rothwell et al., 1993; Vetter and Poggio, 1994). Shape recovery methods such as shape-from-shading that can, in principle, work with a single image are not of much help for this problem. While shape from shading algorithms have been applied in pre-

vious work to recover the surface structure of a face (Horn, 1987), the inhomogeneous reflectance properties of faces make surface integration over the whole face imprecise and questionable. Additionally, the fact that the face regions visible from a single image are insufficient to obtain the three-dimensional structure of the whole head makes it clear that the task of synthesizing new views from a single image of a face, cannot be solved without prior assumptions about the structure and appearance of faces in general.

Models that have been proposed previously to learn a face model from images can be subdivided into two groups: those based on the three-dimensional head structure and those considering only view- or image-dependent face models. In general, schemes that have attempted to incorporate knowledge about faces into flexible three-dimensional head models, consist of hand-constructed representations of the physical properties of the muscles and the skin of a face (Terzopoulos and Waters, 1993; Thalmann and Thalmann, 1995). To adjust such a model to a particular face, often two or more images were used (Aizawa et al., 1989; Akimoto et al., 1993). For present purposes, it is difficult to assess the usefulness of this approach, since generalization performance to new views from only a single image has never been reported. An example based approach for forming a flexible three-dimensional head models was explored by Choi et al.(1991). Here a new face is modeled as a weighted sum of given example head models, modeling the three-dimensional shape and the texture data (image intensities) separately.

In the same idea of modeling shape and texture of face images separately, two-dimensional image-based face models have been applied for the synthesis of rigid and nonrigid face transitions (Craw and Cameron, 1991; Poggio and Brunelli, 1992; Beymer et al., 1993; Cootes et al., 1995; Lanitis et al., 1995). These models generally exploit prior knowledge from example images of prototypical faces and work by building flexible image-based representations (*active shape models*) of known objects by a linear combination of labeled examples. Although all these methods differ in their labeling method they all lead to a very similar representation of images, which separates the shape and textural information in an image. These representations have been applied for the tasks of image search and recognition (Cootes et al., 1995) or synthesis (Craw and Cameron, 1991; Lanitis et al., 1995). The underlying coding of an image of a new object or face is based on linear combinations of the two-dimensional shape and texture of examples of prototypical images. The shape model developed in Lanitis et al. (1995) additionally allows rigid rotations in 3D-space of the faces depicted in the images. The shape component of a new individual face image is normalized to a standard three-dimensional orientation before it is coded by other example faces. The transformations necessary for this normalization are similar to these applied by Beymer et al. (1993) to synthesize new images of a face with a different expression or a changed viewpoint making use of only a single image. These methods of warping an image to a new expression or a new view have been already successfully applied for face recognition tasks (Lanitis et al., 1995; Beymer and Poggio, 1996).

In the contrast of the problem posed in this paper of the synthesis of a new view from a single image, these methods are limited in two ways due to their reliance on the solution of the correspondence problem across view changes. First, to establish correspondence over large changes in viewpoint is highly problematic because of the fact that occluding contours in different views of an object do not physically correspond to each other on the surface of the object. Second, the methods are fundamentally incapable of generating or synthesizing those areas or regions in the new view, which previously were not visible. For instance, a change in viewpoint of about 15° from the front to the side for a face usually leads to a complete occlusion of one ear.

To overcome these difficulties in the present work, we draw on the concept of *linear object classes*, which we have introduced recently in the context of object representations (Vetter and Poggio, 1996). This approach does not need correspondence across different viewpoints and therefore is capable of coping with larger viewpoint changes. For each specific viewpoint a separate linear image model is used where each leads to the same view independent representation of an object. In contrast to all the methods mentioned earlier, this approach does not directly transform

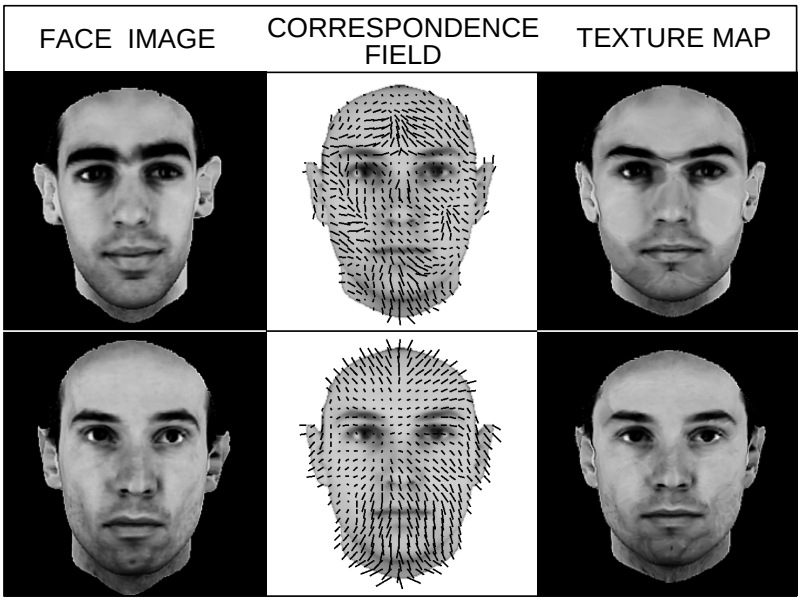

Fig. 1. Two examples of face images (left column) mapped onto a reference face using pixelwise correspondence are shown (right column). The middle column shows a superposition of the reference face and the correspondence field which was established through an optical flow algorithm. The correspondence separates the 2D-shape information captured in the correspondence field from the texture information captured in the texture mapped onto the reference face (right column).

(warp) a given image to an image showing a new view of the face. It rather codes the given image by one linear model and uses this code in an different linear model to synthesize the new view. While this basic coding scheme is advantageous for handling large viewpoint changes, however, it has some drawbacks for information not representable by the linear coding model. For instance textural details will be lost with this linear modeling approach, even when they are clearly visible in the given image. In this paper an extension to the linear object class approach is presented, which retains all its advantages without the loss of textural information.

Overview of the Approach

In the present paper, the *linear object class* approach is improved and combined with a single three-dimensional model of a human head for generating new views of a face. By using these techniques in tandem, the limitations inherent in each approach (used alone) can be overcome. Specifically, the present technique is based on the linear object class method described in (Vetter and Poggio, 1996), but is more powerful because the addition of the 3D model allows a much better utilization of the example images. The 3D-model also allows the transfer of features like moles and blemishes particular to an individual face from the given view into new synthetic views. This latter point is an important addition to the linear class approach, because it now allows for individual identifying features that are present in "non-standard" locations on a given individual face, to be transferred onto synthesized novel views of the face. This is true even when these blemishes, etc., are unrepresented in the "general experience" that the linear class model has acquired from example faces. On the other hand, the primary limitation of a single 3D head model is the well-known difficulty of representing the variability of head shapes in general, a problem that the linear class model, with its exemplar-based knowledge of faces will allow us to solve.

Another way of looking at the combination of these approaches returns us to the two-fold problem we described at the beginning of this paper. The synthesis of novel views from a single exemplar image requires the ability to redraw the regions shared by the two views, and also the ability to generate the regions of the novel face that are invisible in the exemplar view. The 3D head

model allows us to solve the former, and linear object class approach the allows us to solve the latter.

Linear Object Classes

A linear object class is defined as a 3D object class for which the 3D shape can be represented as a linear combination of a sufficiently small number of prototypical objects. Objects that meet this criterion have the following important property. New orthographic views according to uniform affine 3D transformation can be generated for any object of the class. Specifically, rigid transformations in 3D, can be generated exactly if the corresponding transformed views are known for the set of prototypes. Thus, if the example set consists of frontal and rotated views of a set of prototype faces, any rotated view of a new face can be generated from a single frontal view – provided that the linear class assumption holds.

The key to this approach is a representation of an object or face view in terms of a *shape vector* and a *texture vector* (see also Cootes et al., 1995; Beymer and Poggio, 1996). The separation of 2D-shape and texture information in images of human faces requires correspondence to be established to a single reference face. At its extreme, correspondence must be established for every pixel, between the given face image and a reference image. As noted previously, while this is an extremely difficult problem when large view changes are involved, the linear object class assumption requires correspondence only *within* a given viewpoint – specifically, the correspondence between a single view of an individual face and a single reference face image from the same view. Separately for each orientation, all example face images have to be set in correspondence to the reference face in the same pose, correspondence between different poses is not needed. This can be done off-line manually (Craw and Cameron, 1991; Cootes et al., 1995) or automatically (Beymer et al., 1993; Lanitis et al., 1995; Vetter and Poggio, 1996). Once the correspondence problem within views is solved, the resultant data can be separated into a shape and texture vector. The shape vector codes the 2D-shape of a face image as deformation or correspondence field to a reference face, which later also serves as the origin of a linear vector space. Likewise the texture of the exemplar face is coded in a vector of image intensities being mapped onto *corresponding* positions in the reference face image (see also figure 1 right column).

The Three-dimensional Head Model

The linear class approach works well for features shared by all faces (e.g. eyebrows, nose, mouth or the ears). But, it has limited representational possibilities for features particular to a individual face (e.g. a mole on the cheek). For this reason, a single 3D model of a human head is added to the linear class approach. Face textures mapped onto the 3D model can be transformed into any image showing the model in a new pose. The final "rotated" version of a given face image (i.e. including birthmarks, etc.) can be generated by applying to this new image of the 3D model the shape transformation given through the linear object class approach. This is described in more detail shortly. The three-dimensional model and its generation are described in Appendix A.

The paper is organized as follows. First, the algorithm for generating new images of a face from a single image is described. The technical details of the implementation used to realize the algorithm on grey level images of human faces are described in the *Appendix*. Under *Results* a comparison of different implementations of the generalization algorithm are shown. Two variations of the combined approach are compared with a method based purely on the linear object class as described previously (Vetter and Poggio, 1996). First, the linear class approach is applied to the parts of a face separately. The individual parts in the two reference face images were separated using the 3D-model. Second, the 3D-model was used additionally to establish pixelwise correspondence between the two reference faces images in the two different orientations. This correspondence field allows texture mapping across the view point change. Finally, the main features and possible future extensions of the technique are discussed.

2. Approach and Algorithm

In this section an algorithm is developed that allows for the synthesis of novel views of a face from from a single view of the face. For brevity, in the present paper we describe the application of the algorithm to the synthesis of a "frontal" view

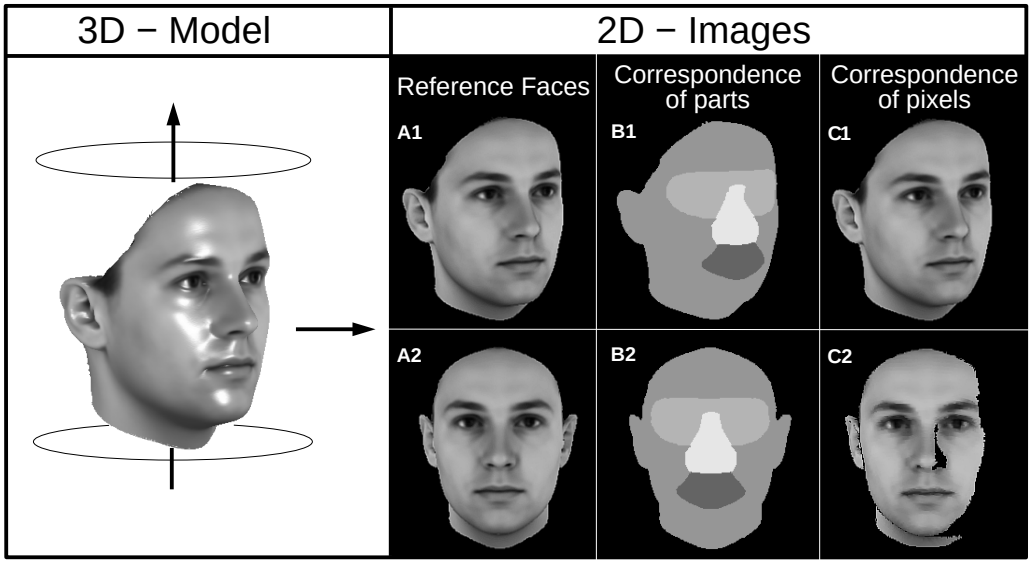

Fig. 2. A three-dimensional model of a human head was used to render the reference images (column A) for the linear shape and texture model. The model defines corresponding parts in the two images (column B) and also establishes pixelwise correspondence between the two views (column C). Such a correspondence allows texture mapping from one view ($C1$) to the other ($C2$).

(i.e., defined in this paper as the novel view) from a single "rotated" view (i.e., defined in this paper as the view 24° from frontal). It should be noted, however, that the algorithm is not at all restricted to a particular orientation of faces.

The algorithm can be subdivided into three parts (for an overview see figure 3).

- First, the texture and shape information in an image of a face are separated.
- Second, two separate modules, one for texture and one for shape, compute the texture and shape representations of a given "rotated" view of a face (in terms of the appropriate view of the reference face). These modules are then used to compute the shape and texture estimates for the new "frontal" view of that face.
- Finally the new texture and shape for a "frontal" view are combined and warped to the "frontal" image of the face.

Separation of texture and shape in images of faces: The central part of the approach is a representation of face images that consists of a separate texture vector and 2D-shape vector, each one with components referring to the same points – in this case pixels. Assuming pixelwise correspondence to a reference face in the same pose, a given image can be represented as follows: its 2D-shape will be coded as the deformation field of n selected points – in the limit of each pixel – to the reference image. So the shape of a face image is represented by a vector $\mathbf{s} = (x_1, y_1, x_2, ..., x_n, y_n)^T \in \Re^{2n}$, that is by the x, y distance or displacement of each point with respect to the corresponding point in the reference face. The texture is coded as a difference map between the image intensities of the exemplar face and its *corresponding* intensities in the reference face. Thus, the mapping is defined by the correspondence field. Such a normalized texture can be written as a vector $\mathbf{T} = (i_1, ...i_n)^T \in \Re^n$, that contains the image intensity differences i of the n pixels of the image. All images of the example set are mapped onto the reference face of the corresponding orientation. This is done separately for each rotated orientation. For real images of faces the pixelwise correspondences necessary for this mappings were computed automatically using a gradient based optical technique which was already used successfully previously on face images (Beymer et al., 1993; Vetter and Poggio, 1996). The technical details for this technique can be found in Appendix B.

Linear shape model of faces: The shape model of human faces used in the algorithm is based on the linear object class idea (the necessary and sufficient conditions are given in Vetter and Poggio, 1996) and is built on a example set of pairs of images of human faces. From each pair of images, each consisting of a "rotated" and a "frontal" view of a face, the 2D-shape vectors s^r for the "rotated" shape and s^f for the "frontal" shape are computed. Consider the three-dimensional shape of a human head defined in terms of a set of points in the three-dimensional space. The 3D-shape of the head can be represented by a vector $\mathbf{S} = (x_1, y_1, z_1, x_2, ..., y_n, z_n)^T$, that contains the x, y, z-coordinates of its n points. Assume that $\mathbf{S} \in \Re^{3n}$ is the linear combination of q 3D shapes $\mathbf{S}_i$ of *other* heads, such that: $\mathbf{S} = \sum_{i=1}^{q} \beta_i \mathbf{S}_i$. It is quite obvious that for any linear transformation R (e.g. rotation in 3D) with $\mathbf{S}^r = R\mathbf{S}$, it follows that $\mathbf{S}^r = \sum_{i=1}^{q} \beta_i \mathbf{S}_i^r$. Thus, if a 3D head shape can be represented as the weighted sum of the shapes of other heads, its rotated shape is a linear combination of the rotated shapes of the other heads with the same weights β_i.

To apply this to the 2D face shapes computed from images, we have to consider the following. A projection P from $3D$ to $2D$ with $\mathbf{s}^r = P\mathbf{S}^r$ under which the minimal number q of shape vectors necessary to represent $\mathbf{S}^r = \sum_{i=1}^{q} \beta_i \mathbf{S}_i^r$ and $\mathbf{s}^r = \sum_{i=1}^{q} \beta_i \mathbf{s}_i^r$ does not change, it allows the correct evaluation of the coefficients β_i from the images. Or in other words, the dimension of a three-dimensional linear shape class is not allowed to change under a projection P. Assuming such a projection, and that s^r, a 2D shape of a given "rotated" view, can be represented by the "rotated" shapes of the example set s_i^r as

$$s^r = \sum_{i=1}^{q} \beta_i \mathbf{s}_i^r, \tag{1}$$

then the "frontal" 2D-shape $\mathbf{s}^f$ to a given $\mathbf{s}^r$ can be computed without knowing $\mathbf{S}$ using β_i of equation (1) and the other $\mathbf{s}_i^f$ given through the images in the example set with the following equation:

$$\mathbf{s}^f = \sum_{i=1}^{q} \beta_i \mathbf{s}_i^f. \tag{2}$$

In other words, a new 2D face shape can be computed without knowing its three-dimensional structure. It should be noted that no knowledge of correspondence between equation (1) and equation (2) is necessary (rows in a linear equation system can be exchanged freely).

Texture model of faces: In contrast to the shape model, two different possibilities for generating a "frontal" texture given a "rotated" texture are described. The first method is again based on the linear object class approach and the second method uses a single three-dimensional head model to map the texture from the "rotated" texture onto the "frontal" texture. The linear object class approach for the texture vectors is equivalent to the method described earlier for the 2D-shape vectors. It is assumed that a "rotated" texture $\mathbf{T}^r$ can be represented by the q "rotated" textures $\mathbf{T}_i^r$ computed from the given example set as follows:

$$\mathbf{T}^r = \sum_{i=1}^{q} \alpha_i \mathbf{T}_i^r. \tag{3}$$

It is assumed further that the new texture $\mathbf{T}^f$ can be computed using α_i of equation (3) and the other $\mathbf{T}_i^f$ given through the "frontal" images in the example set by the following equation:

$$\mathbf{T}^f = \sum_{i=1}^{q} \alpha_i \mathbf{T}_i^f. \tag{4}$$

In contrast to the linear shape model, using the coefficients α_i of equation (3) in equation (4) holds not in general. The assumption of a separate linear texture model, which is independent of (1) is strictly only valid for textures which are a function of albedo only.

The three-dimensional head model: Whereas the linear texture approach is satisfactory for generating new "frontal" textures for regions not visible in the "rotated" texture, it is not satisfactory for the regions visible in both views. The linear texture approach is hardly able to capture or represent features which are particular to an individual face (e.g. freckles, moles or any similar distinct aspect of facial texture). Such features ask for a direct mapping from the given "rotated" texture onto the new "frontal" texture. However, this requires pixelwise correspondence between the two views (see Beymer et al., 1993).

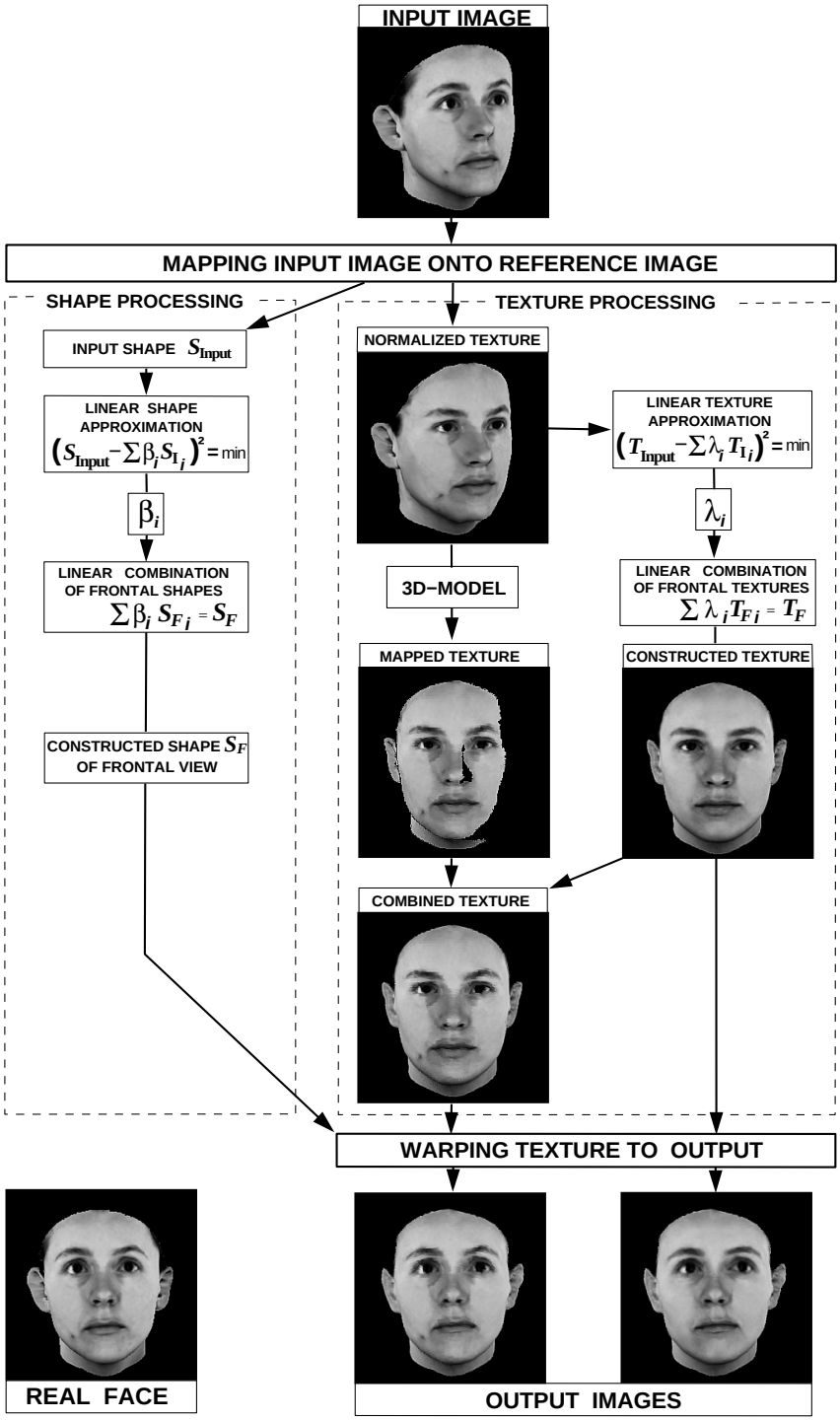

Fig. 3. Overview of the algorithm for synthesizing a new view from a single input image. After mapping the input image onto a reference face in the same orientation, texture and 2D-shape can be processed separately. The example based linear face model allows the computation of 2D-shape and texture of a new "frontal" view. Warping the new texture along the new deformation field (coding the shape) results in the new "frontal" views as output. In the lower row on the right the result purely based on the linear class approach applied to parts is shown, in the center the result with texture mapping from the "rotated" to the "frontal" view using a single generic 3D model of a human head. On the bottom left the real frontal view of the face is shown.

Since all textures are mapped onto the reference face, it is sufficient to solve the correspondence problem across the the viewpoint change for the reference face only. A three-dimensional model of an object intrinsically allows the exact computation of a correspondence field between images of the object from different viewpoints, because the three-dimensional coordinates of the whole object are given, occlusions are not problematic and hence the pixels visible in both images can be separated from the pixels which are only visible from one viewpoint.

A single three-dimensional model of a human head is incorporated into the algorithm for three different processing steps. For more details about the three-dimensional head model and its generation see Appendix A.

1. The reference face images used for the formation of the linear texture and 2D-shape representations were rendered from the 3D-model under ambient illumination conditions (see figure 2A).
2. The 3D-model was manually divided into separate parts, the nose, the eye and mouth region and the rest of the model. Using the projections of these parts, the reference images for different orientations could be segmented into corresponding parts for which the linear texture and 2D-shape representation could be applied separately (see next paragraph on "The shape and texture models applied to parts" and also figure 2B).
3. The correspondence field across the two different orientations was computed for the two reference face images based on the given 3D-model. So the visible part of any texture, mapped onto the reference face in one orientation, can now be mapped onto the reference face in the second orientation (see figure 2C and figure 3).

To synthesize a complete texture map on the "frontal" reference face for a new view, (i.e., the regions invisible in the exemplar view are lacking), the texture of the region visible in both views, which has been obtained through direct texture mapping across the viewpoint change, is merged with the texture obtained through the linear class approach (see figure 3). The blending technique used to merge the regions is described in detail in the Appendix D.

The shape and texture models applied to parts. To apply equations (1 – 4) to individual parts of a face, it is necessary to isolate the corresponding *areas* in the "rotated' and "frontal" reference images. Such a separation requires the correspondence between the "rotated' and "frontal" reference image or equivalent between equations (1) and (2) of the shape representation and also between equations (3) and (4) for the texture. The 3D-model, however, used for generating the reference face images determines such a correspondence immediately (for example see figure 2B) and allows the separate application of the linear class approach to parts. To generate the final shape and texture vector for the whole face, this separation adds only a few complexities to the computational process . Shape and texture vectors obtained for the different parts must be merged, which requires the use of blending techniques to suppress visible border effects. The blending technique used to merge the regions is described in detail in Appendix D.
The linear object class approach for 2D-shape and texture, as proposed in (Vetter and Poggio, 1996), can be improved through the 3D-model of the reference face. Since the linear object class approach did not assume correspondence between equations (1) and (2) or (3) and (4), shape and texture vectors had to be constructed for the complete face as a whole. On the other hand, modeling parts of a face (e.g. nose, mouth or eye region) in independent separate linear classes is highly preferable, because it allows a much better utilization of the example image set and therefore gives a much more detailed representation of a face (see also Choi et al., 1991). A full set of coefficients for shape and texture representation is evaluated separately for each part instead of just one set for the entire face.

3. Results

The algorithm was tested on 100 human faces. For each face, images were given in two orientations (24° and 0°) with a resolution of 256-by-256 pixels and 8 bit (more details are given in Appendix A).

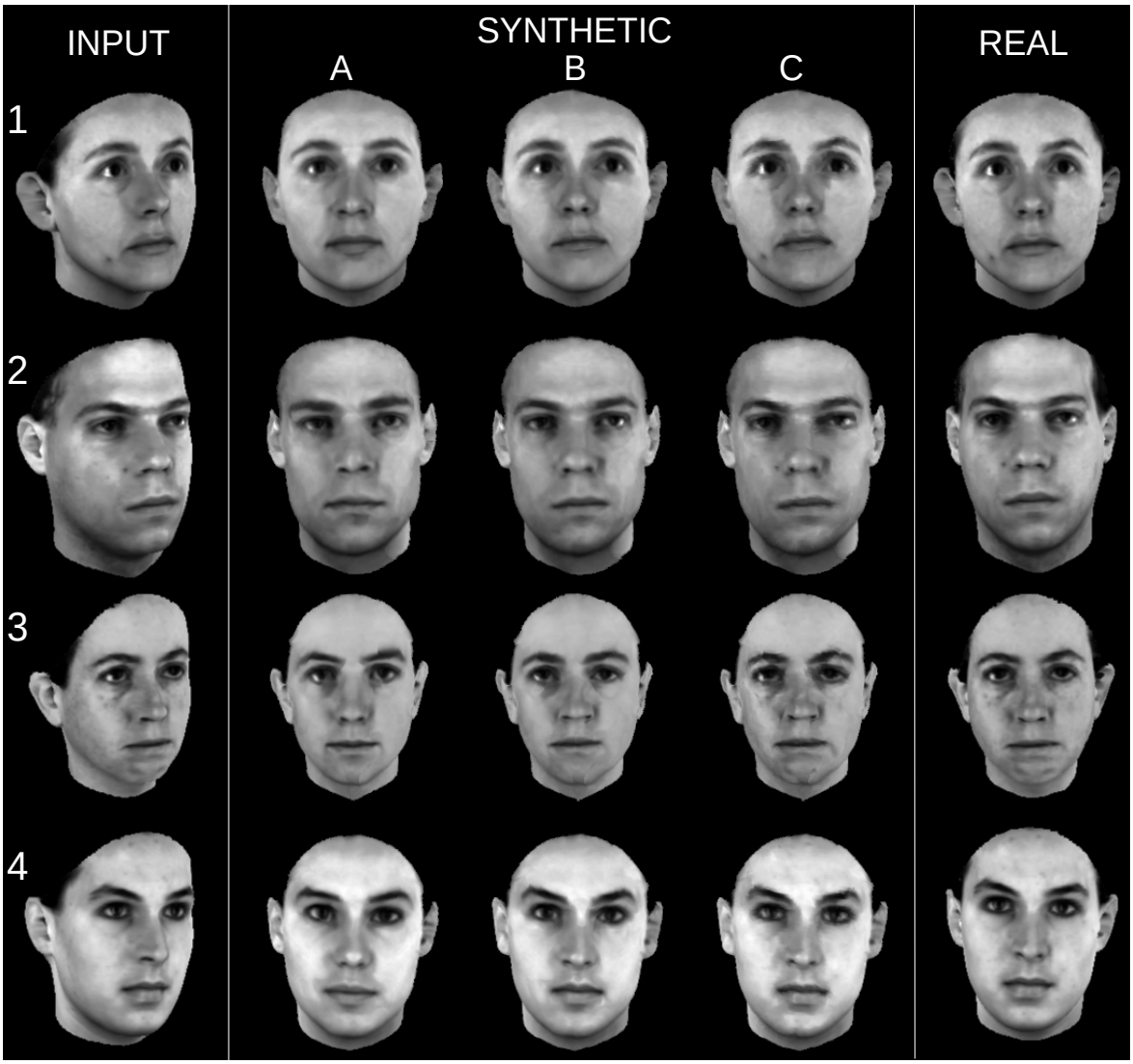

Fig. 4. Synthetic new frontal views (center columns) to a single given rotated (24°) image of a face (left column) are shown. The prior knowledge about faces was given through a example set of 99 pairs of images of different faces (not shown) in the two orientations. Column A shows the result based purely on the linear object class approach. Adding a single 3D-head model, the linear object class approach can be applied separately to the nose, mouth and eye region in a face (column B). The same 3D-model allows the texture mapping across the viewpoint change (column C). The frontal image of the real face is shown in the right column. The synthesized images in column A resemble the real faces already in general shape and appearance. In column B the similarity of details is improved, see for example the mouth of face 1 or the eyebrows and nose of face 2. Images of type C additionally capture localized peculiarities like the mole on the cheek of face 1 or the dimples on the chin of face 3. For faces without peculiarities all methods result in almost identical images (face 4).

In a *leave–one–out* procedure, a new "frontal" view of a face was synthesized to a given "rotated" view (24°). In each case the remaining 99 pairs of face images were used to build the linear 2D-shape and texture model of faces. Figure 4 shows the results for four faces for three different implementations of the algorithm (center rows A,B,C). The left column shows the test image given to the algorithm. The true "frontal" view to each test face from the data base is shown in the right column. The implementation used for generating the images in column A was identical to the method

already described in (Vetter and Poggio, 1996), the linear object class approach was applied to the shape and texture vector as a whole, no partitioning of the reference face or texture mapping across the viewpoints was applied. The method used in B was identical to A, except that the linear object class approach was applied separately to the different parts of a face. The three-dimensional head model was divided into four parts (see figure 2B) the eye, nose, mouth region, and the remaining part of the face. To segment the two reference images correctly, it was clearly necessary to render both of them from the same three-dimensional model of a head. Based on this segmentation, the texture and 2D-shape vectors for the different parts were separated and for each part a separate linear texture and 2D-shape model was applied. The final image was rendered after merging the new shape and texture vectors of the parts. The images shown in column C are the result of a combination of the technique described in B and texture mapping across the viewpoint change. After mapping a given "rotated" face image onto the "rotated" reference image, this normalized texture can be mapped onto the "frontal" reference face since the correspondence between the two images of the reference face is given through the three-dimensional model. The part of the "frontal" texture not visible in the "rotated" view is substituted by the texture obtained by the linear texture model as described under B.

The quality of the different synthesized "frontal" views was compared in a simple simulated recognition experiment. For each synthetic image, the most similar frontal face image in the data base of 100 real face images was computed. For the image comparison, two common similarity measures were used: a) the *correlation coefficient*, also known as *direction cosine*; and b) the *Euclidean distance* (L_2). Both measures were applied to the images in pixel representation without further processing.

Table 1. Comparing the different image synthesis techniques using Direction Cosines and L_2-Norm as distance measures. First, for all real frontal face images the average distance to its nearest neighbor (an image of a different face) was computed over an images test set of 100 frontal face images. Second, for all synthetic images (type A,B,C) the average value to its nearest neighbor was computed for both distance measures. For all synthetic images the real face image was found as nearest neighbor. Switching from technique A to B and from B to C the average values of Direction Cosines increase whereas the values of the L_2-Norm decrease, indicating an improved image similarity.

Average Image Distance to Nearest Neighbor

	L_2	*Direction Cosines*
Real Face Images	4780.3	0.9589
Synthetic Images Type A	3131.9	0.9811
Synthetic Images Type B	3039.3	0.9822
Synthetic Images Type C	2995.0	0.9827

The recognition rate of the synthesized images (type A,B,C) was 100 % correct, both similarity measures independently evaluated the true "frontal" view to a given "rotated" view of a face as the most similar image. This result holds for all three different methods applied for the image synthesis. The similarity of the synthetic images to the real face image improved by applying the linear object class approach separately to the parts and improved again adding the correspondence between the two reference images to the method. This improvement is indicated in table 1 where L_2-norm decreases whereas the correlation coefficients increase for the different techniques.

A crucial test for the synthesis of images is a direct comparison of real and synthetic images by human observers. As can be seen in figure 4 the synthesized images of type A already resemble the real faces in their general shape and appearance. The images of type B show the improved similarity of the details of the faces. The shape of the nose of face 2 for example, is highly improved changing from type A to B. However, both types are not able to capture the textural peculiarities of the faces. Only the images of type C reproduce features like birthmarks (see face 1 and 2) or freckles (see face 3). On the other hand, for faces without peculiarities all methods result in almost identical images (see face 4). The quality of the images synthesized to all 100 faces in the data set was evaluated by human observers. In a two alternative forced choice task 10 subjects were asked to decide which of the two frontal face images

matches a given rotated image (24°) best. One image was the "real" face the other a synthetic image generated applying the linear class method to the parts of the faces separately (method B). The first five images of the data set were used to familiarize the subjects with the task, whilst the performance was evaluated on the remaining 95 faces. Although there was no time limit for a response and all three images were shown simultaneously, there were only 6 faces classified correctly by all 10 subjects (see table 2). In all other cases the synthetic image was at least by one subject classified as the true image and in one case the synthetic image was found to match the rotated image better as the real frontal image. On average each observer was 74% correct whereas the chance level was at 50%.

On a second extended data set of 200 human faces (including the 100 faces used in the previous experiments) automated recognition experiments across various view point changes were performed. For each face, images were now available in four orientations (0°, 30°, 60° and 90°).
The data were divided into two subsets of 100 faces each. The two sets were used as a test and a training set and vice versa. For each orientation separately a linear model was built from the 100 faces of the training set. New views were generated using the synthesis technique type C for all faces of the test set. For each synthetic image, the most similar image in the whole data set of 200 different faces in the same orientation was computed. For comparison the *Euclidean distance* (L_2) was applied to the images in pixel representation without further processing. The error rates evaluated over all 200 faces are plotted in figure 5.

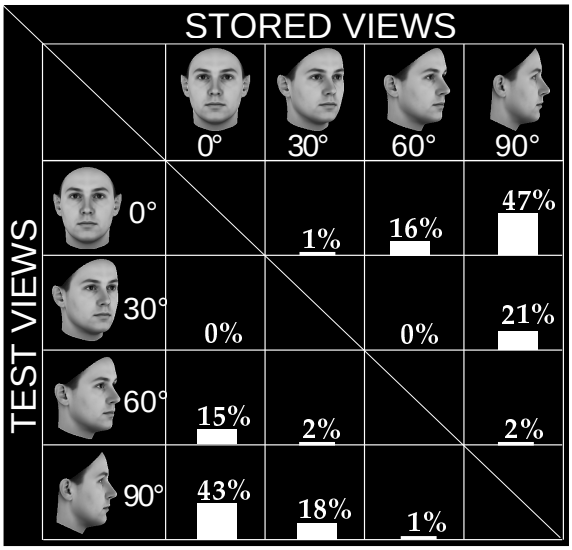

Fig. 5. Recognition errors evaluated on 200 faces for each transfer condition. For each test view new views were generated and compared to 200 images showing different faces in the same orientation. The model for the view synthesis was built on 100 training faces.

For rotations of 30° the error rate was at maximum 2%. For rotations of 60° the error rate was on average 17.5% and for 90° the error reached 45%. Chance would have lead to an error rate of 99.5%

4. Discussion

The results demonstrate a clear improvement in generating new synthetic images of a human face from only a single given view over techniques proposed previously (Beymer and Poggio, 1995; Vetter and Poggio, 1996). Here a single three-

Table 2. For 95 different faces a rotated image (24°) and two frontal images were shown to human observers simultaneously. They had to decide which of the frontal images was the synthesized image (type B) and which one was the real image. The table shows the error rate for 10 observers and the related number of faces. In average each observer was correct in 74% of the trails (chance level was 50%).

Classification of Synthetic Versus Real Face Images								
Error	0%	10%	20%	30%	40%	50%	60%	70 – 0%
Number of Faces	6	17	22	24	19	6	1	0

dimensional model of a human head was added to the linear class approach. Using this model, the reference images could be segmented into corresponding parts, and additionally any texture on the reference image could be mapped precisely across the viewpoint change. The information used from the three-dimensional model is equivalent to the addition of a single correspondence field across the viewpoint change. This addition increased the similarity of the synthesized image to the image of the real face for the shape as well as for the texture. The improvement could be demonstrated in automated image comparison as well as in perceptual experiments with human observers.

The results of the automated image comparison indicate the importance of the proposed face model for viewpoint independent face recognition systems. Here the synthetic rotated images were compared with the real frontal face image. It should also be noted that coefficients, which result from the decomposition of shape and texture into example shapes and textures, already give us a representation that is invariant under any 3D affine transformation, supposing of course the linear face model can produce a good approximation of the target face.

The synthesis of new views is not only interesting for the field of automated face recognition, it is also of great interest for applications in the fields of teleconferencing and virtualized reality. In contrast to pure machine vision tasks, there the human observer is the best judge of the quality of a method. (The problem of automating the measurement of picture quality is discussed in Xu and Hauske, 1994.)

The difficulties experienced by human observers in distinguishing between the synthetic images and the real face images indicate that a linear face model of 99 faces segmented into parts gives a good approximation of a new face. The low number of images that were consistently recognized as synthetic by all observers demonstrates that the model can already be applied to a wide range of faces and indicates possible applications of this method in the area of computer graphics. Clearly, the linear model depends on the given example set, thus to represent faces from a different race or a different age group, the model would need examples of these, an effect well known in human perception (cf. e.g. O'Toole et al., 1994).

The key step in the proposed technique is a dense correspondence field between images of faces seen from the same viewpoint. The optical flow technique used for the examples shown worked well. However, for images obtained under less controlled conditions, a more sophisticated method for finding the correspondence might be necessary. The use of images derived from three-dimensional head models allowed the generation of identical illumination conditions for all the example and test images. An extension of the proposed method to images obtained by a normal camera will lead to more unconstrained lighting conditions and will influence the correspondence finding step as well as the image synthesis. Both problems have been investigated by Hallinan (1995) for frontal face images. He demonstrated that a low dimensional linear illumination model is able to explain most variations in lighting. By fitting this model to an image, he could determine the lighting conditions as well as the correspondence. Additionally, the model allows the synthesis of new images of a face to given lighting conditions. This approach is very similar to the correspondence techniques based on *active shape models* (Cootes et al., 1995; Lanitis et al., 1995; Jones and Poggio, 1995; Vetter et al., 1997), which are more robust to local occlusions when applied to a known object class. There model parameters are optimized actively to model a target image.

Several open questions remain for a fully automated implementation. The separation of parts of an object to form separated subspaces could be done by computing the covariance between the pixels of the example images. However, for images at high resolution, this may require thousands of example images.

One of the most critical assumptions in the method presented here, is that the orientation of a face in the image must be known. Different techniques have already been reported to estimate the orientation of faces. Beymer (1993) used templates of faces of known orientation to estimate the pose of a face in a new image. Lanitis et al. (1995) used a flexible model for pose estimation, which was precice to $2 - 5°$. It is not clear yet

how precisely the orientation should be estimated to yield satisfactory results. However, considering that there was still some variance in the pose of the faces in our data set, a precision of 2° seems promising.

Appendix A. Face Images and Head Model

Images of 200 caucasian faces, showing a frontal view and views taken 24°, 30°, 60° and 90° from the frontal were available. The images were originally rendered for psychophysical experiments under mainly ambient illumination conditions from a data base of three-dimensional human head models recorded with a laser scanner ($Cyberware^{TM}$). The simulated pin-hole camera was set to a distance of 120 cm from the face. The different views were taken by moving the camera around the face. All faces were without makeup, accessories, and facial hair. Additionally, the head hair was removed digitally (but with manual editing), via a vertical cut behind the ears. The resolution of the grey-level images was 256-by-256 pixels and 8 bit.

Preprocessing: First the faces were segmented from the background and aligned roughly by automatically adjusting them to their two-dimensional centroid. The centroid was computed by evaluating separately the average of all x, y coordinates of the image pixels related to the face independent of their intensity value.

A single three-dimensional model of a human head was used to render the two reference images and to compute the correspondence field between these two images. This model was the average of 50 three-dimensional models of human heads, recorded with a laser scanner ($Cyberware^{TM}$). The averaging of the models was performed in a semiautomatic way. After manual editing and spatial alignment of the three-dimensional models, the correspondence between the models was computed using the same optic flow procedure on the texture data of the models as it was used for the images in this paper. The obtained correspondence between the three-dimensional models was not correct in the individual case, however, the average based on this correspondence resulted in a perfect three-dimensional model of a human head without any noticeable errors (see figure 2).

Appendix B. Computation of the Correspondence

To compute the 2D-shape vectors $\mathbf{s}^r, \mathbf{s}_i^r, \mathbf{s}_i^f$, used in equations (1) and (2), which are the vectors of the spatial distances between corresponding points in the face images, the correspondence of these points has to be established first. That means we have to find for every pixel location in an image, e.g. a pixel located on the nose, the corresponding pixel location on the nose in the other image. This is in general a hard problem. However, since all face images compared are in the same orientation, one can assume that the images are quite similar and occlusions are negligible. The simplified condition of a single view make it feasible to compare the images of the different faces with automatic techniques. Such algorithms are known from optical flow computation, in which points have to be tracked from one image to the other. We used a coarse-to-fine gradient-based method (Bergen et al., 1992) applied to the Laplacians of the images and following an implementation described in (Bergen and Hingorani, 1990). The Laplacian of the images were computed from the Gaussian pyramid adopting the algorithm proposed by (Burt and Adelson, 1983). For every point x, y in an image I, the error term $E = \sum(I_x \delta x + I_y \delta y - \delta I)^2$ is minimized for $\delta x, \delta y$, with I_x, I_y being the spatial image derivatives of the Laplacians and δI the difference of the Laplacians of the two compared images. The coarse-to-fine strategy refines the computed displacements when finer levels are processed. The final result of this computation $(\delta x, \delta y)$ is used as an approximation of the spatial displacement vector s in equation (1)and (2). The correspondence is computed towards the reference image from the example and test images. As a consequence, all vector fields have a common origin at the pixel locations of the reference image.

Appendix C. Linear shape and texture synthesis.

First the optimal linear decomposition of a given shape vector in equation (1) and a given texture vector in equation (3) was computed. To com-

pute the coefficients α_i (or similar β_i) the "initial" vector $\mathbf{T}^r$ of the new image is decomposed (in the sense of least square) to the q example image vectors $\mathbf{T}_i^r$ given through the example images by minimizing

$$||\mathbf{T}^r - \sum_{i=1}^{q} \alpha_i \mathbf{T}_i^r||^2.$$

The numerical solution for α_i and β_i was obtained by an standard SVD-algorithm (Press et al., 1992). The new shape and texture vectors for the "frontal" view were obtained through simple summation of the weighted "frontal" vectors (equations(2) and (4)).

Appendix D. Blending of patches

Blending of patches is used at different steps in the proposed algorithm. It is applied for merging different regions of texture as well as for merging regions of correspondence fields which were computed separately for different parts of the face. Such a patch work might have little discontinuities at the borders between the different patches. It is known that human observers are very sensitive to such effects and the overall perception of the image might be dominated by these.

For images Burt and Adelson (1983,1985) proposed a multiresolution approach for merging images or components of images. First, each image patch is decomposed into bandpass filtered component images. Secondly, this component images are merged separately for each band to form mosaic images by weighted averaging in the transition zone. The weights for each band were computed by generating a Gaussian pyramid of the binary mask of each component. The sum of the weights of the different components is normalized for each band to one at each image pixel. This is necessary, since it cannot be guaranteed in general that the sum of the weights is one. Finally, these bandpass mosaic images are summed to obtain the desired composite image. This method was applied to merge the different patches for the texture construction as well as to combine the texture mapped across the viewpoint change with the missing part taken from the constructed one. Originally this merging method was only described for an application to images, however, the application to patches of correspondence fields eliminates visible discontinuities in the warped images. Taking a correspondence field as an image with a vector valued intensity, the merging technique was applied to the x and y components of the correspondence vectors separately.

Appendix E. Synthesis of the New Image

The final step is image rendering. The new image can be generated combining the texture and shape vector generated in the previous steps. Since both are given in the coordinates of the reference image, for every pixel in the reference image the pixel intensity and coordinates to the new location are given. The new location generally does not coincide with the equally spaced grid of pixels of the destination image. The final pixel intensities of the new image are computed by linear interpolation, a commonly used solution of this problem known as forward warping (Wolberg, 1990).

Acknowledgements

I am grateful to T. Poggio, H.H. Bülthoff and V. Blanz for useful discussions and suggestions. Special thanks to Alice O'Toole for editing the manuscript and for her endurance in discussing the paper. I would like to thank Nikolaus Troje for providing the images.

References

1. Aizawa, K. Harashima, H. and Saito, T., "Model-based analysis synthesis image coding (MBASIC) system for a person's face." Signal Processing: Image Communication, 1:139–152, 1989.
2. Akimoto, T., Suenaga, Y. and Wallace, R.S., "Automatic creation of 3D facial models." IEEE Computer Graphics and Applications, 13(3):16–22, 1993.
3. Bergen, J.R. Anandan, P., Hanna, K.J. and Hingorani, R., "Hierarchical model-based motion estimation." Proceedings of the European Conference on Computer Vision, pp 237–252, Santa Margherita Ligure, Italy, 1992.
4. Bergen, J.R. and Hingorani, R., "Hierarchical motion-based frame rate conversion." Technical report, David Sarnoff Research Center Princeton NJ 08540, 1990.

5. Beymer, D., " Face recognition under varying pose." A.I. Memo No. 1461, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1993.
6. Beymer, D. and Poggio, T., " Face recognition from one model view." Proceedings of the 5th International Conference on Computer Vision, 1995.
7. Beymer, D. and Poggio, T., "Image representation for visual learning." Science, 272:1905–1909, 1996.
8. Beymer, D., Shashua, A. and Poggio, T., "Example-based image analysis and synthesis." A.I. Memo No. 1431, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1993.
9. Burt, P.J. and Adelson, E.H., "The Laplacian pyramide as a compact image code." IEEE Transactions on Communications, (31):532–540, 1983.
10. Burt, P.J. and Adelson, E.H., " Merging images through pattern decomposition." Applications of Digital Image Processing VIII, 575, pp 173–181. SPIE The International Society for Optical Engeneering, 1985.
11. Choi, C.S., Okazaki, T., Harashima, H. and Takebe, T., "A system of analyzing and synthesizing facial images." In Proc. IEEE Int. Symposium of Circuit and Syatems (ISCAS91), pp 2665–2668, 1991.
12. Cootes, T.F. , Taylor, C.J., Cooper, D.H. and Graham, J., "Active shape models - their training and application." Computer Vision and Image Understanding, 61:38–59, 1995.
13. Craw, I. and Cameron, P., " Parameterizing images for recognition and reconstruction." In Peter Mowforth, editor, Proc. British Machine Vision Conference, pp 367–370. Springer, 1991.
14. Hallinan, P.W. "A deformable model for the recognition of human faces under arbitrary illumination." Doctoral thesis, Harvard University, Cambridge, Massachusetts, 1995.
15. Horn, B.K.P. " Robot vision." MIT electrical engineering and computer science series. MIT Press, Cambridge, Ma, 1987.
16. Huang, T.S., and Lee,C.H., " Motion and structure from orthographic projections." IEEE Transactions on Pattern Analysis and Machine Intelligence, 2(5):536–540, 1989.
17. Jones, M., and Poggio, T., "Model-based matching of line drawings by linear combination of prototypes." Proceedings of the 5th International Conference on Computer Vision, 1995.
18. Lanitis, A., Taylor, C.J., Cootes,T.F. , and Ahmad, T., "Automatic interpretation of human faces and hand gestures using flexible models." In M.Bichsel, editor, Proc. International Workshop on Face and Gesture Recognition, pp 98–103, Zurich, Switzerland, 1995.
19. O'Toole, A.J., Deffenbacher, K.A., Valentin, D. and Abdi, H., "Structural aspects of face recognition and the other-race effect." Memory and Cognition, 22:208–224, 1994.
20. Poggio, T. and Brunelli, R., "A novel approach to graphics." Technical report 1354, MIT Media Laboratory Perceptual Computing Section, 1992.
21. Press, Teukolsky, Vetterling and Flannery. " Numerical recipes in C : the art of scientific computing." Cambridge University Press, Cambridge, 1992.
22. C.A. Rothwell, D.A. Forsyth, Zissermann, A. and Mundy, J.L., "Extracting projective structure from single perspective views of 3D point sets." In Proceedings of the International Conference on Computer Vision (ICCV), pp 573–582, Berlin, Germany, May 1993.
23. Terzopoulos, D. and Waters, K., "Analysis and synthesis of facial image sequences using physical and anatomical models." IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(6):569–579, 1993.
24. Thalmann, N.D. and Thalmann, D., "Digital actors for interactive television." Proceedings of the IEEE, 83(7):1022–1031, 1995.
25. Vetter, T., Jones, M. and Poggio, T., "A bootstrapping algorithm for learning linearized models of object classes," IEEE Conference on Computer Vision and Pattern Recognition, 1997.
26. Vetter, T. and Poggio, T. "Symmetric 3D objects are an easy case for 2D object recognition" Spatial Vision, 8(4):443–453,1994
27. Vetter, T. and Poggio, T., "Image synthesis from a single example image." In B. Buxton and R. Cippola, editors, Computer Vision – ECCV'96, Cambridge UK, 1996. Springer, Lecture Notes in Computer Science 1065.
28. Wolberg, G., "Image Warping." IEEE Computer Society Press, Los Alamitos CA, 1990.
29. Xu, W. and Hauske, G., "Picture quality evaluation based on error segmentation" Proc. SPIE, Visual Communications and Image Processing, 2308:1–12, 1994.