8-1993

Facial image retrieval, identification, and inference system

Jian Kang WU

Yew Hock ANG

Chiam Prong LAM

S. K. MOORTHY

Arcot Desai NARASIMHALU
Singapore Management University, desai@smu.edu.sg

DOI: https://doi.org/10.1145/166266.166271

Citation

WU, Jian Kang; ANG, Yew Hock; LAM, Chiam Prong; MOORTHY, S. K.; and NARASIMHALU, Arcot Desai. Facial image retrieval, identification, and inference system. (1993). *Multimedia '93: Proceedings of the 1st International ACM Conference on Multimedia, Anaheim, California, August 1-6, 1993*. 47-55. Research Collection School Of Information Systems.
Available at: https://ink.library.smu.edu.sg/sis_research/3263

Facial Image Retrieval, Identification, and Inference System

J. K. Wu*, Y. H. Ang, P. C. Lam, S. K. Moorthy, A. D. Narasimhalu
Institute of Systems Science
National University of Singapore

Abstract

Recognition of a human face is very easy even to a child, but is extremely difficult for computers. Here we present a Computer Aided Facial Image Identification, Retrieval and Inference System (CAFIIRIS) for criminal identification. The system stores and manages facial images and criminal records, providing necessary image and text processing, and editing tools. Inference of facial images of different ages of a person is also possible. Access to facial images can be done via key words, fuzzy descriptions, and visual browsing.

A facial image database system stores and manages a large amount of facial images together with text-based criminal record. It provides users with a flexible means to manipulate, archive, retrieve, and make use of facial images and text data. The facial images are visual rather than descriptive. Each digital image is a large array of pixels of various sizes, and a facial image database contains thousands or even hundred thousands of images. Therefore, this huge visual database needs special techniques for its management, namely, embedded functions for image pre-processing, feature extraction, presentation (screen display and report formatter); visual access to image data via special indexing techniques; application-specific image inference to derive new images based on images and other available information.

1 CAFIIRIS system

A block diagram of CAFIIRIS is shown in Fig. 1. In the first row of Fig. 1 are modules providing resource and environment management to the system. The system deals with not only storage and retrieval of facial images, but also facial image processing and inference. Therefore the system provides a common work space (like cache memory for database) to hold facial images, criminal records, and facial components. The work space acts as *communication blocks* between various processing stages, and as buffers for processing modules and to line up facial images.

* email: jiankang.iss.nus.sg, Tel: +65 772 2544

As soon as a user logs in, a *session management* is started. It allocates database views and functions to users according to their priority and creates a log file to keep a record of all activities for system statistics and for rollback in case of a disaster. Object communication becomes very easy because all functions available to users are all well defined by *function management*. When a user invokes a high level function, a sub-process can be created. That means a user can gain access to many functions simultaneously. User interface is implemented via session management.

The system consists of five functional subsystems or modules as depicted in the second row of Fig. 1 and will be described in following subsections.

1.1 Image Preprocessing and Feature Extraction

The image preprocessing and feature extraction module provides various input and processing functions on the complete form of facial image data (composed of social record, feature measure, facial image data) and criminal records, including image digitization, enhancement, segmentation, feature extraction, image pyramid structure (hierarchical multi-resolution) construction, compression, social records input and editing, viewing the features and descriptions, display images, perform facial image classification and produce facial classification tables etc..

A complete form of facial image data consists of four parts: attribute of the person and image, facial description, facial feature measures, and facial image data. The key issue here is feature extraction and mapping between feature measures and facial description. The difficulties [1] lie on the following facts: It is not clear so far how humans recognize faces, and what features are used by humans to distinguish between faces. There does not exist an image processing method which can readily solve our problem. Subjective descriptions of face characteristics are fuzzy. The mapping between subjective descriptions and feature measures should be derived so that the access from descriptions are possible.

By working together with domain experts, a visual description form for facial images is defined. It contains 6 descriptive slots: facial outline, hair, eyes, eyebrows, nose, and mouth. Each descriptive slot contains items and possible descriptive values (such as large, middle, small, etc.) for those items. To be consistent with this visual description form, image feature extraction

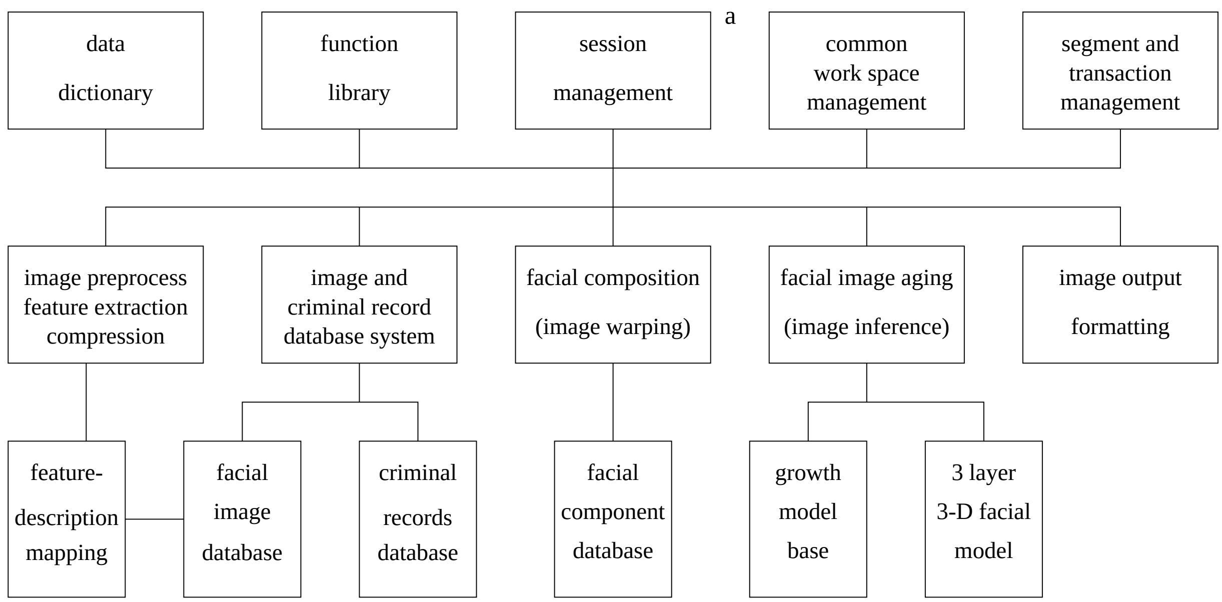

Figure 1: Block diagram of Computer-Aided Facial Image Identification, Retrieval and Inference System (CAFIIRIS)

is carried out on whole images as well as facial components to generate features on these 6 slots. We use face landmarks and selected principal component analysis coefficients of face images to generate feature measures. To reduce the effects of image variations, a normalization process is performed before feature extraction.

1.2 Facial Composition

For the photo-fitting module, various facial components are selected from the facial component database to fit into a facial image. When a facial template image is first selected, it is modeled by a 2-D wireframe. As compared to the selected facial components the original facial components of the template face are of different sizes, shapes and orientations. Hence, when fitting a new facial component onto the template, the nodes of the wireframe will be moved to accommodate the facial component. As the result of the movement of the wireframe, the underlying image will be warped [5] to fit the new positions of the wireframe nodes. Finally, the edges where the new facial components meet the template image are blended together. Two images with different eyes, nose, and mouth generated by the facial composition sub-system are shown in Fig. 2.

1.3 Aging

Given a facial image of a particular person, aging is to construct facial image of the same person but at a different age. Obviously aging is an inference problem, and can be solved based on human growth model, and using expert system techniques.

The aging process is a slow motion of the event in the time scale taken over a long period [2]. The skeletal element growth and soft tissues changes are subjected to the internal as well as external factors, and causes to perceive the facial growth. There is a need to understand the mechanism of growth and to implement a model to simulate it. So far, no attempt has been made to model the human face and impart the growth aspects in it from the engineering point of view. To obtain approximate growth, we proposed to implement a 3-d facial model to represent the complete cranio-facial complex, in a three layer (bone structure, soft tissue, skin) approach. A reference database of norms for different age groups is created. The modeling process is decomposed into two steps. First, a model corresponding to a given age group is selected. Second, the model is fitted to the specific person using the given 2-D image(s). Parameters from 2-D image(s) are derived to make the 3-D model specific to the person. More expert knowledge is used together with personal data to inference the effect of occupation and personal characteristics on the 3-D three layer model parameters. Due to the nature of parallel execution of inter functional units of the model, the modeling process is parallel and distributed.

1.4 Output Image Formatting

The facial image output formatting module is mainly designed to produce hardware copy of a report of a facial identification activity. The output can consist of facial images, social records, criminal records, and description of processes, which may include the date of the process, the operator's name, the details of the processes (mainly facial composition and aging). Therefore, the module provides a means to interactively edit the descriptions, put images in desired position, add frames, legends, coloring, etc.. The results are either displayed on the screen, stored in disk files, or output to image output devices.

Figure 2: Facial images derived by facial composition subsystem

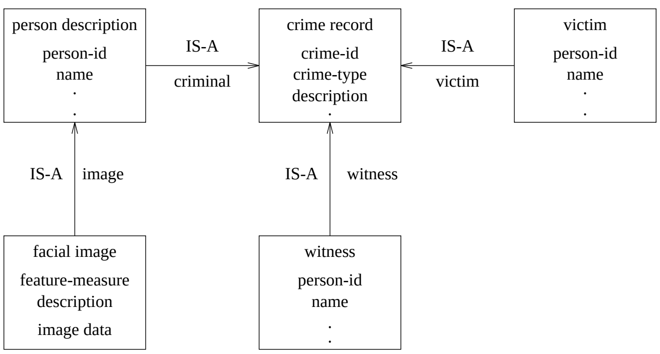

Figure 3: Database schema of CAFIIRIS by frame mechanism

1.5 Facial Image and Criminal Record Database System

Facial images and criminal records data can be represented using frames and their hierarchy shown in Fig. 3. It is an object-oriented knowledge representation scheme. There are five frames describing five data items: personal data, facial image and their features (measures and description), criminal records, victim, and witness. The relationship between these frames are all "IS-A". By proper definition and implementation of frame mechanism, data processing functions such as automatic feature extraction, display, can be all embedded in the model. Here the association between frame slots and functions are built by both schema definition and function library definition.

The person identified via his/her facial image can be a criminal of some crimes recorded by criminal records in criminal record database. The criminal records related to the identified person can be immediately retrieved via "IS-A" links. Similarly, when a criminal record is identified, related persons can be retrieved via the link.

2 Iconic Index of Facial Images

The requirements of facial image queries can be summarized as follows: 1) the query should be visual, providing views of facial image database at every position, so as to enable visual navigation. 2) querying by fuzzy descriptions should be possible. There are 3) retrieval based on similarity measures should be a basic means to access the facial image database.

These three requirements, namely, visual, fuzzy and similarity queries cannot be implemented by conventional indexing techniques such as B-trees and inverted file. For the purpose of defining visual, fuzzy and similarity indexing, we first extend the concept of indexing with abstraction and classification. A spatial self-organization neural network combining Kohonen's MAP [3] neural model and Wu's LEP model [6] is used to generate spatially self-organized index tree. To facilitate fuzzy queries multi-variate fuzzy membership functions are then used to build links between feature measures and facial descriptions. Finally, icon images for non-leaf nodes of index trees are created and added to visually navigate users in queries.

Now we give a brief description on how to use the iconic index tree to achieve visual query, fuzzy query and similarity retrieval.

The similarity retrieval is the basic retrieval operation of this index tree. The index tree can be regarded as a decision tree. When a sample facial image is presented, the similarity retrieval behaves in the same way as pattern classification via a decision tree. It follows the tree down to the leaf nodes. At each level the decision is made by similarity measure. At the leaf node level all leaf nodes similar to the sample image will be selected.

Visual query of facial image database is done by visually browsing through the database via interactive index tree traversal. The system presents the user with the root of index tree by displaying the icons of its descendants. At a node of the index tree, the user chooses browsing directions: up, down, left, right, zoom in, and zoom out.

Within the records of nodes there are feature vectors and descriptions. Both feature vector and description of a node are consistent. Therefore index tree traversal can also be carried out by description matching. Queries are defined in the forms of description syntax, which can be either complete or incomplete. In the case of a completely defined query, the query processing goes down along the index tree until the leaf nodes, which represent the right facial images. At each node of the index tree, a descendant which is the closest to the query is chosen as a next node along the searching path.

If a query is not completely defined, or defined in terms of fuzzy description. The query can be processed to reach a intermediate node of the iconic index tree via fuzzy reasoning. The query is then be continued by visual browsing to find the right answer.

Social records are chosen as accessing perspective for filing purpose. A QBE like user interface is developed.

2.1 Indexing by Abstraction and Classification

Generally speaking, an index consists of a collection of entries, one for each data item, containing the value of a key attribute for that item, and a reference pointer which allows immediate access to that item. Most database systems use tree indexing mechanism to accelerate searching of specific data items. The intermediate nodes in an index tree are abstractions of their child nodes. An index tree can be constructed either through abstraction in a bottom-up process, or through classification in a top-down process.

In conventional databases, 1) an index tree is generated using only one key attribute, 2) key attributes are of primitive data types, such as string, integer or float, and 3) the grouping criterion is usually primitive logical expression such as $\theta_1 \leq a_j < \theta_2$. We shall make three generalizations so as to define the indexing mechanism for facial images, hopefully it is general enough for other kinds of data items.

The first generalization is to let attribute A_j be an abstract data type. It can be a vector (in our case, it is a feature vector), a multi-dimensional array (multi-dimensional feature vectors, image data, or image sequences), or a pointer to a data structure (in our data schema, facial image frame appears as an abstract data type in person frame).

As a result of first generalization, the grouping criterion and grouping function are by no means primitive logical expressions. The grouping criterion should be then generalized (second generalization) to be similarity measures in the case of vector data type. For a data structure like facial image frame it should be further generalized to be a well defined set of functions which extract and measure the similarities between images. Because the attributes now are multi-dimensional, spatial self-organization of child nodes of an abstract data item node, and automatically grouping to balance between nodes in terms of population are necessary.

The third generalization is that for the different levels of the index tree the key attributes can be different. For example, at the top level of the facial image index tree the key attribute is the facial outline feature vectors, and at the second level it is hair feature vectors, and so on.

Iconic index works in a 2-D visual environment, it requires that the nodes are 2-D spatially arranged such that visual browsing of images is possible.

2.2 Implementation of Iconic Indexing

Indexing can be considered as subsequent operation of categorization. With our feature extraction method we extract a set of feature vectors for all six descriptive items: face outline, hair, eyes, eyebrows, nose, and mouth. We refer to the feature measures for these descriptive items as *feature aspect*. Evaluation of feature extraction methods and classification of facial images are done against expert visual categorized sample images.

The algorithm to build such iconic index tree can be either top-down (split) or bottom-up (merge). A top-down algorithm is summarized as follows.

1. Select a feature aspect, cluster the facial images into m classes by an extended self-organization neural network. The weight vectors in the neural network are then used as reference feature vectors of the selected feature aspect. An iconic image is constructed for each node.
2. Repeat the first step until each node has at most m descendants.
3. For each node, fields are allocated for following items:
   (a) Node type, 0 - root, 1 - branch node, 2 - leaf node.
   (b) Feature vector.
   (c) Iconic image.
   (d) Pointer to its parent node.
   (e) Number of sons, a list of pointers to sons.
   (f) Pointers to neighbor nodes which are slightly different in certain feature aspects, which have been already used for clustering at levels above. For example, if a index tree is created using facial outline, eyes, hair, nose in top-down process, we are now at the third level (hair). All images under any node at this level are supposed to be similar with respect to facial outline, eyes, and hair features. In case that one might be satisfied with an image retrieved at this level, but want eyes little bit bigger, horizontal link to images with slightly different eyes is very useful. This link is aimed to provide users with multi-dimensional views of the facial images with respect to feature aspect.

   Not many facial images have outstanding features. One class is dedicated to faces having outstanding features. Outstanding features are described by two fields: type and position. Null type represents normal face. Facial images with the same type and position of outstanding features are then indexed the same way as normal face images. The node structures is depicted in Fig. 6.

2.3 Fuzzy Membership Functions

Facial image features are multi-dimensional. Firstly there are 6 dimensions of feature aspects, secondly there are many dimensions for each feature aspect. Indexing is to build a path which navigates the users to a desired destination in a 6 dimensional feature aspect space. In this subsection, we deal with dimension of feature vectors of each feature aspect. To reduce the complexity of the problem, we use fuzzy membership functions to convert multi-dimensional feature vector space into a limited number of fuzzy sets. After fuzzification the indexing feature space is 6 dimensional and discrete. This reduces the access complexity and enable image retrieval by fuzzy descriptions.

Multi-dimensional fuzzy membership functions [7] relate facial description items to feature measures. Definition of those multi-dimensional fuzzy functions are based on domain experts knowledge. A set of sample facial images are selected which can well represent classes of all facial images in consideration. Facial images are visually classified by domain experts according to descriptive definition. For example, face outline are described in terms of 7 categories, which are described by fuzzy sets. they are oval, pointed, rounded, tapered, squared, double chin, bony cheek. Feature measures are also extracted for all sample images. Within each feature vector space of a feature aspect, the histograms of categories are then used as corresponding fuzzy membership functions. Note that the fuzzy membership function defined this way do not necessarily need to be convex, but they are normalized and overlap each other.

As the fuzzy subsets are defined in a multi-dimensional space, they do have certain spatial arrangement. This spatial relationship between fuzzy sets should be well preserved so that the neighborhood of fuzzy subsets is meaningful, and that with fuzzy query definition, we can easily locate the position of the query in the feature aspect space. That is the reason we use spatial self-organization neural network to create nodes of the index tree. Fuzzy similarity measures are discussed in [4].

2.4 Spatial Self-Organization Neural Networks to Create Indexing Trees

Self-Organizing Map (SOM, also called Topological Map) was first introduced by Teuvo Kohonen [5] and has been developed as an effective neural network paradigm for unsupervised or competitive learning. It is well suited for creating spatially organized internal representations of various features of input signals and their abstractions.

The index tree of CAFIIRIS should show category consistency between feature measures and descriptions. Feature measures are internal and serve as a criterion for index tree construction. Therefore, the similarity measures here perform very important role. The neural network model, Learning based on Experiences and Perspectives (LEP) suggests combining multi-feature perspectives to achieve reliable learning. We have developed a spatial self-organization neural network for index tree construction based on SOM and LEP.

The network has two layers. The input layer receives input vector therefore has the same number of units as input feature vector dimension. The output units are arranged as a two-dimensional array and the number of units are greater than the number of fuzzy subsets. That is to say that a fuzzy subset may be represented by several output units. Suppose there are M input units and N output units in the network. Each input unit is connected to every output unit with a certain synaptic weight $\{w_{mn}, m = 1, 2, ..., M; n = 1, 2, ..., N\}$. For an output unit n a template vector $\{p_{mn}, i = 1, 2, ..., M\}$ and a weighting vector for input feature vector to define relative importance of their elements $\{r_{mn}, m = 1, 2, ..., M\}$ are stored. They will be matched against input vectors during learning. Both link weights and template vectors are long term memory items and stored as two link weights from input units to output units.

Let $\mathbf{x} = (x_1, x_2, ..., x_M)^T$ be the M-dimensional real input feature vector presented to the input array at time $t = 1, 2, 3, ...$. Then the output units begin to compete each other. The winning unit c is selected based on both correlation and minimum distance basis:

$$\begin{aligned} a_c &= \min_n a_n = dis(\mathbf{x}, \mathbf{p}_n)/cr^k(\mathbf{x}, \mathbf{w}_n) \\ dis(\mathbf{x}, \mathbf{p}_n) &= [\sum_m (x_m r_{mn} - p_{mn})]^{1/2} \\ cr(\mathbf{x}, \mathbf{w}_n) &= abs(\sum_m x_m w_{mn})/[\sum_m x_m^2 \sum_m w_{mn}^2]^{1/2} \end{aligned} \quad (1)$$

where k is the parameter to adjust the effect of normalized correlation to whole similarity measure.

Similarity measures are essential for image indexing and retrieval. Usually in neural network evolution weighted summation is used to get the total input for an unit. This is actually the correlation between input pattern and the weight vector. Let us now see the result of correlation without normalization as shown in Fig. 4 (a). It is obviously affected by the cosine of the two vectors and the norm of these two vectors. If the cosine between two vectors is zero then their correlation is zero as well. This is shown as a line across origin and perpendicular to the vector. The equal correlation contours are lines parallel to this zero correlation line. As the length of vector becomes large, the correlation becomes large as well. It is proportional to the vector length. When a threshold is set for the correlation between two vectors, for example, by $correlation(\mathbf{x}, \mathbf{w}) < t$ in a 2-D space a half plane is defined which is to the left side of constant correlation line $correlation = t$. Therefore the pattern space is not uniform for correlation measure, and it does not properly reflect any similarities between patterns.

Note that the problem of correlation measure is mainly due to the norm of two vectors. Normalized correlation measure eliminates the effect of vector norms. It is just the cosine between two vectors (see Fig. 4 (c)). Therefore the normalized correlation measures are defined on the unit sphere. It does not spread to the whole space.

Distance measure defines similarity by the distance metric in the pattern space. Patterns are similar if they are near each other in the pattern space. Distance measure is widely used in the field of pattern recognition to group patterns near each other to one class. Although differences and similarities are well reflected in the distance measure, the constitution of patterns with their components is not well represented. Two orthogonal vectors may be quite near, but they are totally different with respect to their constitution, which is measured by normalized correlation. The distance measure is depicted in Fig. 4(b).

Here we combine normalized correlation and weighted distance to form our similarity measure in equation 1. If two vectors are correlated the normalized correlation measure is 1 and there will not be any effect on weighted distance. Otherwise, the normalized correlation will be less than 1 and enlarge the weighted distance. The contribution of normalized correlation to the similarity measure is controlled by parameter k. This similarity measure is shown in Fig. 4(c).

To show the effectiveness of this similarity measure, a test was carried out for distance measure only and for our combinatorial measure. In the test the same input image data and the same network parameter were used. The results obtained only differ

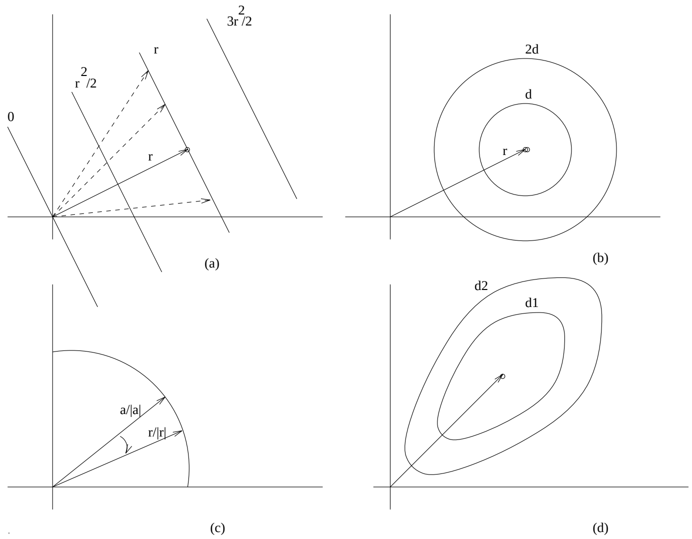

Figure 4: Similarity measures: (a) correlation, (b) distance, (c) normalized correlation, (d) combinatorial measure of distance and normalized correlation.

in two categories. As shown in Fig. 5 it is obvious that the result obtained using our combinatorial measure should be more consistent with our visual perception.

After a winning unit is selected, all the units within its neighborhood are updated. Let $N_c(t)$ be the neighborhood around unit c at time t. N_c is usually set very wide in the beginning to acquire a rough global order, and then shrunk monotonically with time in order to improve the spatial resolution of the map. This procedure is crucial for the topological ordering. The weight vector updating formula is

$$\mathbf{p}_i(t+1) = \begin{cases} \mathbf{p}_i(t) + \alpha(t, d_i)[\mathbf{x} \cdot \mathbf{r}_i - \mathbf{p}_i(t)] a_c \beta(e_c) & i \in N_c(t) \\ \mathbf{m}_i(t) & \text{otherwise} \end{cases} \quad (2)$$

where $\alpha(t, d_i)$ is an *adaptation gain* between 0 and 1 and decreases with time. It is a function of both time t and distance d_i, and usually has the shape of Gaussian function. As d_i becomes large the update becomes smaller.

a_c represent the confidence measure of the winning unit. If a_c approximates 1 the unit wins with high confidence therefore the template update can be large. $\beta(e_c)$ is a function of experience record e_c, which counts the number of times the unit has won. $\beta(e_c)$ is inversely proportional to the experience record e_c. If an unit has experienced a lot of learning the templates should not vary too much. On the other hand experience records are attenuated by so-called forgetting function, which is a simulation of human beings forgetting phenomena. Forgetting enhances the adaptability of the neural network. The forgetting function takes the form of exponential form $e_c exp^{-\gamma t}$.

Similarly the update of correlation weights is defined as

$$\mathbf{w}_i(t+1) = \begin{cases} \frac{\mathbf{w}_i(t) + \alpha(t, d_i) a_i \beta(e_c) \mathbf{x}(t)}{||\mathbf{w}_i(t) + \alpha(t) \mathbf{x}(t)||} & if \;\; i \in N_c(t) \\ \mathbf{w}_i(t) & otherwise \end{cases} \quad (3)$$

Figure 5: Two categories obtained with distance measure (the first three and last two) and combinatorial measure (the first two and last three)

As a result of competitive learning with dynamic neighborhood window, the weight vectors (templates) tend to approximate the probability density function of the input vectors in a spatially ordered fashion.

2.5 Iconic Images Construction

As defined previously, an intermediate node of the index tree is a abstract facial image. There is no direct pointer to any actual facial image in the database system. On the contrary, it represents a set of facial images pointed by its child leaf nodes. Therefore an abstract facial image icon should be constructed which must be the abstraction of all those actual facial images.

An immediate way of constructing the abstract icon image is to use the mean facial image. Because at each level of the index tree the clustering was done using one feature aspect, the descendants of a node are only similar with respect to this feature aspect. Assume that at the current level only facial shape feature has been used for clustering, then the facial images belonging to a node are only similar with respect to facial shape, while other features, such as hair, may show large diversity. In this case, the icon image generated by averaging over those images will have distortion on hair region. Two alternative methods can be adopted. One is to find a facial image which is closest to the template, and use it as icon image. The other is to average over the region with respect to feature aspects which have been already used for clustering so far, and other regions of the icon image are taken from the image which is closest to the template.

2.6 Handling Special Features

Quite often people are identified by special features, such as mole, and scar. Facial special features are described by three attributes: type, size and position. All the three attributes are descriptive to the users for friendly interaction. Types are names of special features, sizes are relatively described by “large”, “average”, and “small”. The position is in terms of the facial component the special feature is closest to. The internal representation of size and position are numerical and in terms of area and distance from the facial component.

When dealing with facial special features, some facts must be considered. First, not all people have special facial features. Therefore special feature cannot be used as indexing keys in the same way as other features. Second, The attributes of special features should be treated differently. The definition of "type" is certain, so retrieval by special feature types is a sort of exact match. On the other hand, the descriptions of the size and position of the special features by users are fuzzy. They should be converted into internal numerical measures according to fuzzy membership functions. For each type of special features there is a set of fuzzy membership functions defined for fuzzy variables "large", "small", "average", "close to mouth", etc..

To speed up image retrieval, an index tree is built for all those facial images which have special features. The index tree is again an iconic index tree with the special feature being the first feature aspect. When facial image retrieval is performed using this iconic index tree, the final decision is made based on the similarity function as follows

$$S(\mathbf{t}, \mathbf{p}) = \prod_i s_{exact}(t_i, p_i) \sum_j s_{sim}(t_j, p_j) w_j \qquad (4)$$

where $s_{exact}()$ is the exact match between feature aspects of template $\mathbf{t}$ and the stored pattern $\mathbf{p}$. It takes values either 0 or 1. Any mismatch between these feature aspects will reject the hypothesis. $s_{sim}()$ is the similarity measure between feature aspects which do not require exact match. w_j is weight for i-th feature aspect. The overall similarity measure is the weighted summation of similarity measures of those individual feature aspects. A similarity retrieval example is shown in Fig. 7. where higher weight is put to facial outline. Other facial aspects are ranked as eyes and eyebrows, hair, nose and mouth. The actual weights for these feature aspects are 9, 3, 1, 0.8, 0.5. The system can be tuned by varying these weights.

2.7 Performance of the Iconic Index Tree

The advantages of the iconic index tree are its ability to provide user with visual navigation, similarity retrieval and fuzzy query which are essential to image databases, and which conventional indexing methods cannot provide.

Usually evaluation of indexing methods are carried out based on two criteria: memory space cost and access cost. Iconic index tree is constructed in a multi-dimensional feature space, and the nodes of the tree represent clusters in the feature space. It actually converts a multi-dimensional problem to one dimensional cluster, and consequently simplifies the problem a great deal. Suppose we are given a M dimensional data. With conventional indexing methods, index trees should be created for all M elements of the feature vector. The memory space cost is certainly much larger than iconic index. It is extremely costly if similarity retrieval is processed using conventional index tree.

As in the definition, the iconic index tree uses abstract data type as indexing attribute, with the discriminant function based on similarity measures as construction criterion. This leads to the generalization of indexing mechanism. Usually the iconic index is constructed against a large data set, which is the representation of data statistics. Creation of index is the learning phase. When the index tree is constructed, data insertion in the index usually does not change the node characteristics, otherwise an insertion will require the updating of large portions of the index tree. Therefore the update due to insertion and deletion is accumulated, and actually performed when the amount of update is up to a threshold. The updating of index tree will affect the whole system and take some time, hence it is recommended to run as a batch job in the weekend when necessary. When an iconic index tree is constructed, it is very well balanced because the self-organization network self-organize the data according to their distribution.

3 Experimental Results of the Prototype

The prototype of the CAFIIRIS have been developed, except the facial aging module, which is still in the stage of researching. The algorithms and functions of the system have been tested against 100 Chinese male facial images of police officers and our colleagues and students. We are now try to collect more images.

The composition results are satisfactory to our users (police) for criminal identification. The module has been tested for various facial components. The composed faces look very natural. No artifacts are noticeable.

The feature extraction method can distinguish between different types of hairs and eyebrows. The classification results are consistent with visual classification with a degree of 90%. But for facial outline and eyes, the percentages is lower. It is very difficult to detect the size of eyes because of the expression and motion of eyes. A closed big eye may have a small size in its image. An important factor in facial outline is the shape of the cheek. But exact detection of cheek contour is not so easy. We are now developing a model-based method for better facial outline and eye categorization.

Iconic index has been implemented. It works very well for visual retrieval of facial images. It needs to be tested for large amount of images. Other functions such as similarity retrieval, free text retrieval, and fuzzy description retrieval are also being developed and tested. Some results are shown in Fig. 7 and 8.

4 Conclusion and Remarks

We have presented here a computer-aided facial image identification, retrieval and inference system CAFIIRIS. As explicitly indicated by the name of the system, the goal is to provide the users with an effective way to access the large amount of facial images stored in the system.

More than half of this paper is dedicated to the discussion of facial image indexing mechanism. We have generalized the indexing concept to use key of abstract data types. Spatial self-organization neural networks are used to build the tree index. Use of fuzzy membership functions greatly reduces the complexity of the index tree, and makes the image access much more friendly to users.

We hope that our work would lead to a general indexing method of multi-media objects. More work should be done to refine the present techniques and to extend the system to include other media such as voice. We also hope to extend the inference power of the system and extend the application area.

5 Acknowledgment

The authors would like to thank Inspector Lau and Inspector Chan of the Criminal Identification Department, Dr. Rexon Ngim of Singapore General Hospital, and Dr. Kenneth Lew of National University Hospital for sharing their knowledge with us.

REFERENCES (short list)

[1] Bruce, V. and M. Burton, *Processing Images of Faces*, Ablex Publishing, Norwood, New Jersey, (1992)

[2] Enlow, D. h., Case Western Reserve University, Facial Growth, 3rd Ed, W. B. Saunders company, (1990)

[3] Kohonen, T., The Self-Organizing Map, *Proc. IEEE*: 78, No.9, pp.1464–1480, Sept. (1990).

[4] Narasimhalu, A. D., CAFIIR: An Image-based CBR/IR Application, *1993 AAAI Symposium on the Integration of CBR & IR Technologies*, Stanford, March (1993)

[5] Wolberg, G., *Digital Image Warping*, IEEE Computer Society Press, Los Alamitos, (1990)

[6] Wu, J. K., LEP - Learning based on Experiences and Perspectives, *ICNN-90* Paris, (1990)

[7] Zadeh, L. A., Fuzzy sets as a basis for a theory of possibility, *J. FSS*, *1*: pp 3-28 (1978)

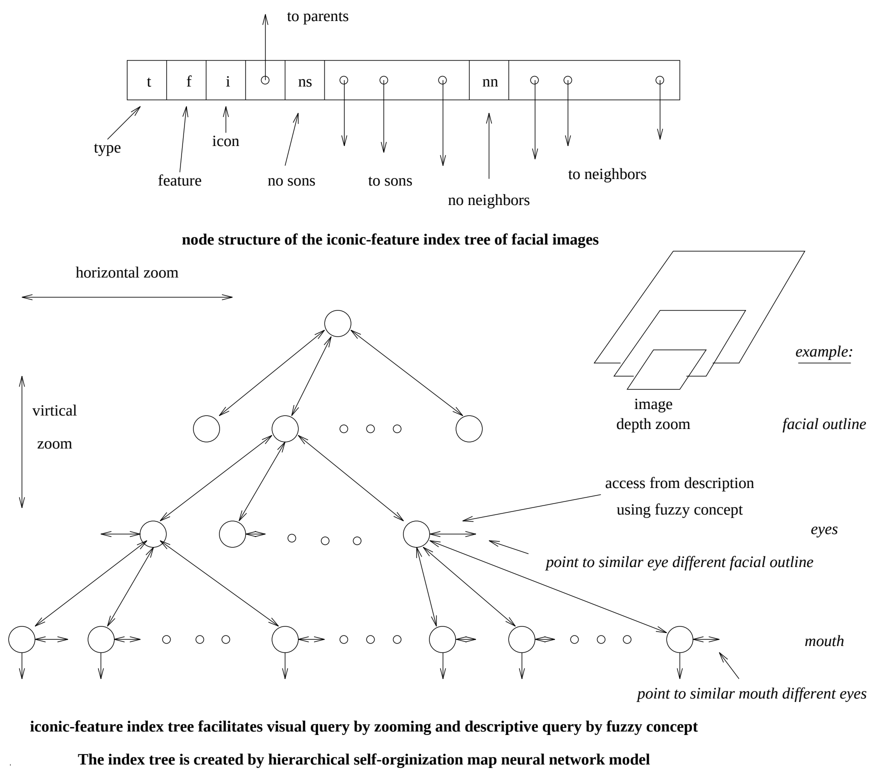

Figure 6: The facial image index tree and its node structure

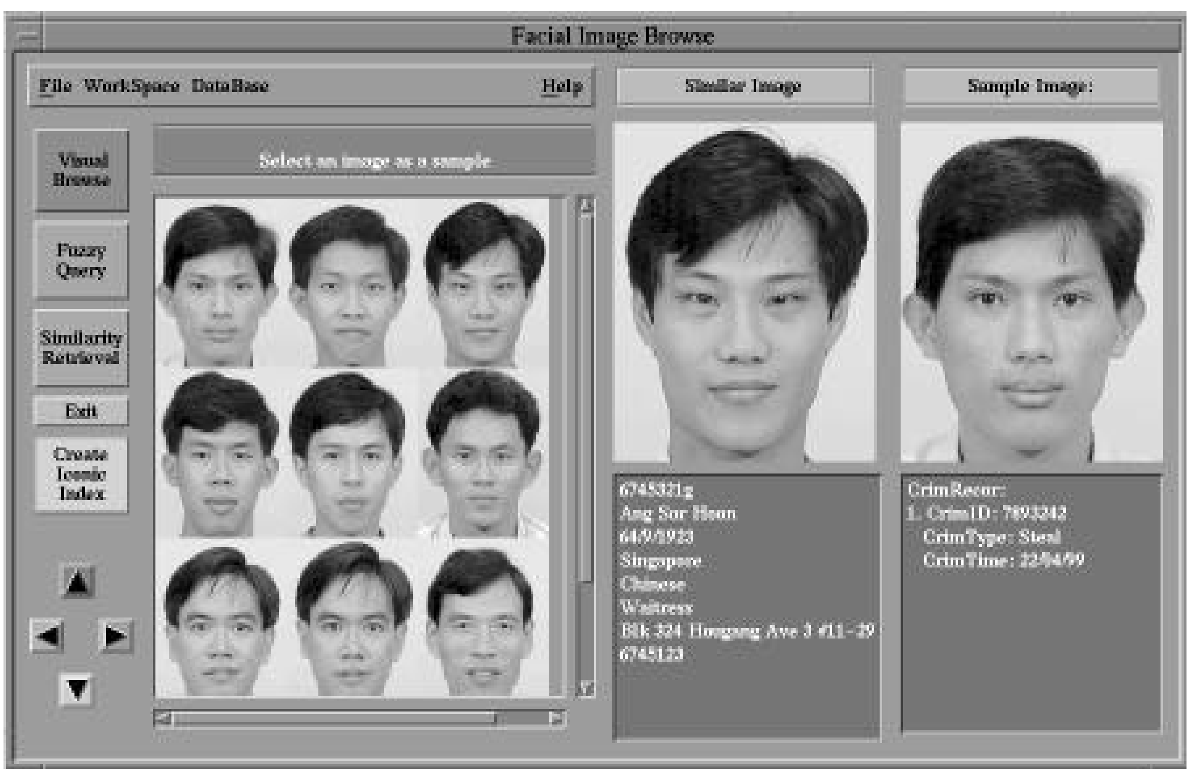

Figure 7: Similarity retrieval. The retrieved images are arranged according to similarity defined by Equation 4

Figure 8: Interface of Facial Image Database Management Module