コンピュータビジョンと
イメージメディア 106−11
（1997. 7. 25）

ファジーな知識を用いた無校正画像からのユークリッド構造の復元：顔画像生成への適用

張 正友† 磯野 勝宣 赤松 茂
ATR 人間情報通信研究所
† On leave from INRIA Sophia-Antipolis, France.

無校正画像の応用には画像生成のような多くのアプリケーションがある。しかし、射影空間またはアフィン空間における、新たな画像の姿勢を指定することは容易ではない。本研究では、距離や角度といったファジーな知識を用いて無校正画像からのユークリッド構造の復元を行う。我々は通常、特定の物体に対しては正確ではないが、その物体のカテゴリーに対する知識を持っている。知識はガウス変数としてモデル化する。今回は知識として、6種類の一般的な知識を用いる。一旦、ユークリッド構造が得られれば、ユークリッド空間における、任意の姿勢の指定が可能となる。本手法を新規顔画像の生成に適用する。画像生成における多くの問題点を明らかかし、解決する。例えば、エッジ点を用いることにより、occlusion 問題を解決する。

Euclidean Structure from Uncalibrated Images Using Fuzzy Domain Knowledge: Application to Facial Images Synthesis

Zhengyou Zhang† Katsunori Isono Shigeru Akamatsu
ATR Human Information Processing Research Labs.
† On leave from INRIA Sophia-Antipolis, France.

Use of uncalibrated images has found many applications such as image synthesis. However, it is not easy to specify the desired position of the new image in projective or affine space. This paper proposes to recover Euclidean structure from uncalibrated images using domain knowledge such as distances and angles. The knowledge we have is usually about an object category, but not very precise for the particular object being considered. The variation (fuzziness) is modeled as a Gaussian variable. Six types of common knowledge are formulated. Once we have a Euclidean description, the task to specify the desired potion in Euclidean space becomes trivial. The proposed technique is then applied to synthesis of new facial images. A number of difficulties existing in image synthesis are identified and solved. For example, we propose to use edge points to deal with occlusion.

1 Introduction

The study of uncalibrated images [2, 3] has many important applications such as the reconstruction of the environment from a sequence of video images where the parameters of the video lens is submitted to continuous modification such that camera calibration in the classical sense is not possible. We cannot extract any metric information, but a projective structure is still possible if the camera can be considered as a pinhole. The projective structure still contains rich information, such as coplanarity, collinearity, and cross ratios (ratio of ratios of distances), which is sometimes sufficient for artificial systems, such as robots, to perform tasks such as navigation and object recognition [8, 9, 1].

However, for many other applications, such as virtual reality, we need Euclidean structure, but not the projective structure, because a human being is familiar with Euclidean environments. For those who have worked on animation from uncalibrated images, they must know how it is difficult to specify the desired trajectory of the camera in projective space. In this paper, we propose a technique to recover a quasi-Euclidean structure from uncalibrated images using fuzzy domain knowledge. The application domain we consider is the generation of new face images.

In this paper, we consider Euclidean structure recovery from affine cameras using fuzzy knowledge. Use of affine cameras [5] is motivated by the fact that the depth variation in our domain application (faces) is not very large. In consequence, affine camera model is a reasonably good approximation to the true camera. We develop a technique which takes into account variety of fuzzy knowledge, ranging from simple ones such as location of points and distances between two points to more complex ones such as ratios of distances. By "fuzzy", we mean that we do not have exact measure of the particular object being considered. What we have is the knowledge about the object category (faces in our case). Therefore, the recovered Euclidean structure is not necessarily a precise Euclidean description of that object, but is useful for many applications such as new image synthesis. (Note that, the recovered Euclidean structure still contains all exact affine information, even if the Euclidean one is not correct.)

Section 2 outlines the path from uncalibrated images to affine structure using affine camera model, then to Euclidean structure using fuzzy knowledge. A method is proposed to fix an arbitrary Euclidean coordinate system. Section 3 formulates different types of fuzzy knowledge and describes how to use them to recover the Euclidean structure. Section 4 provides the details of our system for synthesizing new facial images from two real images. This has a number of important applications such as video conferencing and virtual reality. A number of difficulties have been identified. In particular, we propose to combine points of interest and edges to deal with occlusion and apply the technique described in Section 3 to recover a quasi-Euclidean structure which considerably facilitates the specification of the desired position of the new image. Domain knowledge has been extracted through statistical analysis of 3D range data of real female and male faces.

2 Overview

Using affine camera model, we can use the technique described in [7] to determine the affine fundamental matrix $\mathbf{F}_A$ between two images. Once $\mathbf{F}_A$ is known, affine structure of the scene can be reconstructed, i.e., given a pair of matched image points $(\mathbf{m}_i, \mathbf{m}'_i)$, we can reconstruct its corresponding space point $\mathbf{x}_i$ with respect to an affine coordinate system. If we denote the Euclidean coordinates of this point by $\mathbf{y}_i$, then there exists the following relationship:

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i + \mathbf{b} , \tag{1}$$

where $\mathbf{A}$ is a 3×3 non-singular matrix and $\mathbf{b}$ is a 3D vector. There are in total 12 *degrees of freedom* (DoF) in an affine transformation. Our objective is to determine $\mathbf{A}$ and $\mathbf{b}$ from the given $\mathbf{x}_i$ and the knowledge of $\mathbf{y}_i$.

The first solution is an easy one. If we know the Euclidean coordinates of at least 4 points, $\mathbf{A}$ and $\mathbf{b}$ can be solved with a linear least-squares technique. However, the Euclidean coordinates of points are in general difficult to obtain.

Alternatively, we can use other form of knowledge. For example, if we have knowledge of at least 3 distances and 3 angles in general configuration, we can redress the structure in Euclidean space by choosing arbitrarily a reference frame, as to be discussed below.

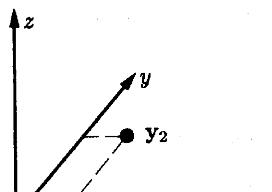


Figure 1: *Fixing a Euclidean reference frame by choosing three points. Points $\mathbf{y}_0$, $\mathbf{y}_1$ and $\mathbf{y}_2$ are the chosen points in Euclidean space.*

A Euclidean reference frame has 6 degrees of freedom: 3 for the position and 3 for the orientation. We can fix the Euclidean reference frame by arbitrarily choosing three non-collinear points in the following way. Let $\mathbf{x}_0$, $\mathbf{x}_1$ and $\mathbf{x}_2$ be the 3 chosen affine points. Refer to Fig. 1. We can set one point, say $\mathbf{x}_0$, after affine transformation as the origin $\mathbf{0} = [0,0,0]^T$:

$$\mathbf{A}\mathbf{x}_0 + \mathbf{b} = \mathbf{0} \,, \tag{2}$$

which removes 3 DoF. We can set the vector joining $\mathbf{x}_0$ and $\mathbf{x}_1$ after affine transformation to be parallel to, say, x-axis, i.e.,

$$\mathbf{A}(\mathbf{x}_1 - \mathbf{x}_0) = k[1,0,0]^T \,, \tag{3}$$

where k is an arbitrary non-zero scalar. This removes 2 more DoF. Let $\mathbf{v}_1 = \mathbf{A}(\mathbf{x}_1 - \mathbf{x}_0) \equiv [v_{11}, v_{12}, v_{13}]^T$. The 2 constraints contained in (3) are:

$$v_{12} = 0 \quad \text{and} \quad v_{13} = 0 \,.$$

Finally, we can set the plane passing through $\mathbf{x}_0, \mathbf{x}_1$ and $\mathbf{x}_2$ after affine transformation as the xy-plane. This is equivalent to setting the vector joining $\mathbf{x}_0$ and $\mathbf{x}_2$ after affine transformation to be orthogonal to z-axis, i.e.,

$$[0,0,1]^T \mathbf{A}(\mathbf{x}_2 - \mathbf{x}_0) = 0 \,. \tag{4}$$

Let $\mathbf{v}_2 = \mathbf{A}(\mathbf{x}_2 - \mathbf{x}_0) \equiv [v_{21}, v_{22}, v_{23}]^T$. The above constraint is equivalent to $v_{23} = 0$. We now completely specify the Euclidean frame.

In order to recover the Euclidean structure, we need 6 additional constraints. We identify the following 6 types of knowledge:

1. the Euclidean coordinates of a point;
2. the distance between two points;
3. the angle between two vectors;
4. a vector should be parallel to a given vector;
5. a vector should be equal to a given one;
6. the ratio of two distances between non-collinear points.

The knowledge available will not, in general, very precise. The fuzziness is modeled by a Gaussian variable. By providing a piece of domain knowledge, the user should also specify his belief on its precision in terms of standard deviation. In our case, the knowledge about faces structure is obtained through statistical analysis of 36 sets of range data for female faces and 31 sets of range data for males (see Sect. 4.2).

3 Fuzzy Knowledge

This section will formulate the 6 types of constraints in terms of Mahalanobis distances and explain how to use them in order to recover the Euclidean structure.

In the following, we use $\mathbf{x}_i$ to denote the affine coordinates of a given point, $\mathbf{y}_i$ to denote its expected Euclidean coordinates, and $\mathbf{z}_i$ to denote its estimated Euclidean coordinates:

$$\mathbf{z}_i = \mathbf{A}\mathbf{x}_i + \mathbf{b} \,.$$

3.1 Constraints

Euclidean coordinates of a point. The Euclidean coordinates of a point $\mathbf{x}_i$ should be $\mathbf{y}_i$ with covariance matrix $\mathbf{\Lambda}_i$. This constraint can be expressed as a squared Mahalanobis distance as

$$r_p^2 = (\mathbf{z}_i - \mathbf{y}_i)^T \mathbf{\Lambda}_i^{-1} (\mathbf{z}_i - \mathbf{y}_i) \,. \tag{5}$$

This knowledge provides three constraints.

Distance between two points. The distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ should be equal to $\hat{d}_{ij}$ with standard deviation $\sigma_{d_{ij}}$. The Mahalanobis distance is

$$r_d = (d_{ij} - \hat{d}_{ij})/\sigma_{d_{ij}} \,, \tag{6}$$

where $d_{ij} = \|\mathbf{z}_i - \mathbf{z}_j\| = \|\mathbf{A}(\mathbf{x}_i - \mathbf{x}_j)\|$. This knowledge provides one constraint.

Angle between two vectors. Let $\tilde{\mathbf{v}}_i$ and $\tilde{\mathbf{v}}_j$ be two affine vectors. The angle between them in Euclidean space is required to be $\hat{\theta}$ with standard deviation σ_θ. Let the corresponding Euclidean vectors be $\mathbf{v}_i = \mathbf{A}\tilde{\mathbf{v}}_i$ and $\mathbf{v}_j = \mathbf{A}\tilde{\mathbf{v}}_j$. There are two ways to express the angle:

$$\cos\theta = \frac{\mathbf{v}_i^T \mathbf{v}_j}{\|\mathbf{v}_i\| \, \|\mathbf{v}_j\|} \quad \text{and} \quad \sin\theta = \frac{\|\mathbf{v}_i \times \mathbf{v}_j\|}{\|\mathbf{v}_i\| \, \|\mathbf{v}_j\|} \,.$$

Let $\theta = \hat{\theta} + \delta\theta$. Assume $\delta\theta$ is small, we can use either cosine or sine function to do linearization. If cosine function is used, then

$$\cos\theta = \cos(\hat{\theta} + \delta\theta) \approx \cos\hat{\theta} - \sin\hat{\theta}\,\delta\theta \,,$$

and the Mahalanobis distance is given by

$$r_\theta = \frac{\delta\theta}{\sigma_\theta} = \frac{\cos\hat{\theta} - \mathbf{v}_i^T \mathbf{v}_j / (\|\mathbf{v}_i\| \, \|\mathbf{v}_j\|)}{\sigma_\theta \sin\hat{\theta}} \,. \tag{7}$$

If sine function is used, then

$$\sin\theta = \sin(\hat{\theta} + \delta\theta) \approx \sin\hat{\theta} + \cos\hat{\theta}\,\delta\theta \,,$$

and the Mahalanobis distance is given by

$$r_\theta = \frac{\delta\theta}{\sigma_\theta} = \frac{\|\mathbf{v}_i \times \mathbf{v}_j\| / (\|\mathbf{v}_i\| \, \|\mathbf{v}_j\|) - \sin\hat{\theta}}{\sigma_\theta \cos\hat{\theta}} \,. \tag{8}$$

When $-\pi/4 < \hat{\theta} < \pi/4$, sine function should be used; otherwise, cosine function should be used. This knowledge provides one constraint.

Orientation of a vector. Let $\tilde{\mathbf{v}}_i$ be the affine vector, and $\mathbf{v}_i = \mathbf{A}\tilde{\mathbf{v}}_i$ be the Euclidean vector. $\mathbf{v}_i$ is required to be parallel to $\hat{\mathbf{v}}$ with standard deviation of angle σ_u. Note that the fuzziness of this knowledge is expressed in terms of that on angle, because the angle between $\mathbf{v}_i$ and $\hat{\mathbf{v}}$ is expected to be 0.

Let $\mathbf{u}_i = \mathbf{v}_i / \|\mathbf{v}_i\|$ and $\hat{\mathbf{u}} = \hat{\mathbf{v}} / \|\hat{\mathbf{v}}\|$ be unit vectors. Let the error vector $\mathbf{e} = \mathbf{u}_i - \hat{\mathbf{u}}$ is on the unit sphere. For a given error angle θ, $\mathbf{r}$ is on a circle around $\hat{\mathbf{u}}$ (see Fig. 2a). Let us consider the projection of $\mathbf{e}$ on

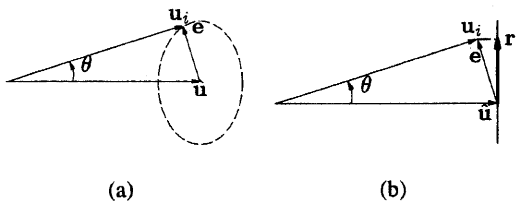


Figure 2: *Knowledge about the orientation of a vector.*

the plane perpendicular to $\hat{\mathbf{u}}$ (Fig. 2b shows an orthogonal section). The projection, denoted by $\mathbf{r}$, is simply given by $\mathbf{r} = \mathbf{u}_i - (\mathbf{u}_i^T\hat{\mathbf{u}})\hat{\mathbf{u}}$. If θ follows Gaussian distribution with standard deviation σ_θ, then $\mathbf{r}$ follows 2D Gaussian distribution on the plane around $\hat{\mathbf{u}}$ with covariance matrix $\mathrm{diag}(\sin^2\sigma_\theta, \sin^2\sigma_\theta)$. In 3D, if $\hat{\mathbf{u}} = [0,0,1]^T$, then the covariance matrix of $\mathbf{r}$, $\Lambda_\mathbf{r}$, is given by $\mathrm{diag}(\sin^2\sigma_\theta, \sin^2\sigma_\theta, 0)$. For a general $\hat{\mathbf{u}}$, we can always find a rotation matrix $\mathbf{R}$ such that $\hat{\mathbf{u}} = \mathbf{R}\ [0,0,1]^T$, then the covariance matrix of $\mathbf{r}$ is

$$\Lambda_\mathbf{r} = \mathbf{R}\ \mathrm{diag}(\sin^2\sigma_\theta, \sin^2\sigma_\theta, 0)\ \mathbf{R}^T\ .$$

There are many ways to define $\mathbf{R}$. We can choose any vector perpendicular to $\hat{\mathbf{u}} \equiv [a,b,c]^T$, for example $\hat{\mathbf{u}}_0 = (1/\sqrt{2a^2+b^2+c^2+2bc})[b+c, -a, -a]^T$, then the rotation matrix can be $\mathbf{R} = [\hat{\mathbf{u}}_0 \times \hat{\mathbf{u}} \quad \hat{\mathbf{u}}_0 \quad \hat{\mathbf{u}}]$. Therefore, the Mahalanobis distance can then be formulated as follows

$$\begin{aligned} r_u^2 &= \mathbf{r}^T\Lambda_\mathbf{r}^{-1}\mathbf{r} \\ &= \mathbf{r}^T\mathbf{R}\ \mathrm{diag}(1/\sin^2\sigma_\theta, 1/\sin^2\sigma_\theta, 0)\ \mathbf{R}^T\mathbf{r}\ . \end{aligned} \quad (9)$$

As $\Lambda_\mathbf{r}$ is not invertible, pseudo-inverse has been used. This knowledge provides two constraints.

A given vector. An affine vector is required to be equal to a given vector. This is equivalent to two pieces of knowledge: the length of the vector (distance) (6) and its orientation (9).

Ratio of two distances. Affine transformation conserves the ratio of lengths of two vectors if they are parallel to each other. If they are not parallel, then the ratio of lengths provides one constraint in recovering Euclidean structure. Let $\tilde{\mathbf{v}}_i$ and $\tilde{\mathbf{v}}_j$ be two affine vectors. The ratio of their lengths in Euclidean space is required to be $\hat{r}$ with standard deviation σ_r. The Mahalanobis distance is given by

$$r_r = \frac{\|\mathbf{A}\tilde{\mathbf{v}}_i\|/\|\mathbf{A}\tilde{\mathbf{v}}_j\| - \hat{r}}{\sigma_r}\ . \quad (10)$$

3.2 Euclidean Recovery

If we are given a sufficient number of constraints in the above form, we can estimate the affine transformation $(\mathbf{A}, \mathbf{b})$ which brings the affine structure into a Euclidean space by minimizing the sum of Mahalanobis distances:

$$\sum r_p^2 + \sum r_d^2 + \sum r_\theta^2 + \sum r_u^2 + \sum r_r^2\ . \quad (11)$$

This is a nonlinear least-squares problem. For our applications, we just start with initial guess $\mathbf{A} = \mathrm{diag}(1,1,1)$ and $\mathbf{b} = [0,0,0]^T$, and it works very well.

If no constraints (Euclidean coordinates of points and orientation of vectors) are provided to specify the Euclidean coordinate system, we have to use the technique described in Sect. 2 to fix it. The 6 constraints contained in (2), (3) and (4) are added to (11). The standard deviation for each constraint is set to a small value, because these constraints are required to be satisfied exactly.

4 Application to Facial Images Synthesis

Synthesizing new facial images from a small number of real images is important for many applications such as video conferencing and virtual reality. Previous work includes that of Mukaigawa et al. [4] and that of Seitz and Dyer [6]. Mukaigawa et al. assume orthographic projection and generate new images through linear combination of the original images. The coefficients of the linear combination are determined by specifying the desired position of a few points in the new image. Seitz and Dyer use a technique called view morphing to generate an image sequence corresponding to a camera moving along a line joining the optical centers of the two original images.

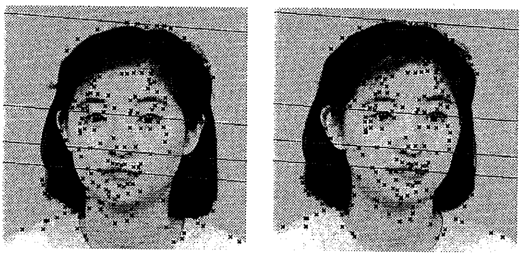
4.1 General Principle

Image synthesis from real images works as follows. *First*, point correspondences are established between images, from which the epipolar geometry is determined. Using points as vertices, we can divide each image into a set of triangular patches. *Second*, points are transfered into a new image. *Finally*, textures (colors) from the original images are mapped to the triangular patches in the new image.

There are a few difficulties which one should solve in synthesizing new views from real images, which are briefly described below.

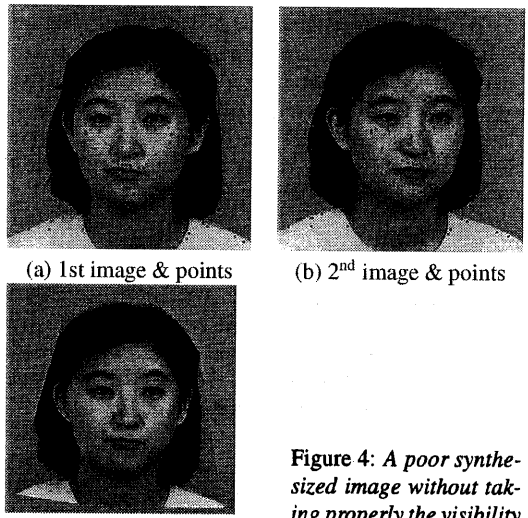
4.1.1 Establishing point correspondences between images

Since we assume affine camera model, we combine the robust matching technique described in [10] and the technique described in [7] to establish point correspondences and determine the affine fundamental matrix between two images. It works reasonably well for facial images differed by a rotation in depth of up to 30 degrees. Psychological studies also show that recognition rate by human decreases rapidly with the rotation angle in depth. Figure 3 shows an example of matched points between 2 facial images differed by a rotation in depth of about 20 degrees.



(a) first image (b) second image

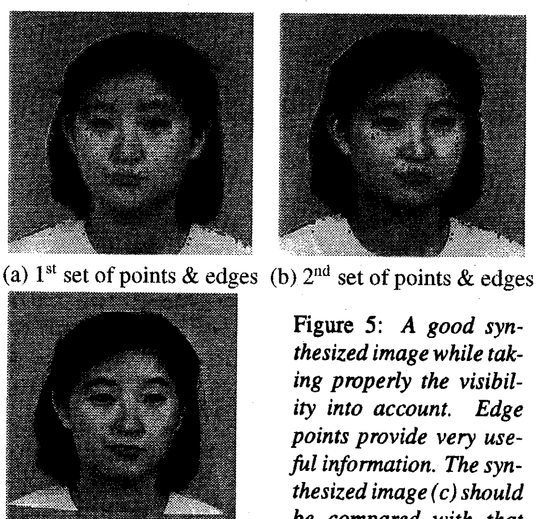
Figure 3: *Matched points (indicated by crosses) between two facial images with four corresponding epipolar lines overlaid*



(a) 1st image & points (b) 2nd image & points

(c) synthesized image

Figure 4: *A poor synthesized image without taking properly the visibility into account.*



(a) 1st set of points & edges (b) 2nd set of points & edges

(c) synthesized image

Figure 5: *A good synthesized image while taking properly the visibility into account. Edge points provide very useful information. The synthesized image (c) should be compared with that shown in Fig. 4c.*

4.1.2 Dealing with changes in visibility

However, using points obtained above alone, we do not generate images of good quality, especially for regions containing significant changes in visibility, for example a part is only visible in one image. One example of synthesized image without taking into account visibility is shown in Fig. 4. The synthesized image (Fig. 4c) corresponds to the intermediate position (10 degrees of rotation in depth) between the two original images. The texture of the first image is mapped to the generated image. We can see that the right cheek and ear are not very good and that the left ear appears too big. If the texture of the second image is used, the result is even worse because the left ear is not visible in the second image.

We found that combining edges and feature points improves considerable the quality of synthesized images. In order to deal with occlusion, we assign the occluded edges to the closest visible ones. Refer to Fig. 5a and b. The two edge chains of the first image corresponding to the left cheek and the lower part of the left ear are assigned to the same edge chain of the second image corresponding to the left cheek. Of course, the 3D information about these edges will not be correct (actually, there is no geometric way to recover 3D information from a single image), but the rendering in the new image looks very realistic. The synthesized image (Fig. 5c) should be compared with that shown in Fig. 4c.

4.1.3 Dealing with photometric variation in different images

A patch in the new image has two corresponding patches in the original images, except for those which are only visible in one image. During texture mapping, we have to decide which color to use. The difficulty lies in the fact that two images usually do not have the same color and that the corresponding texture patches do not have the same resolution. If we only use texture from one image, then the quality of rendering is poor for small triangular patches. Figure 6a shows a synthesized image using the first image as texture, and we can see that the quality on the right part of the face (i.e., the left part of the image) is poor. Figure 6b shows a synthesized image using the second image as texture, and we can see that the left ear is not visible.

We therefore choose to use both images by weighting them according to the areas of the triangles. Let Δ_1 and

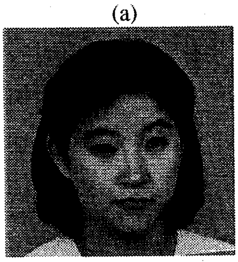
Δ_2 be two corresponding triangles of the original images, and Δ_3 be the corresponding triangle in the new image. Their areas are denoted by S_1, S_2 and S_3, respectively. Let $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$ be a pixel in correspondence inside the triangles Δ_1, Δ_2 and Δ_3. Let $I(\mathbf{p})$ be the intensity of pixel $\mathbf{p}$. We use the following formula to compute the intensity of $\mathbf{p}_3$ in the new image:

$$I(\mathbf{p}_3) = \frac{S_1 I(\mathbf{p}_1) + S_2 I(\mathbf{p}_2)}{S_1 + S_2} .$$

Thus, the patch having a larger area contribute more to the texture in the new image. An extreme case is: when a patch is occluded in one image, its area is 0 and the texture from the other image is then taken. An example using both images as texture is shown in Fig. 6c.

(a)

(b)



(c)

Figure 6: *Comparison of two texture mapping techniques. (a) texture from the first image; (b) texture from the second image; (c) texture from two images.*

4.1.4 Specifying desired position of the new image

We must supply information of the desired position of the new image. This can be done by specifying the position of at least four feature points in the new image. This is tricky because we do not know in general where these points should be. We have to try different positions before obtaining the desired result. Figure 7 shows a synthesized image which is not visually realistic. If we have a reference image of what the new image should look like, then we can use it as an indicator to specify the position of the feature points. Figure 8 is the reference image used to generate the images shown in Fig. 6. In the next section, we propose to build a Euclidean model of the face using domain knowledge, which facilitates enormously the specification of the desired position of the new image.

Figure 7: *An unrealistic synthesized image because the position of the feature points is not well specified.*

Figure 8: *The reference image (of a different person) used to generate the images shown in Fig. 6*

4.2 Recovering the Euclidean Structure Using Fuzzy Knowledge

If we have the Euclidean model of the face, then it is extremely easy to specify the desired position of the new image: front view, side view to the left by 10 degrees, etc.

4.2.1 Fuzzy domain knowledge

We have a rich knowledge of our face shape, for example, the line joining two eyes is almost perpendicular to the vertical symmetric line of the head. The domain knowledge for our application is obtained through statistical analysis of 36 sets of 3D range data of female heads and 31 sets of male heads. Seven points on the face are selected, which are used to express the domain knowledge of the faces. These points are indicated by number 0 through 6 in Fig. 9, corresponding to the corners of the eyes, the tip of the nose, the corners of the mouth, and the middle points of the ears. Point 7 and 8 are virtual ones: they are the middle point between point 0 and 1 and that between point 3 and 4. Table 1 lists a subset of the fuzzy knowledge base we have built for faces. In Table 1a, distances between two feature points are shown. For example, the first row shows that the average distance between point 0 and 1 is 122 mm with sample deviation 5 mm for female faces and 125 mm with sample deviation 6 mm for male faces. In Table 1b, angles between two vectors are shown. For example, the first row shows that the average angle between vector from point 0 to point 1 and vector from point 0 to point 2 is equal to 44° with sample deviation 2° for female faces and to 46° with sample deviation 2° for male faces. Table 1c shows ratios of distances and sample deviations for d_{02}/d_{32} and d_{50}/d_{53}.

4.2.2 Recovered Euclidean Shape

In order to recover the Euclidean shape of the face from two images shown in Fig. 3, we first select manually 6

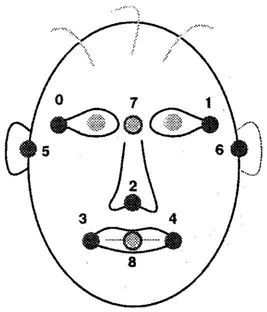


Figure 9: *Feature points selected for the domain knowledge base*

Table 1: *A subset of the fuzzy knowledge base for faces*

(a) distances (in millimeters)

	female faces		male faces	
points	distance	s.d. (σ_d)	distance	s.d. (σ_d)
$0 \leftrightarrow 1$	122	5	125	6
$0 \leftrightarrow 2$	84	4	89	5
$0 \leftrightarrow 3$	95	4	102	6
$3 \leftrightarrow 2$	63	3	66	4
$3 \leftrightarrow 4$	58	4	59	5
$5 \leftrightarrow 0$	99	5	106	5
$5 \leftrightarrow 2$	159	6	171	6
$5 \leftrightarrow 3$	132	6	144	7

(b) angles (in degrees)

	female faces		male faces	
vectors	angle	s.d. (σ_θ)	angle	s.d. (σ_θ)
$\angle(01,02)$	44	2	46	2
$\angle(01,34)$	3	2	3	2
$\angle(01,78)$	90	2	89	2
$\angle(02,03)$	41	3	40	3
$\angle(32,34)$	63	2	65	4
$\angle(50,03)$	94	4	93	3
$\angle(50,78)$	100	4	99	3
$\angle(53,03)$	44	3	44	2

(c) ratio of distances

	female faces		male faces	
distances	ratio	s.d. (σ_r)	ratio	s.d. (σ_r)
d_{02}/d_{32}	1.33	0.08	1.35	0.09
d_{50}/d_{53}	0.75	0.04	0.74	0.03

point matches on the original images (shown in Fig. 3) corresponding to feature points 0 to 5 shown in Fig. 9. Feature point 6 is not visible in the second image and is thus not selected. Because we have already estimated the affine epipolar geometry between the two images, we can compute the affine coordinates of these points. Then using the technique described in Sect. 3 with our knowledge base on female faces given in the previous section, we estimate an affine transformation which brings the affine structure to Euclidean space. Points 0, 1 and 3 are chosen to fix the Euclidean coordinate system. The Euclidean coordinates of the 6 points are shown in Table 2a. The distances, angles and ratios are shown in Table 2b, c and d, respectively, which should be compared to the values given in Table 1.

Table 2: *Results on the recovered Euclidean structure*

(a) Euclidean coordinates (in mm)

point	x	y	z
0	0.0	0.0	0.0
1	122.6	0.0	0.0
2	62.7	47.1	−37.1
3	31.4	88.3	0.0
4	91.5	92.8	−3.9
5	−26.4	19.4	91.3

(b) distances (in mm)

points	distance
$0 \leftrightarrow 1$	122.6
$0 \leftrightarrow 2$	86.8
$0 \leftrightarrow 3$	93.7
$3 \leftrightarrow 2$	63.7
$3 \leftrightarrow 4$	60.4
$5 \leftrightarrow 0$	97.0
$5 \leftrightarrow 2$	158.7
$5 \leftrightarrow 3$	128.2

(c) Angles (in degrees)

vectors	angle
$\angle(01,02)$	43.7
$\angle(01,34)$	5.7
$\angle(02,03)$	41.1
$\angle(32,34)$	61.4
$\angle(50,03)$	95.6
$\angle(53,03)$	48.9

(d) Ratios of distances

distances	ratio
d_{02}/d_{32}	1.36
d_{50}/d_{53}	0.76

Once we have the Euclidean structure of the face, we can specify the desired position in Euclidean space, which is trivial. Figure 10 show six synthesized images corresponding to different rotation angles.

5 Conclusion

In this paper, we have described a technique to build a Euclidean structure from uncalibrated images using fuzzy domain knowledge. We usually have rich knowledge of what surrounding us such as the angles between two sides of a window, the size of a door, the distance between two eyes, etc. In the work described in this paper, affine camera projection model is used. We have formulated 6 types of domain knowledge: location of a point, distance between two points, angle between two vectors, orientation of a vector, a given vector including the length, and ratio of two distances. The knowledge we have is usually about an object category, but not a particular object being considered. The variation (fuzziness) between one object to another is modeled in our

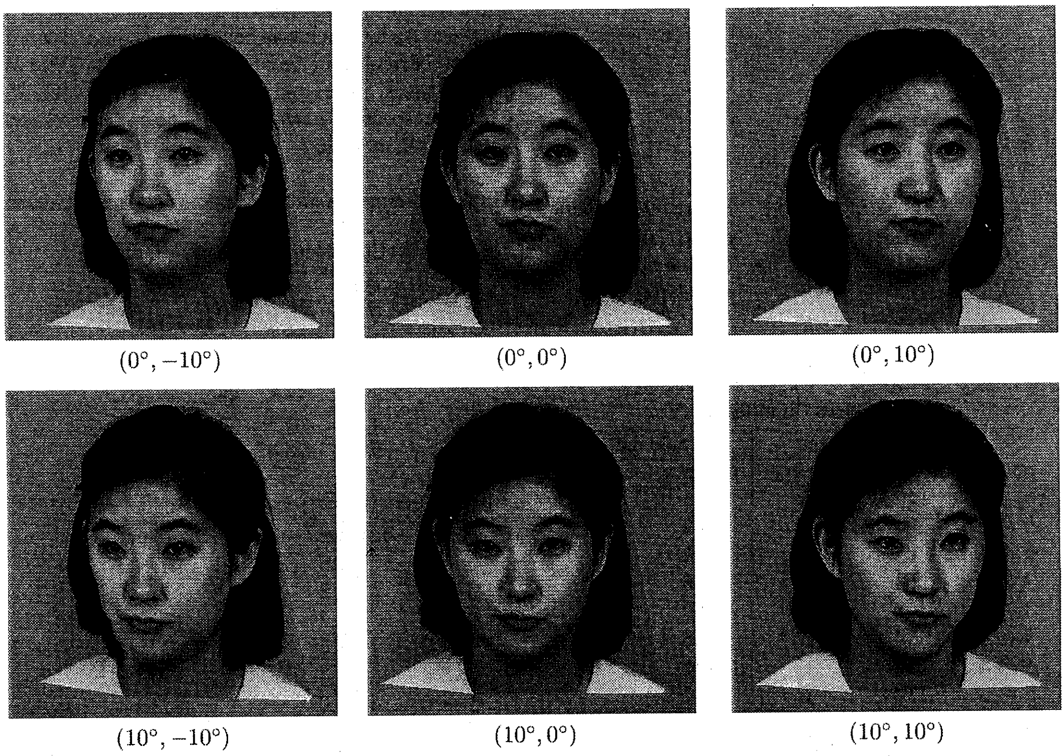


Figure 10: *Synthesized images from recovered Euclidean shape. Rotation angles* (θ, ϕ) *are equal to* $(0^\circ, -10^\circ)$, $(0^\circ, 0^\circ)$, $(0^\circ, 10^\circ)$, $(10^\circ, -10^\circ)$, $(10^\circ, 0^\circ)$, *and* $(10^\circ, 10^\circ)$, *respectively, where* θ *is the rotation angle around the horizontal axis and* ϕ, *around the vertical axis.*

work as a Gaussian variable. The proposed technique can be actually used to calibrate a camera using domain knowledge if the imaged scene contains objects familiar to us.

The proposed technique has been applied to synthesis of new facial images from uncalibrated images. We have described a number of difficulties existing in image synthesis, one of which is the specification of desired position of the new image. Since we can obtain a Euclidean description of a face using fuzzy domain knowledge, the task to specify the desired potion in Euclidean space becomes trivial. We have also proposed to combine points of interest and edges to deal with occlusion, and much better results have been obtained.

References

[1] P. Beardsley, A. Zisserman, and D. Murray. Navigation using affine structure from motion. In *Proc. 3rd ECCV*, volume 2, pp.85–96, May 1994.

[2] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig. In *Proc. 2nd ECCV*, pp.563–578, May 1992.

[3] R. I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *Proc. 2nd ECCV*, pp.579–587, May 1992.

[4] Y. Mukaigawa, Y. Nakamura, and Y. Ohta. Synthesis of arbitrarily oriented face views from two images. In *Proc. 2nd ACCV*, volume III, pp.718–722, Dec. 1995.

[5] J. L. Mundy and A. Zisserman, editors. *Geometric Invariance in Computer Vision.* MIT Press, 1992.

[6] S. Seitz and C. Dyer. View morphing. In *Computer Graphics, Annual Conference Series*, pp.21–30. ACM SIGGRAPH, 1996.

[7] L. Shapiro, A. Zisserman, and M. Brady. 3D motion recovery via affine epipolar geometry. *IJCV*, 16:147–182, 1995.

[8] A. Shashua. Projective structure from uncalibrated images: structure from motion and recognition. *IEEE Trans. PAMI*, 16(8):778–790, 1994.

[9] C. Zeller and O. Faugeras. Applications of non-metric vision to some visual guided tasks. In *Proc. ICPR*, pp.132–136, Oct. 1994.

[10] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 78:87–119, Oct. 1995.