SpotIt! an Interactive Identikit System

R. Brunelli and O. Mich
Istituto per la Ricerca Scientifica e Tecnologica
I-38050 Povo, Trento, ITALY
e-mail: brunelli@irst.itc.it, mich@irst.itc.it

IRST Tech. Rep. #9507-03

Abstract— **In this paper we present a system for browsing large mug-shot databases and the creation identikits of photographic quality. The two functions are interrelated: the available database provides direct feedback to the user building the identikit and the identikit itself can be used as an access key to the image database. SpotIt! provides a virtually unlimited set of alternative features that can be browsed efficiently in the appropriate context, interactive *holistic* feature modification coupled to *syntactic* access to a feature database, and quantitative, automatic computation of face similarities, providing *real-time* feedback of the system which constantly shows the most promising matches to the identikit being built.**

1. Introduction

Eyewitnesses play an important role in the investigative aspects of police work and their descriptions of suspects may help the investigators. When a suspect is unknown to the witness and not in custody, there are two main methods to obtain a description. The first one consists in showing to the eyewitness a set of photographs from the available police archives while the second method is based on the pictorial reconstruction of the facial features of the suspect.

In the first case, spotting of the suspect implies identification and access to useful data on it. On the other side, a pictorial reconstruction of the suspect face, even if very accurate, does not necessarily implies its explicit identification.

The police archives may be very large: thousands of images are typical and they can grow up to hundreds of thousands. Managing these databases using traditional, not computer-based, methods is a time consuming and expensive process. The necessity of assembling a particular selection of photographs into albums for a specific investigation imposes additional burden. The size of the albums to be shown to an eyewitness is one of the major drawback of the first method. Successful scanning of thousands of pictures requires an uncommon ability in keeping one's attention focused for a long period. It has also been shown that the number of misses and false alarms increases with the length of the picture series [1]. Another obvious drawback is the requirement that the suspect be in the photographic police archives. If this requirement is not met, one must resort to the pictorial reconstruction of the suspect.

People are not equally good at recognizing and recalling faces, the latter ability being markedly inferior to the former [2]. Even when a subject has no difficulties in any of the aspects of mental image recall identified by Kosslyn[1][3], he/she may still have great difficulties in making public the recalled mental image.

The pictorial reconstruction of the suspect has then two major drawbacks: the first one is related to the inherent difficulty of recalling a good mental image and the second one to making this description explicit. The interaction of the eyewitness, usually not a professional drawer, with the operator actually constructing the identikit represents a delicate point of the process. The eyewitness must textualize the visual description he/she retains, and interact with the drawer who translates the description back to visual data. This requires good verbal capacity and positive interaction with a skilled operator.

Several systems have already been demonstrated for the construction of identikits and for browsing in an *efficient* way large image databases. CAPSAR (Computer Aided Photographic search and Retrieval [4]), FRAMES [5] and Photobook [6] are systems for searching large image databases. Several systems for building identikits are also available:

- WHATSISFACE [7] is a man-machine system enabling a nonartist to build on a computer display a face using pre-stored facial features;
- Identi-Kit [8], [9] provides a set of alternatives for each face feature (each alternative is drawn on a trasparent sheet for easy composition);
- PhotoFit is similar to the previous system with the exception that the features are represented by pictures;
- Mac-a-Mug Pro and E-FIT [5] are computerized counterparts of Identikit and PhotFit.

These systems are characterized by the common assumption that faces are perceived as a set of independent features. While this assumption simplifies the construction of face synthesis systems it conflicts with reported psycophysical evidence [10], [11] which suggests a more holistic process coupled to a sequential feature comparison [11]. Experiments in face recognition by computers [12] highlight the different dominance of the two processes when the image resolution is changed. The dominant process seems to depend also on the specific face discrimination task. The feature based comparison is itself complex: the facial features seem to have different salience based on the familiarity of the subject with the stimulus. More specifically, external features (e.g. hair and outline) and internal features (e.g. eyes, nose and mouth) seem to be equally good for discriminating among unfamiliar faces, while discrimination of familiar faces *prefers* internal features [13]. Several ex-

[1]They are: locating an image in memory, maintaining the image, *zooming in* on parts of the image and ease of generating an image

periments for establishing feature saliency are reported in the literature (see [11] for a review). The results agree with the following ordering of features by decreasing saliency: face outline, eyes, mouth, and nose. The low saliency of nose can be explained by the use of full-face (frontal) portraits in the experiments. It must also be noted that the agreement on feature saliency may simply reflect the average distinctiveness of the features in an average population of caucasian faces (different orderings could be found for other races). The existence of a configural comparison in the face recognition process may adversely affect the performance of the above mentioned systems which do not allow free adjustments of the features position within the overall frame. Another, probably major, drawback stems from the limited number of alternatives available for each of the features. Furthermore, browsing alternative features out of their original context, that of a full-face, may hinder their appropriate selection. The system we introduce in this paper, SpotIt!, is an attempt at overcoming the many limitations of the available systems by providing:

- a virtually unlimited set of alternative features that can be browsed efficiently in the appropriate context;
- the possibility of acting both at the pictorial feature level and at the configural level of features layout;
- an iconic user-machine interface;
- interactive *holistic* feature modification coupled to *syntactic* access to a feature database;
- user-defined geometric warping;
- quantitative, automatic computation of face similarities, providing *real-time* feedback of the system which constantly shows the most promising matches to the identikit being built.

SpotIt! can be used both for the construction of identikits of photographics quality by a non-artist witness and for browsing large digital photographic albums. The system comprises several modules:

- digital image input;
- image preprocessing (geometric and intensity normalization);
- image coding and database update;
- graphical interface to the database.

The next section will briefly describe the first two modules. The compact coding used by the system for representing faces is then introduced. Finally, the graphical interface to the kernel functionalities of the system is presented.

2. Image Preprocessing

Grey level digital images[2] can be obtained from the photographic archives by using scanning devices. Each image should be annotated with information such as sex, race, age, and identity.

Images managed by the system must have *standardized* size and orientation. Normalization is automatic: the system, after locating the eyes, transforms the digital image by means of an affine transform so that the pupils's position corresponds to predefined values in the image coordinate system [12]. The system also requires the nose and mouth position which are also computed automatically [12] and then submitted to human verification.

Images may derive from many, diversified, sources thereby exhibiting intensity variations which are not directly related to the subject. This variability should be removed, if possible, to obtain a set of homogeneous images. A simple way to normalize image intensity (tonal matching) is to select a reference area and constrain its average value. A possible choice is given by the skin below the eyes [14]. This operation eliminates some variability related to the characteristics of photographic and acquisition processes. A different type of normalization may be required to get rid of influences due to the illumination source (type and direction). A possible choice is to transform a digital image I by computing its local contrast:

$$M = \begin{cases} M' & \text{if } M' \leq 1 \\ 2 - \frac{1}{M'} & \text{if } M' > 1 \end{cases} \tag{1}$$

where

$$M' = \frac{I}{I * K_{G(\sigma)}} \tag{2}$$

where $*$ represents the convolution operation, $K_{G(\sigma)}$ represents a Gaussian convolution kernel whose parameter σ is related to the interocular distance, and the arithmetical operation are to be applied to corresponding image pixels. The resulting image M can be clearly recognized even if *flatter* than the original image I (see Figure 1). Other pre-processing techniques could be necessary in particular cases.

Fig. 1. The effect of intensity normalization by computation of the *local contrast.*

3. Principal components analysis

SpotIt! provides tools for working both at a feature pictorial level and at a geometrical, configural level. In this section we describe how a virtually unlimited set of alternative features can be generated. Each face image can be decomposed into a set of Q regions or features. A possible choice is given by the hair, eyes, nose, mouth, chin (i.e.

[2]SpotIt! does not support color images. The color dimension is a difficult one both to textualize and to use efficaciously in the comparison of visual patterns due to the variability introduced by the photographic/grabbing processes

the inferior part of the face). To each feature F we can associate a region r_F whose size corresponds to the average size of the feature in the available database. Given the set $\{F\}$, the next step is to obtain a compact representation for the images of each facial feature. This representation should be flexible enough to represent the great diversity of facial features yet compact to enable a non-specialist to make effective use of it in *making public* the internal image he/she is able to recall. While a pixel-based representation offers maximum flexibility, it lacks compactness and can impair the process of image explicitation by requiring an high number of operations to achieve the visualized image. The strategy employed by SpotIt! is that of reducing the original dimensionality of images (that corresponds to the number of pixels) by restricting to the *principal components* describing the database of features [15]. Let $\{F_i\}_{i=1,\ldots,N}$ be a set of images of feature F, each image being associated to one of the N images of the database. These images have the same dimensions, shape r_F and are aligned[3]. Each feature image can then be expressed as a linear combination of N images $\{\Phi_j\}_{j=0,1,\ldots,N}$:

$$F_i = \sum_{j=1}^{N} c_{Fij}\Phi_j + \Phi_0 \tag{3}$$

where

$$\Phi_0 = \frac{1}{N}\sum_{i=1}^{N} F_i \tag{4}$$

$$c_{Fij} = (F_i - \Phi_0)\cdot\Phi_j \tag{5}$$

$$\Phi_j\cdot\Phi_t = \delta_{jt} \quad j,t = 1,\ldots,N \tag{6}$$

with usual notation[4]. The images Φ_j con $1 \leq j \leq N$ are chosen as the eigenvectors of the covariance matrix of the set $\{F_i - \Phi_0\}$ [14], [16], [17], [6]. Some of the resulting eigenvectors for eyes are reported in Figure 2. Each eigenvector is related to an eigenvalue λ_{Fj} (see Figure 3). The set of eigenvectors $\{\Phi_j\}_{j=0,\ldots,N}$ can be sorted by decreasing eigenvalue ($\lambda_{Fj} \geq \lambda_{Ft},\ \ j < t$) and represents the *principal components* of the dataset (the representation used in Eq. 3 is also named Karhunen-Loewe expansion). By truncating the expansion of Eq. 3 at $j = k$ we introduce an error Δ whose magnitude decreases when k is incresed. Each image in the database can then be approximately described by the vector of the expansion coefficients:

$$\boldsymbol{c}_{Fi} = \{c_{Fi1}, \cdots, c_{Fik}\} \tag{7}$$

The corresponding image can be reconstructed when $\boldsymbol{c}_{Fi}$ and $\{\Phi_j\}_{j=0,\ldots,k}$ are known. Each face available in the database is then described by a set of vectors, one for each of the used features. The effectiveness of the principal component analysis (i.e. the possibility of using vector of reduced dimensionality for a given Δ) increases if the original database is suitably partitioned (according to race, sex and age) before applying the computations. Computing the eigenvectors is a time expensive process. However, it is not necessary to recompute them every time the database is updated: if it is already sufficiently representative of the face population, a new image can be reconstructed well by using the eigenvectors previously computed. Adding a new image to the database would then imply the sole computation of the expansion coefficients.

[3]The position of a fixed reference point, e.g. the pupil for the eye, is the same for all of the images.

[4]Images are considered as vectors by concatenation of the rows of the original image.

4. Feature construction

SpotIt! enables the user to build the identikit picture by changing the feature appearance through modifications of its Karhunen-Loewe coefficients.

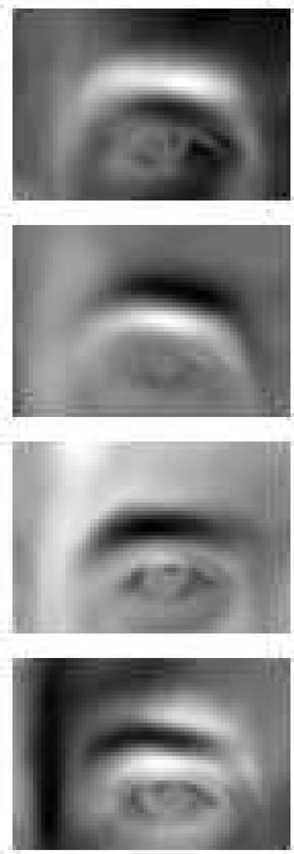

Fig. 2. The eigenvectors corresponding to the largest four eigenvalues. The original images are obtained by adding a linear combination of eigenvectors to the average feature.

The identikit image is built by adding the available features to an average face (the arithmetic average of the database faces). The user can select the feature to work on using a menu. By using an electronic pointer (a mouse or a pen) he/she can modify the position of a set of sliders, each one related to one component of the Karhunen-Loewe expansion. Sliders are ordered top-down by decreasing eigenvalue: the larger the eigenvalue the greater the effect on the reconstruction of the image (see Figure 4). The position x_{Fi} of a slider is represented by a number in the interval $[0, 1]$: 0 corresponds to the extreme left while 1 to the extreme right. The slider position x_{Fi} is associated to a value c_{Fi} for the corresponding coordinate so that $D(c_{Fi}) = x_{Fi}$

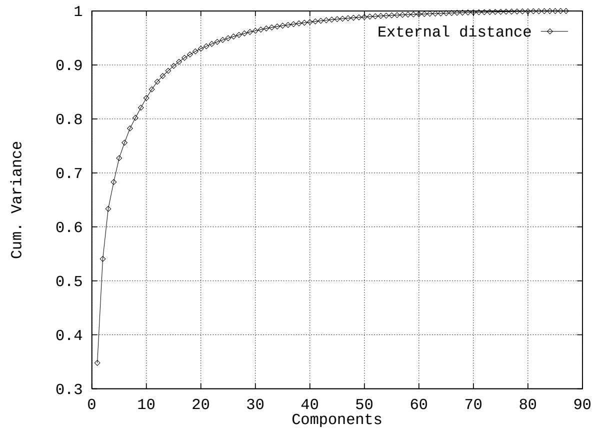

Fig. 3. The cumulative variance ($\sum_{i=1}^{j} \lambda_i / \sum_{i=1}^{N} \lambda_i$) captured by the restricted expansion basis as a function of the number of its components. The number of images used for the analysis was 87.

where $D(\cdot)$ denotes the cumulative probability distribution of the given coordinate in the database. This association maximizes the selectivity of the slider as intervals of the same size cover the same number of database images. The sliders position for each of the available features is stored by the system so that their configuration can be restored whenever the feature is selected for operation. The user can drag each of the sliders or place them at a desired point: the feature is reconstructed and the whole identikit image redisplayed (in *real time*) whenever the position of one of the sliders is changed. The user can also select a feature in a more traditional way by providing a descriptive keyword: the system accesses a database of named features and set the sliders position to the configuration required by the named feature (syntactic feature coding).

The construction of a realistic image by composition of a set of features requires that the region representing each of them be blended with a background face image to avoid annoying artifacts. This suggests the use of regions R_F, greater than r_F, in the computations of Eq. 3, while still using regions r_F in the computation of the principal components. The frame between r_F and R_F is then used for blending the features. Each feature F is associated to a priority value P_F ($P_{eyes} > P_{mouth} > P_{nose} > P_{hair} > P_{chin} > P_{face}$). It is further characterized by a map of *blending* weights W_F whose size corresponds to that of the entire face. This map presents three different regions:

- a region external to the feature, filled with zeros;
- an internal region r_F, used for the computation of the principal components, filled with ones;
- a transition (*blending*) region $R_F - r_F$ whose values belongs to the interval $(0, 1)$.

Let us consider for ease of exposition a 1-dimensional feature:

$$r_F = [x_1, x_2] \tag{8}$$

$$R_F = [x_0, x_3] \tag{9}$$

where $x_0 < x_1 < x_2 < x_3$. The weights of the map $W(x)$

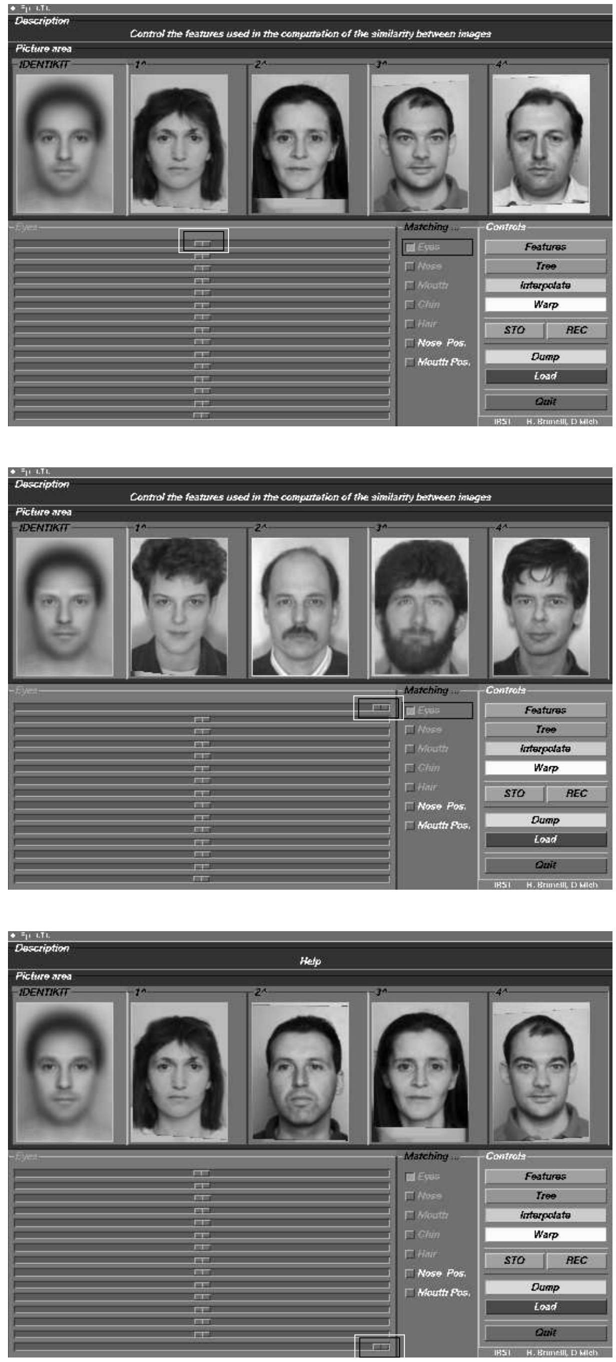

Fig. 4. The sliders are arranged by decreasing effect on the synthesized image. As each slider corresponds to a single principal component, the sorting key is given by the related eigenvalue. Note the different effect on image sorting obtained by moving the first slider (middle image) and the last slider (bottom image). Upper sliders have a major effect on image appearance while lower ones operate more detailed modifications.

are then specified by:

$$W(x) = \begin{cases} 0 & x < x_0 \\ 0 & x > x_3 \\ 1 & x \in [x_1, x_2] \\ \frac{1}{2}\left[1 - \cos\left(\pi \frac{x - x_0}{x_1 - x_0}\right)\right] & x \in [x_0, x_1] \\ \frac{1}{2}\left[1 - \cos\left(\pi \frac{x - x_3}{x_2 - x_3}\right)\right] & x \in [x_2, x_3] \end{cases} \tag{10}$$

The 2-dimensional case is handled by the taking the product of two 1-dimensional weight functions (see Figure 5). Let us denote with $W_1(x,y), W_2(x,y), \ldots$ the maps of the different regions sorted by increasing priority. The value of each pixel $I(x,y)$ of the identikit image in the region modified by the user is computed as:

$$\begin{aligned} I(x,y) &= (((((1 - W_1)F_0 + W_1 F_1)(1 - W_2) + \\ &\quad W_2 F_2))(1 - W_3) + W_3 F_3 \cdots) \end{aligned} \tag{11}$$

This corresponds to building the image by sequential integration of features with increasing priority. It is also important to note that alternative features are built in their appropriate context, that of a complete face, thereby overcoming one of the drawbacks of previously available identikit systems.

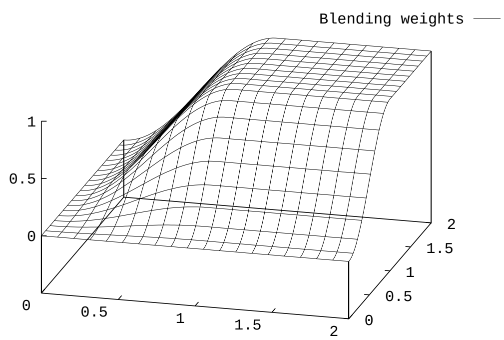

Fig. 5. To obtain a smooth insertion of the different features in the image, the synthesized image is obtained by a weighted average of the feature itself and the available *background* image. The more salient the feature, the later it is inserted in the picture.

The user can select a (sub)set of the available features for comparison with the images of the database. The similarity of the identikit image X with each image I_i of the database is determined by:

$$d(X, I_i) = \sum_{F=1}^{Q} a_F \omega_F |\boldsymbol{c}_F - \boldsymbol{c}_{Fi}|^2 \tag{12}$$

where Q is the number of available characteristics, $a_F = 1$ if the feature has been selected by the user (using the corresponding toggle button in the graphical interface) or 0 otherwise. The weights ω_F

$$\omega_F = \left[\sum_{i=1}^{k} \lambda_{Fi}^2\right]^{-1} \tag{13}$$

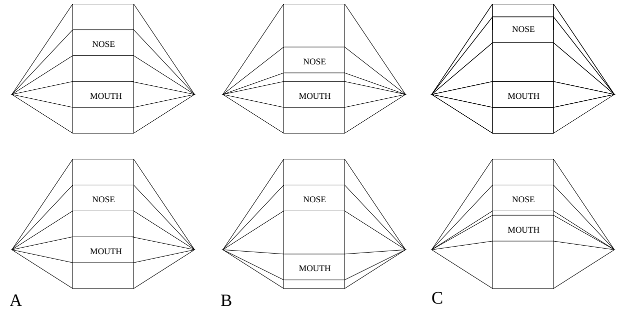

Fig. 6. Some of the features, namely the nose and the mouth, can be moved vertically (inter-feature warping). Their (limited) movement is obtained by warping a set of polygons as depicted in the picture.

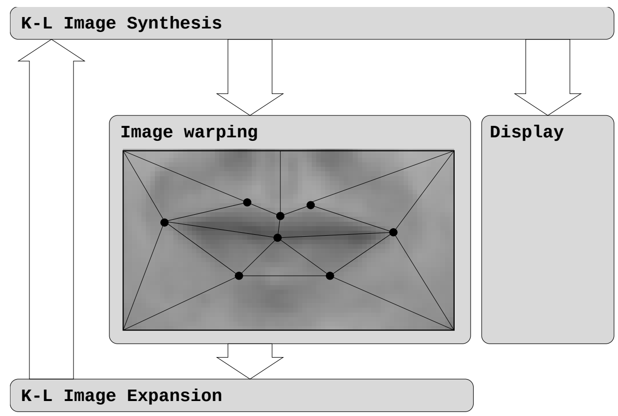

Fig. 7. Features can also be built by interactive modeling of an available picture (intra-feature warping). The user can modify the image by moving a set of predefined control points. These points are feature specific and changing their position warps the image. The resulting image is then expanded on the available Karhunen-Loewe basis and resynthesized before being displayed. This synthesis-expansion loop preserves the possibility of an effective (and efficient) match with the database by comparison of a limited set of coefficients.

are normalizing factors for the different features. They are necessary as the vectors describing the facial features have different characteristic scales: computing a distance without scaling the different contributions would have scarce perceptual meaning. The user can also specify the weights using a set of sliders.

The whole database is sorted by increasing values of $d(X, I_i)$ and the first images are displayed by the graphical interface whenever the identikit image is modified. The remaining images can be accessed with a toolbar. This real-time feedback provided by the system eases the exploration of a mug-shot databases and helps in the construction of the identikit: the user can insert into the identikit the features of any of the displayed database images simply by clicking on it with a mouse/pen. The use of geometric data (i.e. nose length and mouth position) in the computation of image similarity clearly fits into formula 12.

5. Feature Warping and Interpolation

While the use of sliders is particularly effective in the exploration of an image database or to modify a feature that already approximates the visualized internal image, it may not be the most efficient way to obtain modifications clearly textualized or visualized by the witness. Whenever a good textualization of a feature appearance is available, an image can usually be found by matching the description to a set of labelled characteristic features. Support for visualized geometrical modifications is provided by two different type of warping. Some of the features, namely the nose and mouth, can be independently shifted in the vertical direction (see Figure 6). All of the features can be modified by warping them using predefined decompositions into triangles and quadrilaterals (see Figure 7 and [18] for the actual warping techniques). Both warping, intra and inter feature, are valuable in the process of creating an identikit. However, their *nature* differs considerably. The warping *inter features* does not change the expansion of the given features onto the principal components. Intra feature warping does modify the expansion coefficients and poses a problem for the feedback mechanism of the system. The adopted solution is sketched in Figure 7. The system tracks the intra feature warping of a feature, computes the expansion coefficients of the warped image and displays to the user the resulting reconstructed image.

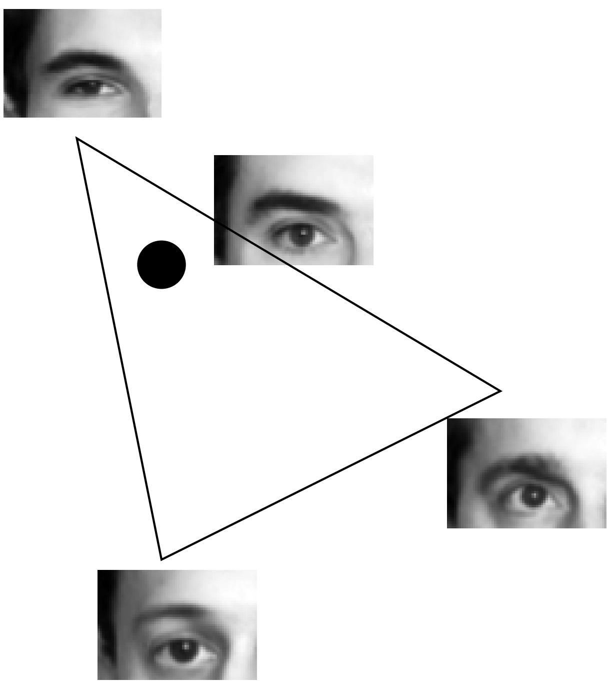

Fig. 8. The user can select up to three different features and interpolate between them by moving a probe in the linear space delimited selected examples. This can be useful to explore the space defined by a small set of equally good candidate features.

Sometimes, two or more feature instances shown by the system can be considered as equally good by the user. In such cases it is possible to interpolate between the two or three most promising candidates. The user can move a point in the space delimited by the selected candidates (which is simply a segment or a triangle identified by the vectors $\{\boldsymbol{c}_k\}$ corresponding to the selected features, see Figure 8).

6. Archive navigation

In the previous sections we have described the features of SpotIt! for building identikits. The other major functionality of SpotIt! is that of database browsing. A first browsing modality is given by the feedback mechanism attached to the construction of the identikit image. The identikit can be used as a key to access the database. Each feature can be considered as a user selectable field. The result of the query is a list of images sorted by similarity with the access key. The database can also be explored by moving along a tree structure computed by hierarchical clustering of the available images. Activation of this modality disables the previous (default) one. The user must select a (sub)set of features for computing image similarity (using $d(I_i, I_j)$ from Eq. 12). The database is then partitioned into 4 clusters. Each resulting cluster is then recursively partitioned into 4 clusters: the recursive clustering stops when the number of images in a cluster is less than 4. A standard algorithm such as the Linde-Buzo-Gray algorithm can be used (but see [19] for a review of clustering techniques). Each cluster is then represented by the element nearest to its center. The branches of the resulting tree lead to faces which are more and more similar to one another. The user can move along the tree by using up/down arrows in correspondence of each cluster representative (see Figure 9). The system also shows for each of the representatives its distance from the available identikit image, thereby guiding the user exploration. Clicking on any of the displayed images causes its feature to be copied onto the identikit image and makes the system return to the default browsing modality, i.e. the identikit feedback mechanism.

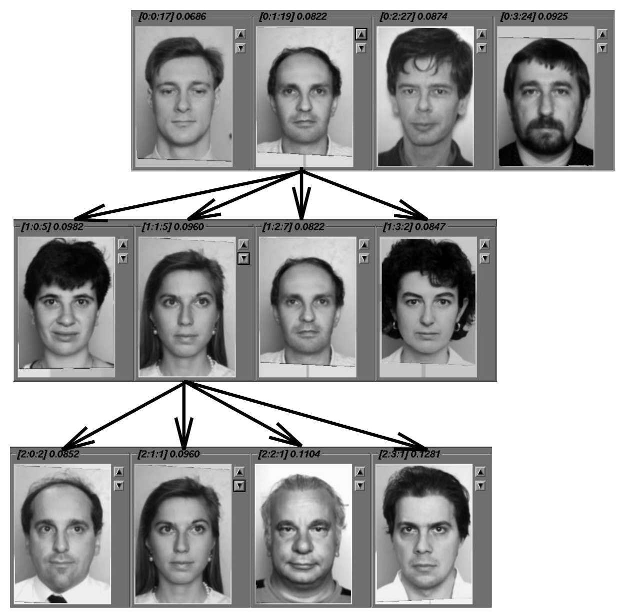

Fig. 9. The database is organized hierarchically to make the exploration easier. In the current system, clustering by a tree with branching factor 4 is employed. For each node the system presents to the user the centroids of the corresponding cluster and their distance from the current identikit image. Eyes were used in this example for the clustering process.

Fig. 10. The figure shows the reconstruction of one of the people in the database.

7. Conclusions

In this paper we have described SpotIt!, a system for creating identikits and for browsing large image archives of mug-shots. The system overcomes the major drawbacks affecting available computerized systems, providing unlimited number of feature variants and a quantitative, automatic estimation of image similarity. The system described in this paper is now a prototype and some of the functionalities are missing. Several police departments have already expressed interest and the system is waiting for a field test.

References

[1] G. M. Davies, J. W. Shepherd, and H. D. Ellis. Effect of interpolated mugshot exposure on accuracy of eyewitness identification. *Journal of Applied Psychology*, 64:232–237, 1979.

[2] H. D. Ellis. Practical aspects of face memory. In G. Wells and E. Loftus, editors, *Eyewitness testimony.* Cambridge University Press, 1984.

[3] S. M. Kosslyn. *Ghosts in the mind: Creating and using images in the brain.* Norton, 1985.

[4] D. R. Lenorovitz and K. R. Laughery. A witness-computer interactive system for searching mug files. In G. R. Wells and E. F. Loftus, editors, *Eyewitness Testimony. Psychological Perspectives.* Cambridge, England: Cambridge University Press, 1984.

[5] J. Shepherd and H. Ellis. Face recognition and recall using computer-interactive methods with eye witnesses. In Vicki Bruce and Mike Burton, editors, *Processing Images of Faces*, chapter 6, pages 129–148. Ablex Publishing Corporation - Norwood, New Jersey, 1992.

[6] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook: Tools for content-based manipulation of image databases. In *SPIE, Storage and Retrieval Image and Video Databases II*, 1994.

[7] M. L. Gillenson and B. Chandrasekaran. A Heuristic Strategy for Developing Human Facial Images on a CRT. *Pattern Recognition*, 7:187–196, 1975.

[8] V. Bruce. *Recognising Faces*, chapter Faces as Patterns, pages 37–58. Lawrence Erlbaum Associates, 1988.

[9] K. R. Laughery and R. H. Fowler. Sketch Artist and Identi-Kit. Procedure for Recalling Faces. *Journal of Applied Psychology*, 65(3):307–316, 1980.

[10] A. Young and D. Hay. Configurational information in face perception. Experimental Psychology Society, London Meeting, January 1986.

[11] R. Brunelli. HEBO. Contract Report #8911-19, IRST, 1989.

[12] R. Brunelli and T. Poggio. Face Recognition: Features versus Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, 1993.

[13] H. D. Ellis, J. W. Shepherd, and G. M. Davies. Identification of familiar and unfamiliar faces from internal and external features: some implications for theories of face recognition. *Perception*, 8:431–439, 1979.

[14] M. Kirby and L. Sirovich. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, 1990.

[15] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1972.

[16] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[17] A. J. O'Toole, H. Abdi, K. A. Deffenbacher, and D. Valentin. Low-dimensional representation of faces in higher dimensions of the face space. *J. Opt. Soc. Am.*, 10(3):405–411, 1993.

[18] Z. C. LI, Y. Y. Tang, T. D. Bui, and C. Y. Suen. Shape transformation models. *Int. Jour. of Pattern Recognition and Artificial Intelligence*, Vol. 4(1):65–94, 1990.

[19] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

This page intentionally left blank