

Entitled:

Wavelet-Based Hierarchical Organization of Large Image
Databases: ISAR and Face Recognition

Authors:

John S. Baras and Sheldon Wolk

Conference:

SPIE 12th International Symposium on
Aerospace, Defense Sensing,
Simulation and Control
Orlando, Florida
April 13-17, 1998

PROCEEDINGS OF SPIE REPRINT



SPIE—The International Society for Optical Engineering

Reprinted from

Wavelet Applications V

14–16 April 1998
Orlando, Florida



Volume 3391

Wavelet - Based Hierarchical Organization of Large Image Databases: ISAR and Face Recognition

John S. Baras^{a*} and Sheldon I. Wolk^b

^aDepartment of Electrical Engineering and Institute for System Research, 2249 A.V. Williams Building,
University of Maryland, College Park, MD 20742

^bNaval Research Laboratory, Code 5755, 4555 Overlook Avenue, SW,
Washington, DC 20375-5320

ABSTRACT

We present a method for constructing efficient hierarchical organization of image databases for fast recognition and classification. The method combines a wavelet preprocessor with a Tree-Structured-Vector-Quantization for clustering. We show results of application of the method to ISAR data from ships and to face recognition based on photograph databases. In the ISAR case we show how the method constructs a multi-resolution aspect graph for each target.

Keywords: Wavelets, Hierarchical Classification, Aspect Graph, Face Identification

1. INTRODUCTION

The classification of image data and the recognition of objects in images provide formidable difficulties due to the size of image databases, the lack of systematic procedures for data compaction for recognition and the multitude of situations where images appear in practice.

In this paper we describe a novel methodology for the hierarchical representation of large image databases based on wavelets. We report on theoretical results as well as on experimental results with synthetic ISAR images of ships and face pictures.

We use a combination of a wavelet preprocessor and a Tree-Structured Vector Quantization (TSVQ) post processor to construct this hierarchical representation of the images. This representation of images is progressive, in the sense that at various levels of the hierarchy different details appear. As a result we can design efficient classification schemes which are progressive. This is an essential feature of our algorithms.

In our method, a small amount of information in the form of a coarse approximation of the image, is used first to provide partial classification and progressively finer details are added until satisfactory performance is obtained. This approach results in a scheme where small amounts of computation are used initially and additional computations are performed as needed, resulting in extremely fast searches while preserving high fidelity in the search.

We have previously developed and applied such methods to one-dimensional signals (radar pulses, or acoustic returns) with success [1,3,4,5]. In the present paper we extend these methods to images. The resulting algorithms have proven to have some universal qualities. In this regard we have found analogs of such algorithms in animals and humans, notably in hearing and sound classification and in vision and identification of objects by humans. We report on further evidence of such similarities, especially on human vision, in this paper.

We describe a mathematical formulation of the problem as a combined compression and classification problem for images. This leads to certain types of algorithms for classification that need to be analyzed further. This analysis will be presented elsewhere.

* Also with AIMS, Inc., Research supported by AIMS, Inc., 6237 Executive Blvd., Rockville, MD 20852

2. REPRESENTATION AND CLASSIFICATION OF ISAR DATA

Inverse Synthetic Aperture Radar (ISAR) images can be thought of as sets of reflectivity estimates of a target plotted in a slant-range versus Doppler frequency coordinate system which forms the image projection plane. Within each range resolution interval, target rotations manifest themselves as Doppler shifts and reflectivity is apparent as signal strength. ISAR images contain in their structure substantial information about the target ship which can be used to better identify a complex target, such as ships, consisting of many scatterers. Due to the tremendous amount of data in ISAR image databases it is extremely important to discover efficient representations of large such databases for storage and search. We describe such a scheme here which provides a progressive search scheme and is amenable to learning.

The problem of automatic target recognition based on ISAR returns when a large number of ships is possible, presents formidable algorithmic and computational difficulties. A key step in the design and implementation of high performance ATR algorithms is the organization and construction of efficient and economic target models which will result in significant search speed-up and memory requirements reduction. In this paper we use as target models scale space aspect graphs constructed using the two basic techniques mentioned above (i.e. wavelets and hierarchical clustering) as described in our earlier work [1,3,4,5]. The fundamental extension here is image data.

2.1 Multi-resolution aspect graphs of ISAR data

A key difficulty in developing algorithms for classification from ISAR data (and more generally from images) is the dependence of the image on the so called "view point"; that is the relative orientation of the sensor registering the image and the object in the image scene. We solve this problem by constructing a tree representing the minimal number of "views" needed to represent the object. This tree is called the aspect graph [2] and provides an economical representation of the image data. We take a vector quantization (VQ) approach and use VQ in its clustering mode to cluster viewpoints into equivalent classes. These equivalent classes have the interpretation that within viewpoint angles from the equivalent class it is not possible to discriminate the projection and representation of the object of interest in the image data. We actually construct a multi-resolution aspect graph, in the sense that the tree has different resolution at each level. The nodes of the tree correspond to representative views from each equivalent class of viewpoints. The appearance (and disappearance) of specific features cause transitions from one node to another.

In Figure 1 below we show the intuitive explanation of a multi-resolution aspect graph, from our previous work on pulsed radar data [1,3,4,5]. We extend this construction here to ISAR data.

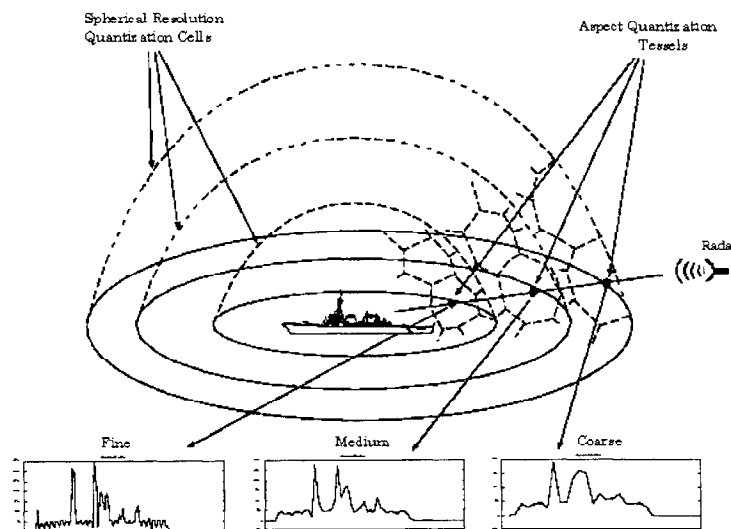


Figure 1: Illustrating multi-resolution aspect graph for radar ship data

Our algorithm combines a wavelet preprocessor (in two dimensions) followed by a Tree-Structure-Vector Quantization (TSVQ) exactly as developed in [1,3,4,5]. The only difference here is that the signals (or inputs to the algorithm) are images.

As is now standard for wavelet analysis we select a mother wavelet ψ and a scale function Φ . We did not find much performance sensitivity in our experiments todate regarding the selection of a particular mother wavelet.

Let \mathbf{f} represent the image data. The wavelet representation involves projecting \mathbf{f} in a sequence of subspaces of increasing resolution.

$$\dots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \dots$$

We represent by $\mathbf{S}^0 \mathbf{f}$ the image in fine resolution, which for this paper will be the given data. That is $\mathbf{S}^0 \mathbf{f} = \mathbf{f}$. The wavelet representation replaces the given data $\mathbf{S}^0 \mathbf{f}$, with the set $\{\mathbf{W}^m \mathbf{f}, m = 1, 2, \dots, J, \mathbf{S}^J \mathbf{f}\}$ where $\mathbf{W}^m \mathbf{f}$ represents the m^{th} level residual wavelet representation and $\mathbf{S}^J \mathbf{f}$ is the wavelet representation at the J^{th} level. Here J corresponds to the coarsest level of the representation.

To construct the multi-resolution aspect graph, we first apply VQ to the coarse level wavelet representation. This creates a number of cells (equivalence classes of viewpoints) needed to achieve the distortion (fidelity) required and put as a constraint. Then we find the cell with the maximum distortion and we split it in smaller cells using data from the next (finer) resolution level. The process is repeated until we reach the desired overall fidelity in the representation. This process and the associated TSVQ algorithm is described in our earlier work [1,3,4,5] and is illustrated in Figure 2. We call the resulting algorithm the Wavelet -- TSVQ (WTSVQ) algorithm.

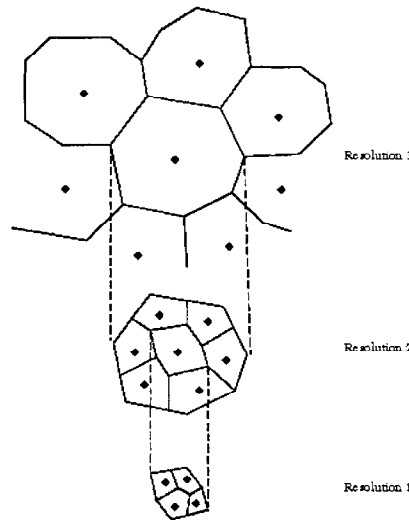


Figure 2: Illustrating the TSVQ clustering

We have shown in [1,3,4,5], that this representation of the data, although "greedy", is quite faithful in an information theoretic rate-distortion sense. Indeed comparison of the rate-distortion curve obtained by applying VQ to images at fine resolution and the rate-distortion curve obtained by the WTSVQ algorithm shows that they are very close.

Considerable research has been performed in recent years on algorithms that compute the aspect graph and its related representations [2]. However, todate these conventional methods have addressed only the ideal case of perfect resolution in object shape, in the viewpoint, and the projected image, leading to a set of important practical difficulties.

Almost exclusively, previous work on aspect graphs has focused on computer vision and object geometry. Our notion of the aspect graph as developed in [1,3,4,5] is an extension of these concepts to sensors other than cameras such as ISAR. We

present here further improvements and results from our work on the automatic construction of multi-resolution aspect graphs of ship ISAR returns. Our earlier results incorporate scale (resolution) in the construction of the aspect graph in a manner consistent with the sensor considered. Indeed we have developed an algorithmic construction of *scale space aspect graphs* (or *multi-resolution aspect graphs*). Scale space aspect graphs are equivalent to families of viewpoint space tessellations parameterized by scale.

The multi-resolution aspect graph that the WTSVQ algorithm constructs provides a most efficient model of the image and the objects in it. It is efficient but also highly accurate. It essentially stores an object model based on the set of minimal "views" required. In Figure 3 we show a typical aspect graph constructed from synthetic ISAR images from a ship. In Figure 4 we show the interpretation of the multi-resolution aspect graph of ISAR data from a ship based on viewpoint clusters; this is a similar interpretation to that shown in Figure 1 for pulse radar data.

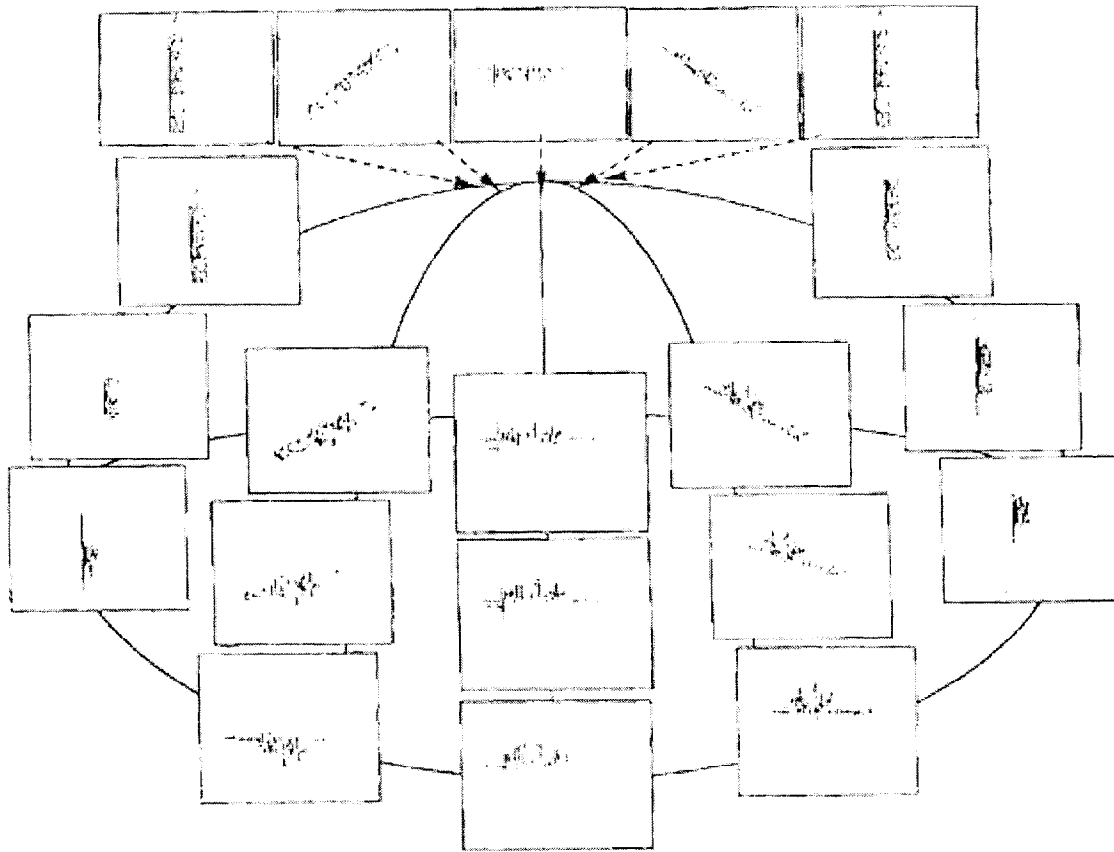


Figure 3: Multi-resolution aspect graph of ISAR data from a ship

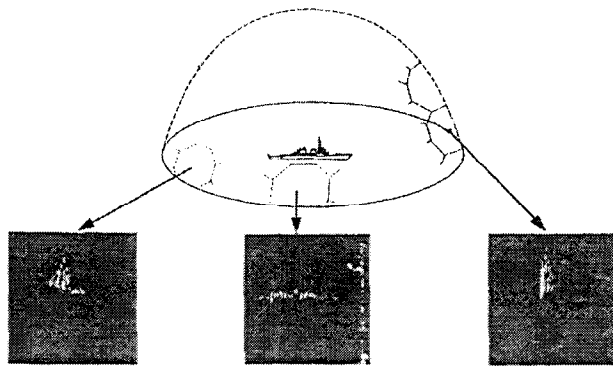


Figure 4: Viewpoint cells form the basis for the multi-resolution aspect graph of ISAR data

2.2 Target classification based on ISAR data

Our target representations lead naturally to classification schemes that are progressive; and this is an essential feature of our algorithms. That is, a small amount of information, in the form of a coarse approximation to the return, is used first to provide partial classification and progressively finer details are added until satisfactory performance is obtained. This results in a scheme where small amounts of computation are used initially and additional computations are performed as needed, resulting in extremely fast searches while preserving high fidelity in the search. Each target is represented by its multi-resolution aspect graph [2], which is a quantization (produced by clustering) of the space of ISAR returns and view points. Using an efficient Tree-Structured Vector Quantization (TSVQ) algorithm we cluster the returns from the various viewpoints into equivalence classes according to an appropriate discrimination measure. This approach automatically accounts for the discrimination capability of the sensor and in effect it performs a quantization of the sensory data which reduces the data input to the classification algorithm by orders of magnitude. In each equivalent class a “paradigm” is selected and the collection of these typical pulses arranged in a multi-scale tree constitute the target model that is guiding the on-line classification search. In this sense the aspect graph provides an indexing of the object (s) similar to geometric hashing. Furthermore, the progressive use of data provides for faster automatic target recognition (ATR) algorithms.

The algorithm constructs a hierarchical organization of the ISAR return data as a tree, which is conformant with the wavelet multi-resolution data representations. This representation can be constructed for a single target or for a collection of targets. In the former case this construction produces a “model” for the target as viewed by the ISAR sensor. In the latter this construction organizes an entire database of target radar returns. The resulting tree is our *aspect graph of the target(s)*. We demonstrate our results by experiments with highly accurate synthetic ISAR returns from the Naval Research Laboratory code 5750 ship radar return simulator. A multi-resolution (or scale space) aspect graph for ISAR ship data results naturally from our construction. In Figure 3 we show a typical aspect graph for one of the ships used in our experiments. In each node we show the Voronoi vector (or the typical return from the corresponding viewpoint cell) which is also the centroid of the ISAR returns from the same viewpoint cell.

To design a classification algorithm, an ATR algorithm in this case, we take a combined compression and classification viewpoint. This is inspired from the aspect graph representation of each target, which at each level uses a compressed representation of the ISAR data. In this sense VQ provides the critical link since the Voronoi vectors (or centroids) can be thought of as either codewords representing the ISAR data in that viewpoint cell or as the typical example of ISAR data from the cell which should be used for identification purposes.

Vector quantization (VQ) has been traditionally used, by the majority of the practitioners as a compression algorithm [6]. VQ is a common method of lossy compression that applies statistical techniques to optimize distortion/bit rate tradeoffs. However, as we have pointed out and used elsewhere, VQ can be also used as a classifier quite successfully. Therefore VQ can be used in a combined mode to perform classification efficiently utilizing compressed data. There are various methods to approach this promising idea. The one described here, which we propose to investigate further, combines some additional advantageous requirements. First, careful design of the overall scheme can emphasize local classifications involving only small regions of the signal. Second, these local classifications can be combined in a hierarchical fashion that reflects the signal mode, the sensor performance and sensitivities, and progressive increase in the confidence of the classification.

In such an approach, which fundamentally combines data compression and classification, one needs to develop efficient adjustment methods to weight the relative importance of the two aspects of the algorithm. Indeed, while efficient compression can reduce significantly the complexity of classification by bringing forward essential local characteristics of the signal, excessive compression may throw away valuable information and thus reduce the accuracy of classification. The design of this tradeoff is the most difficult and less unexplored, albeit most promising, part of our approach.

Our approach applies equally well to one-dimensional (radar pulses, acoustic signals, etc.) or multi-dimensional signals (FLIR, ISAR, SAR, LADAR, etc.). By appropriate indexing we can always consider a given signal (one-dimensional or multi-dimensional) as a vector. VQ operates on subblocks or subvectors of the signal or, in more sophisticated schemes, on subblocks or vectors of a feature vector computed from the given signal by some transformation. These subblocks or subvectors of the vector, on which VQ is to operate, correspond to local information from the signal or local features of the signal. For each subvector, the VQ encoder determines the nearest codeword (which is also a vector of the same dimension) and outputs the chosen codeword's index. When VQ operates in the data compression mode the sequence of indices so generated can be stored and then transmitted. The VQ decoder reverses this operation: it receives as inputs the indices and outputs the appropriate codewords by simple table lookup. One can easily realize that the computational complexity of the encoding and decoding part of the VQ is asymmetrical. The decoder is very simple. The attempt of combining compression and classification stems from the idea that the centroids of the VQ cells (which were called codewords in the compression scheme) are prime candidates for most typical representatives of the subvectors belonging in the same cell. It is likely that subvectors that are assigned by the VQ algorithm to the same cell belong to the same class. Therefore one can assign cells to classes and obtain classification. An efficient way of doing this is represented in the Learning Vector Quantization (LVQ) algorithm of Kohonen [7,8], which we have used and analyzed extensively at AIMS, Inc. [9,10]. This assignment of cells to classes can be best understood and analyzed as a method for partitioning the feature space into decision regions corresponding to each class. Indeed the boundaries of the cells approximate the Bayes decision surfaces for the classification problem at hand, if this assignment of cells to classes has been performed efficiently.

Our objective is to design algorithms that combine compression and classification and show that they result in high performance ATR algorithms. Towards this end we discuss next the various performance measures of such an algorithm based on VQ. Rate and Distortion characterize the performance of VQ from the point of view of compression. The performance of the classifier is measured by Bayes risk, which may include in general different costs for different types of errors. These performance measures are competing and there are various ways of approaching this *multi-objective* design problem. One is to combine the objectives (performance measures) by incorporating the Bayes risk in the distortion measure minimized by the design algorithm. Another is to treat the problem as a multi-objective optimization problem, where the design algorithm optimizes one performance measure while satisfying constraints on the other objectives. We are investigating both approaches and compare the results on ATR problems with real data.

Let us consider first the formulation of the problem and the proposed approach when we incorporate a Bayes risk into the average distortion measure minimized by the design algorithm. Such an approach introduces additional complexity which however occurs only in the design phase of the overall algorithm. The resulting algorithm has complexity equivalent to that of an ordinary VQ algorithm. In addition the combined classification and compression scheme requires no more bits to describe than the bits required for compression alone, which implies that there is no apparent memory overload.

Let F denote the K -dimensional vector space of features. We are given N feature vectors $\{f_1, f_2, \dots, f_N\}$. Each feature vector $f = (f_1, f_2, \dots, f_K)^T$ is mapped by a full search VQ onto a codeword (or centroid) $Q(f) = \theta_i$, from a set of centroids $\{\theta_1, \theta_2, \dots, \theta_M\}$, where M is the number of cells in the tessellation of F constructed by the VQ. This tessellation, or partition, induced by Q is denoted by $P^F = \{C_1, C_2, \dots, C_M\}$, where C_M denotes the generic cell in the tessellation. Let γ, δ denote the encoder, resp. the decoder of the VQ implemented in the overall algorithm. That is for each feature vector $f_i \in F$, if $Q(f_i) = \theta_m$, then $\gamma(f_i) = m$, and $\delta(m) = \theta_m$. Then $Q(f_i) = \delta(\gamma(f_i))$. Since $M \ll N$, the VQ compresses the data.

Next suppose that we have L classes (or hypotheses) $\{H_1, H_2, \dots, H_L\}$. As explained above, to perform classification we assign a class H_l to each cell C_m . This is the same as assigning a class label $l = 1, 2, \dots, L$, to each cell label $m = 1, 2, \dots, M$ in the partition of F induced by the VQ. This last assignment is the decision or classification rule d .

In such a scheme the signal vector x is first transformed by a preprocessor (which in our case is the wavelet transform (WT)) into a transformed vector w , so that $w=Tx$, where T in our case denotes the WT. Then a feature selection map F is applied to bring the transformed vector w into $f \in F$, so that $f=Fw$. The design of the combined compression and classification algorithm is precisely the design of the encoder, decoder and decision rules γ , δ , d . Equivalently this design is the construction of a tessellation $\{C_1, C_2, \dots, C_M\}$ of F with centroids (or codewords) $\{\theta_1, \theta_2, \dots, \theta_M\}$, and a decision rule d . As in LVQ this design can be accomplished by constructing the tessellation and the decision rule by an iterative process working with a training (or learning) set of features $L=\{f_1^L, f_2^L, \dots, f_N^L\}$ where N is the number of training vectors available overall. It is important to emphasize that the overall approach is non-parametric, in the sense that probability distributions for the signal, the transformed signal, and the feature vector are not needed. Instead the approach can be interpreted as using the training set to learn the empirical distributions of the various vectors and use them as if they were true, very much like the interpretation we have given to the LVQ algorithm [9,10].

Given a decoder-encoder pair γ, δ we associate the average distortion

$$D(\gamma, \delta) = E[\rho(f, \delta(\gamma(f)))]$$

Here ρ is the error (distortion) or distance function used in the VQ. Most of the work to date has used a quadratic function $\rho(f, \delta(\gamma(f))) = ||f - \delta(\gamma(f))||^2$, as the distortion measure, basically for its mathematical tractability. The selection and design of appropriate distance functions is an important element of our research and development effort.

We also associate the rate $R(\gamma, \delta)$ to a decoder-encoder pair γ, δ . The rate measures the complexity of the data representation.

Given a classification rule d , the classification performance of the overall scheme can be measured by the Bayes risk

$$J_B(\gamma, d) = \sum_{i=1}^L \sum_{j=1}^L P(d(\gamma(f)) = H_j | f \in H_i) P(H_i) C_{ij},$$

where C_{ij} is the relative cost assigned to the decision that $d(\gamma(f)) = H_j$, while the feature vector f comes from class H_i . The most commonly encountered case is that where $C_{ij} = 0$. An important observation is that the encoder δ does not affect the Bayes risk J_B .

We have therefore three performance metrics $D(\gamma, \delta)$, $R(\gamma, \delta)$, and $J_B(\gamma, d)$. We want to minimize all three but as is well known the resulting requirements are conflicting. We are investigating both a multi-objective approach and an approach where we combine the three criteria in one for some choice of the weights λ_R and λ_B . The two approaches are related by Pareto's theorem.

$$J_\lambda(\gamma, \delta, d) = D(\gamma, \delta) + \lambda_R R(\gamma, \delta) + \lambda_B J_B(\gamma, d),$$

In the optimization of the combined criterion we implement a three step iterative optimization:

Step 1 Choose $d^{(t+1)}$ to minimize $J_\lambda(\gamma^{(t)}, \delta^{(t)}, d^{(t+1)})$.

Step 2 Choose $\delta^{(t+1)}$ to minimize $J_\lambda(\gamma^{(t)}, \delta^{(t+1)}, d^{(t+1)})$.

Step 3 Choose $\gamma^{(t+1)}$ to minimize $J_\lambda(\gamma^{(t+1)}, \delta^{(t+1)}, d^{(t+1)})$.

The iterations continue until the desired sloping level for J_λ is met.

The interpretation of the steps is straightforward. Given a partition of F and a set of centroids, Step 1 minimizes the Bayes risk associated with the centroid labels. As can be seen, this minimization depends only on the partition represented by γ ; the codeword values given by δ do not affect the minimization.

Step 2 minimizes J_λ over the codeword values δ given the partitioning γ and the labeling d . Since the codeword values do not affect the Bayes risk, J_λ is minimized when the codewords are chosen as centroids based on the distortion measure given.

Finally Step 3 determines the partitioning γ that minimizes J_λ given the centroids δ and labels d .

We extended this procedure to a TSVQ. This is achieved by performing the above algorithm with the tree structure provided by the aspect graphs. In addition we have replaced the minimization step associated with the Bayes risk with Learning Vector Quantization (LVQ), appropriately extended to TSVQ. In LVQ Kohonen performs classification using a VQ encoder and codebook, where (assuming a quadratic distance) the encoder operates as an ordinary minimum mean squared error selector of a representative from the codebook, but the codebook is designed in a manner that reduces classification error implicitly rather than directly minimizing mean squared error. Kohonen's stated goal was to imitate a Bayes classifier with less complexity than other neural network approaches, but there is no explicit minimization of Bayes risk in the code design. However, in [9,10] we showed that indeed the way LVQ moves around the centroids during learning, asymptotically approximates the effect of optimizing Bayes risk. The argument is therefore that implementing LVQ in the above algorithm, essentially replaces the explicit optimization of the average Bayes risk. The result is a much more efficient algorithm.

As mentioned earlier we can represent the entire ISAR database on targets by a single aspect graph, which we call *the Global Aspect Graph*. Or we can represent each data set from a single target by an aspect graph. We call the later the *Parallel Aspect Graph*. When performing ATR using the first approach we find the leaf node of the Global Aspect Graph that is closest to the data and this gives us the decision. When performing ATR using the second approach we pass the real-time data in parallel from each aspect graph, find the aspect graph whose leaf provides the best proximity to the data and identify the target with the label of this aspect graph. The second algorithm is much faster due to its parallel implementation. Many experiments have shown that it performs much better than the first consistently, as measured by confusion matrix and ROC curve computations. It also performs close to the optimal provided by LVQ computations. Finally the second algorithm provides the best solution for inserting new targets in the database because it does not require a re-computation of the aspect graphs as the first algorithm does.

3. FACE RECOGNITION

Our basic classification algorithm utilizes a cascade of a wavelet preprocessor followed by a tree-structured clustering algorithm; a learning mode can be also added. We have applied similar ideas with great success to the problem of face recognition. An obvious application of our scheme would be, for instance, the real-time identification of a person thanks to his ID picture and a database where he would have previously been filed. Be it a criminal or not, the answer has to be quick and good. The scheme could work as well with fingerprints, for instance, or many other different images. The data we used for experiments is composed of 349 ID pictures of 95 different people. These photographs are 8-bits per pixel gray levels pictures and they are 128×128 pixels images. Every person is represented by several pictures (2 at least, 4 at most with an average of approximately 3.7 images per person.) Among those, 254 photographs form the database, and the remaining 95 constitute the set of unknowns i.e. the set of the photographs that will be processed through The Tree Structured Data Base in order to be identified. None of the ID pictures is at the same time in the Data Base and in the Test Set of Unknowns.

The global scheme of identification includes 5 steps, 4 of which occur during the design of the Tree-Structured Data-Base which explains why so much time is saved while performing the critical real-time step of recognition: Normalization of the D.B., Wavelet Transform, Vector Quantization, Tree design, Tree Search. First of all, the images are normalized thanks to an eye-detection based algorithm. The wavelet transform is then performed to achieve a multiscale representation of the photographs: different resolutions of every picture are computed so that they can be used alternately during the search, depending on the amount of details needed at a certain step of the search. Then the Tree-Structure is designed using a clustering algorithm in order to sort the images based on resemblance criteria. We describe a novel algorithm that combines global tree search with local full search. We have obtained the best results (in terms of high fidelity ID in very short times) with this algorithm. The key idea is to use a tree search to come to the vicinity of the best match and then use full search within the smaller set. We also implemented a multi-path search variation of this algorithm. Here several paths are followed in the tree until a clear winner emerges. This decreases significantly the error in the ID. We show that this is equivalent with allowing overlapping clusters at each resolution of our progressive classification scheme. For further details we refer to the report [11].

3.1 Normalization of the Image Database

One of the main problems that occur while processing images is the choice of a good measure to compute the distortion. Indeed, the usual Euclidean distance which is very easy to implement and therefore used very often (like in the computation of the PSNR, a traditional measure of quality between a source signal and its representation) does not fit very well the characteristics of the visual identification process of man. Two images can look very similar as well as very different and yet have the same PSNR i.e. the same image-to-image Euclidean distance comparison. In particular, the Euclidean distance is very sensitive to slight shifts or rotations. Such defects are very frequent regarding ID photographs. This is the reason why a proper normalization of every picture is necessary to make sure that the Euclidean distance is not meaningless. The Normalization of a picture is achieved by first enhancing it (i.e. increasing the contrast), then by detecting the eyes of the person, and finally by straightening the picture. This part of the scheme is the one that makes the application of the Multi-resolution Tree Search very specific to face recognition: many other applications are possible and what we did is just a particular one.

The 128×128 image is divided into a few sub-blocks whose means are computed. Then, the intensity of every pixel is recomputed regarding these local means and the distance from the pixel to the four nearest blocks. (Cf. Figure 5 below)

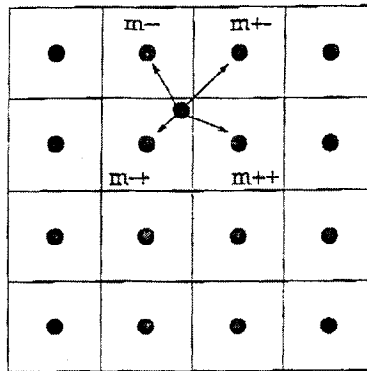


Figure 5: Sub-Blocks 16×16

With m_{+-} mapping the intensity of the pixel (x_+, y_-) and so on in comparison with (x, y) , the new value of the pixel (x, y) intensity is given by:

$$m(i) = a[bm_{++}(i) + (1-b)m_{-+}(i)] + [1-a][bm_{+-}(i) + (1-b)m_{-}(i)]$$

where $a = (y - y_-)/(y_+ - y_-)$ and $b = (x - x_-)/(x_+ - x_-)$.

but this re-mapping is only exact for the pixels at the center of the blocks whose means have been computed. This is why one should perform a bilinear interpolation to re-map correctly all the pixels.

Another point of view is to consider these new values for the pixels as local means. Indeed, these values have been computed according to the intensity of the neighborhood of (x, y) and to the proximity of this neighborhood. Consequently, we naturally applied this to the normalization formula: $I = I / \langle I \rangle$. The result looks better and the details appear more clearly even if the image seems less natural. The eye-recognition algorithm is therefore more accurate.



Figure 6: Image before and after contrast enhancement

We also tried to preprocess the images to see if it could improve the recognition rate. We had a slight improvement with a logarithmic preprocessing which is interesting (up to 5 points of percentage better). We believe that it helped remove the granular noise on the photographs.

Next we developed and implemented an eye-matching algorithm. Several eye-templates (coming randomly from the photographs of the Data Base) have been tested and chosen if their "visual quality" was good. The templates do not include the eyebrows and their size was 14×9 pixels. There exists a template for each eye because the right and left eyes are slightly different and the recognition works better if two templates are used instead of one. Given a template, one computes the cross-correlation:

$$C(x, y) = \frac{\langle IT \rangle - \langle I \rangle \langle T \rangle}{\sigma(I)\sigma(T)}$$

with I and T two blocks of pixels, between the template and the image (which is scanned in a "raster" way: top-left to bottom-right).

The two best matches (i.e. the two areas of highest correlations) are kept for each eye (which makes a total of 4 eyes) in order to make the detection-algorithm more robust. The two best-matches for the eyes should be distant enough to be considered as two different matches because most of the time, the highest values for the correlation are in the same area, around the local best-match. Moreover, the eye detection is done with an a priori knowledge of the position of the eyes (central area of the photograph which has been determined by looking at several ID pictures). This information is taken into account by weighting the correlation coefficients with a gaussian penalty so that the more one gets further away from the assumed position of the eyes, the bigger the penalty is. This leads to a good reduction of mismatches.

Among the four eyes that we have after the completion of the first step, we have to make a selection among the 4 eyes we have detected. The choice is made assuming the following basic properties of the eyes:

- The inter-eyes distance is standard that is to say it does not vary a lot with the photographs. Consequently, a pair of eyes is considered valid if the distance between the two eyes does not exceed a certain range of pixels (between 20 and 30 for our 128×128 pictures).
- The eye-line should have a horizontal position assuming that people do not have excessively bent heads on our photographs though such a defect could be fixed by rotating the picture. (A limit of inclination of 5° has been chosen).

Thanks to these simple criteria, we get rid of most of the mistakes and select the good pair of eyes. Nevertheless, another step is necessary because of the specific mismatches concerning the ears which satisfy perfectly the discrimination criterions pre-cited if the person is sideways and only one of his eyes is detected along with his ear.

Once all of this information about the eyes has been computed, the image is straightened: the middle of the two eyes goes to the center of the photograph frame. The straightening takes into account the shifts and rotations that affect the head. The problem of zooming/unzooming is not considered here as it has already been taken care of (it is a criterion of identification of the eyes) and the images are standard enough (speaking of the zoom effect) to overlook this problem. The background areas that appear after the straightening are filled with the average gray level of the picture. An important thing is that what we straighten are the original images and not the enhanced ones in order to allow any other kind of preprocessing.

Using one pair of eye-templates, one could straighten most of the pictures with no errors on condition that the adjustable thresholds be selective enough. Most of the time, the rejected images were those of side-face people, dark photographs, people with 'strange' eyes (i.e. very different from the template used, like persons with glasses). About a third of the images could be straightened that way, given that about a half of the images had these "defects". The use of different eye templates is therefore necessary for a good recognition of most of the different eyes one might come across (more or less closed, swollen etc. ...) People who wear glasses are especially difficult to straighten because of the various shapes of the glasses and because of the frequent reflection effect due to the flash of the camera. Finally, 308 images out of 349 were straightened, the other being 'pathologic' (sideways, grimacing, too dark ...) or people wearing glasses.

3.2 Wavelet Transfer and Compression of the Pictures

In our scheme we are interested in a gradual recognition of ID pictures: starting from a coarse resolution of the picture, the identification is performed with an increasing precision (i.e., with an increasing number of details taken into account).

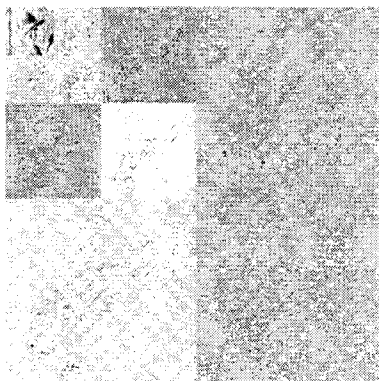


Figure 7: Example of wavelet representation of a picture

We used separable wavelets for application to the image database. We have tested several FIR filters (representing wavelets) for application to pictures. We basically search for good performance based on PSNR as well as for the visual effect of the reconstructed image. Bi-orthogonal wavelets were found to perform better than orthogonal ones, basically because of their better smoothness. This allowed for good rendition of the reconstructed image. The two spline filters selected are shown in Figure 8 below. Nevertheless the improvement was not drastic regarding the selection of the filter (i.e., the wavelet). Thus selection of the wavelet is not an essential parameter to focus on.

| n | 0 | 1 | 2 | 3 |
|-------|-------|--------|-------|--------|
| h_r | 0.6 | 0.25 | -0.05 | 0 |
| h_s | 17/28 | 73/280 | -3/56 | -3/280 |

| n | 0 | 1 | 2 | 3 | 4 |
|-------|----------|----------|-----------|-----------|----------|
| h_r | 0.502949 | 0.266844 | -0.078223 | -0.016864 | 0.026749 |
| h_s | 0.557543 | 0.295636 | -0.028772 | -0.045636 | 0 |

Figure 8: The two spline filters (wavelets) used

We used vector quantization to compress the images following the wavelet transform. In our application VQ was applied to quantize the wavelet coefficients. Since the wavelet transform performs a subband filtering of the image we allocate different bits to the various bands (resolutions). We used well known bit allocation schemes for VQ coding of wavelet coefficients of images. The following bit allocation scheme was selected.

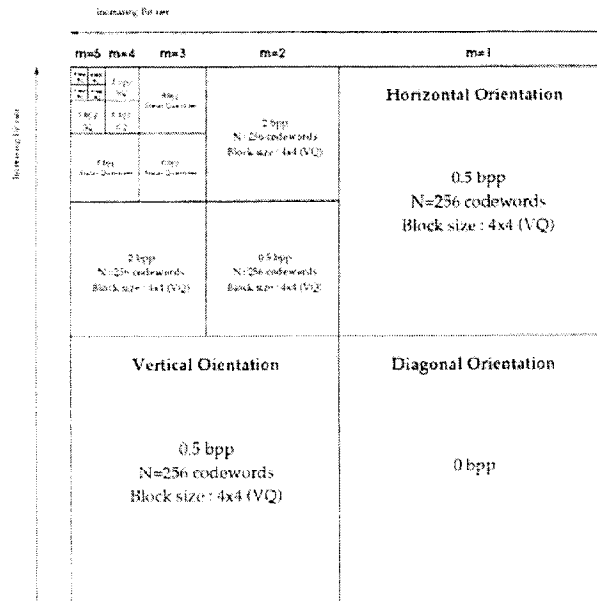


Figure 9: Optimal bit-allocation for a 1bpp bit-rate

3.3 Tree search and tree design

By organizing the database of pictures into a tree we can obtain fast search, which is logarithmic in the number of pictures instead of linear. The TSVQ part of our algorithm achieves this task. We used Euclidean distance as before. For splitting the nodes we introduced a small modification in our algorithm. Instead of splitting the node that has the largest distortion we compute the distortion of the two potential children weighted by their population minus the distortion of the parent cell. In other words we are interested in the decrease of distortion that would result from the splitting of this node. This produced the best results and is in accordance with greedy algorithms of this type. We selected as usual the centroid of each cell as the representative for ID purposes.

We performed tests comparing various search methods. The first was Full Search of the Database. The second was the Tree-search following our WTSVQ algorithm described. We obtained best results by a combination method that used initially a Tree-search followed by a Full-search in the vicinity of the target image. This algorithm proved to be highly efficient and accurate. The key idea is to use Tree-search to zero-in on a set (small) containing the best match. A Full search is then performed within this smallest set to find accurately the best match.

Observations show that the vast majority of the mistakes made while proceeding with the Tree-Search are due to the fact that during the binary choice of the best way down the tree, errors occur when "Unknown" is situated on the border of the two tree-nodes. i.e., when they are located right on the geometrical middle of the two centroids of the children cells (or in a small neighborhood around the middle of these two vectors). Indeed, due to the multi-resolution structure of the scheme, there exists a relative ambiguity on the pictures that decreases with the wavelet synthesis steps. Hence, the borders of the two sets are not reliable areas. A solution to this problem is to consider that, when the ambiguity is too large more paths should be followed down the tree, until the resolution (which improves with the layers) is such that a reliable choice can be made between the various alternatives. If a choice cannot be made (because the vectors are too close or the unbalanced tree forbids

one more go-down-the-tree step) the matches are kept and a decision is made at the end with a multi-resolution Full Search ('Unknown' is compared to the small set of matches at the lowest resolution and the matches are gradually eliminated if they exceed a certain distance from 'Unknown'. Then the resolution is increased until a unique match is reached.) We called this algorithm Multipath Search. We obtained further improvements on performance with this algorithm.

4. EVIDENCE FROM BIOLOGY AND NEUROSCIENCE

Our work emphasizes storage of objects as a collection of a minimal set of views for storage and classification efficiency. How do animals, humans perform these tasks? Recent experiments by Poggio and Logothetis [12,13] with animals (monkeys) and actual brain measurements support our thesis. Monkeys appear to store two-dimensional views of 3-D objects, for various viewpoints. They store more views for new or unknown objects; less views later as they learn to recognize the object. The reduction of views is accompanied by some higher order interpolating function

5. REFERENCES

1. J.S. Baras and S.I. Wolk, "Hierarchical Wavelet Representations of Ship Radar returns", submitted for publication to *IEEE Trans. On Signal Process.*, 1993; also *NRL Technical Report NRL/RF/5755-93-9593*.
2. D.W. Eggert, K.V. Bowyer, C.R. Dyer, H.I. Christensen and D.B. Goldgof, "The Scale Aspect Graph", *IEEE Trans. on Pattern Anal. and Mach. Intel.*, Vol. 15, No. 11, pp. 1114-1130, Nov. 1993.
3. J.S. Baras and S.L. Wolk, "Efficient Organization of Large Ship Radar Databases Using Wavelets and Structured Vector Quantization", *Proc. of the 27th Annual Asilomar Conference on Signals, Systems, and Computers*, Vol 1, pp. 491-498, November 1-3, 1993, Pacific Grove, California.
4. J.S. Baras and S.I. Wolk, "Model based automatic target recognition from high range resolution radar returns", *Proc. of SPIE International Symposium on Intelligent Information Systems*, Orlando, FL., April 4-8, 1994.
5. J.S. Baras and S.I. Wolk, "Wavelet based progressive classification of high range resolution radar returns", *Proc. of SPIE International Symposium on Intelligent Information Systems*, Orlando, FL., April 4-8, 1994.
6. A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Press, 1991.
7. T. Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, third ed., 1989.
8. T. Kohonen, "The Self-Organizing Map," *Proceedings of the IEEE*, pp. 1464-1480, Vol. 78, no. 9, Sept. 1990.
9. J.S. Baras and A. LaVigna, "Convergence of a Neural Network Classifier," *Proc of 29th IEEE Con. on Dec. and Control*, December 1990, pp. 1735-1740.
10. J.S. Baras, D.C. MacEnany and A. LaVigna, "Automatic Target Recognition Algorithms," Quarterly Progress Report on Contract DAAB07-90-C-F425, AIMS-TR-91-1, AIMS, INC., April 1991.
11. J.S. Baras and J.C. Duthou, "Tree Structured Face Recognition Algorithm Using a Multiresolution Representation of Photographs", Technical Report, Institute for Systems Research, in preparation.
12. N.K. Logothetis, J. Pauls, H.H. Bulthoff and t. Poggio, *Current Biology*, 1994, Vol. 4, No. 5, pp. 401-414.
13. N.K. Logothetis, J. Pauls, and T. Poggio, "Shape representation in the inferior temporal cortex of monkeys", *Current Biology*, 1995, Vol. 5, No. 5, pp. 552-563.