VQ-FACES – UNSUPERVISED FACE RECOGNITION FROM IMAGE SEQUENCES

B. Raytchev and H. Murase

NTT Communication Science Laboratories, 3-1, Morinosato Wakamiya, Atsugi-shi,
Kanagawa 243-0198, Japan
{bisser, murase}@eye.brl.ntt.co.jp

ABSTRACT

In this paper we propose a new method for unsupervised face recognition – VQ-faces, which operates on a sequential stream of face images and is able to handle both frontal and side-view faces at the same time. The method consists of two parts: in the first part, the VQ-faces are calculated as prototype vectors of local areas in image-space, coding for different face-views (i.e. a "view codebook" is generated), while in the second part the fact that each face-sequence corresponds to a single person (temporal constraint) is used to cluster the different sequences into face categories, using a combinatorial optimization process which maximizes an objective function, reducing the global representation error (distortion) by the VQ-faces as sequences are grouped together. The method was tested on real-world data gathered over a period of several months and including both frontal and side-view faces from 17 different subjects, achieving correct self-organization rate of 85.4%. The applicability of the proposed method is not limited to face recognition – it can be easily applied to other problems involving multi-view object recognition.

1. INTRODUCTION

Being an area of both theoretical and practical interest, face recognition has recently attracted a lot of attention, leading to many significant achievements [1]. In this paper, however, we would like to address some problems, which in our opinion haven't received enough attention from the face recognition community. In its major part, face recognition research seems to concentrate on tasks where *a few frontal* or near-frontal view face images (typically high resolution images taken under constrained conditions) from *many* people are learnt in a *supervised* manner. This seems to be in contrast to the way people learn faces, where unsupervised learning from temporally-constrained continuous sensory streams, containing the whole spectrum of variations in illumination, viewing angles and object sizes which everyday life provides, seems to be employed. For this reason, we are interested to investigate the possibilities (and limitations) of a similar approach to face recognition, where *unsupervised* learning from *sequences* containing *multiple-view* face images is undertaken.

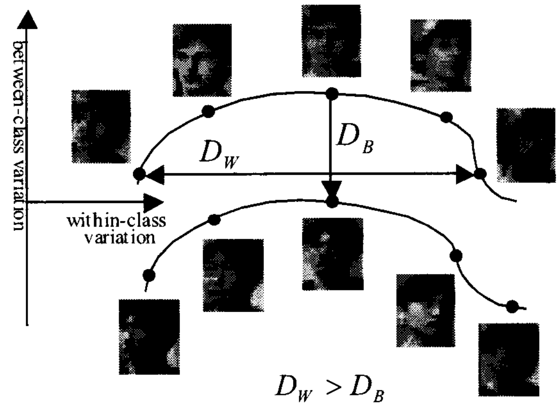

Fig.1 Intraclass distance is greater than interclass distance

In order to be able to perform face recognition in an unsupervised manner, i.e. if no category-specific information is provided in advance, a method for clustering of the unlabelled input stream (face-sequences here) into identity categories will be necessary. However, as people move in unconstrained dynamic scenes, exposing different views of their faces to the camera (and things are further complicated by factors like variation in scale and illumination, changes in facial expressions, and so on), the resulting face sequences would depict complex curves in input image space. Whatever measure for face similarity distance is defined between two faces, generally the within-class distance will be greater than the between-class distance, as illustrated in Fig. 1. Most of the presently known clustering methods [2] would not be able to detect clusters of such complex forms, and therefore wouldn't be suited for our problem. If clustering were attempted with them, instead of grouping together different views of the same person's face (along the within-class, or view-variation axis in Fig.1), most probably they would detect clusters of similar views of different people (along the inter-class variation axis).

0-7803-7622-6/02/$17.00 ©2002 IEEE

In this paper we propose a new method, *VQ-faces*, for unsupervised face recognition from image sequences, which tries to overcome the above problem. The method consists of two parts: in the first part, the so-called VQ-faces are calculated as prototype vectors of local areas in image-space, coding for different face-views (i.e. a "view codebook" is generated), while in the second part the fact that each sequence corresponds to a single person (a "temporal-constraint") is used to group the different sequences into face categories, using a combinatorial optimization process which maximizes an objective function, reducing the global representation error (distortion) by the VQ-faces as sequences are grouped together. Although here we will describe the method in relation to face recognition, the applicability of the proposed method is not limited to faces – it can be easily applied to other problems involving multi-view object recognition from image sequences.

2. FACE RECOGNITION WITH VQ-FACES

Before explaining the VQ-faces method in a greater detail, we will briefly describe the main points which will determine the clustering strategy. Rather than making any assumptions on a global level about the form of the multi-view face clusters (which is unlikely to be of some well-defined distribution), we make the following assumption about the local distribution of face samples in image space. If face-image space is divided into local areas represented by (a) an area prototype calculated as the centroid of the samples in the area; and (b) all samples within a radius R from that prototype; then if a suitable value for R is chosen, the predominant part of the local areas will contain face-samples coming from different sequences, but corresponding to a similar face-view of the same category (same person), plus an insignificant amount of "noise samples", consisting of faces coming from different views of the same category, or similar (to the prototype) views of different category. Based on the above assumption, the VQ-faces approach can be implemented by the following two algorithms:

(1) divide image space into local areas of radius R, i.e. calculate the area prototypes from the input data, and label each area with the label attached to the sequence which has made the greatest contribution in the calculation of the prototype;

(2) use as a temporal constraint the fact that each sequence belongs to a single category only (i.e., different people's faces don't appear in the same sequence), in order to group the sequences by a combinatorial optimization process which maximizes an objective function, reducing the global representation error (distortion) by the prototype in each area after a new sequence is added to the set of labels (represented sequences) for the corresponding area. The two algorithms above will be described in more detail in sections 2.2 and 2.3 below.

2.1. Preprocessing – face sequence formation

Since the concrete implementation of this part of the system is not essential for the operation of the clustering algorithm, a detailed description will be omitted. All that is required from the preprocessing is to obtain image sequences of the moving objects of interest and to guarantee that each separate image sequence corresponds to one and the same object only. We assume that input is provided from a video camera fixed in a constant position and continuously monitoring the scene in front of it. The subjects enter the scene and while walking in front of the camera look around, thus exposing different views of their faces. To extract face-only image sequences, a multi-resolution image pyramids are formed from the binary silhouettes of the moving subjects, and the face area is extracted after analyzing the x and y-histograms of the binary silhouettes at different resolutions. The extracted and normalized face-only image sequences (see Fig. 2) are input to the next stage of the system for clustering.

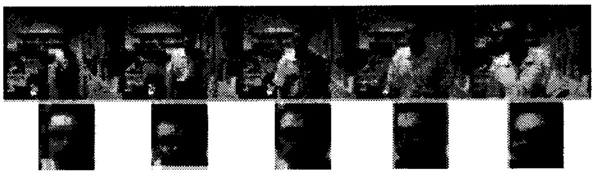

Fig. 2. Example of an original face image sequence (temporally subsampled) together with the corresponding normalized face-only sequence extracted from it.

2.2. Tesselation of input image space

As different unlabelled face image sequences become available from the preprocessor (only time-stamps being attached to each sequence at this stage, with no category-specific information provided), the following algorithm is used to divide input image space into polyhedral regions, each region represented by prototype face vectors, which we call *VQ-faces* because of the similarity of the resulting representation to a Voronoi tessellation as in the vector quantization (VQ) algorithm [3].

VQ-faces : face-space tessellation algorithm

```
INITIALIZE
  $R$; $N = 1$; $\zeta^{(1)}$ = first face in first sequence; $\mu^{(1)} = 0$ .
WHILE $\exists$ unprocessed image sequences
  FOR each face-image vector $x$ in current sequence
    IF $\min_m\{dist(\zeta^{(m)}, x)\} > R$    $(m = 1,\ldots, N)$
      THEN
        $N := N + 1$; $\zeta^{(N)} := x$;  $\mu^{(N)} := 1$;
```

```
ELSE
    $m := \arg\min_n \{dist(\zeta^{(n)}, x)\}$;  $\mu^{(m)} := \mu^{(m)} + 1$;
    $\zeta^{(m)} := \zeta^{(m)} + \frac{x - \zeta^{(m)}}{\mu^{(m)}}$;
    WHILE $\exists\, y_{\max} = \arg\max_{y \in m} \{dist(\zeta^{(m)}, y) > R\}$
        $remove(y_{\max}, m)$;
        $\mu^{(m)} := \mu^{(m)} - 1$;
        $\zeta^{(m)} := \zeta^{(m)} + \frac{\zeta^{(m)} - y}{\mu^{(m)}}$.
```

In the algorithm above, x and y are face vectors (face images represented in a vector form), N is the number of currently allocated prototype vectors (VQ-faces), $\zeta^{(m)}$ and $\mu^{(m)}$ stand for the VQ-face vector and the number of face samples in area m respectively, and $dist(x, y)$ calculates the distance (inversely proportional to the similarity) between vector-faces x and y. Procedure $remove(y_{\max}, m)$ removes face $y_{\max}$ from area m and puts it into an area k, distance to whose prototype $\zeta^{(k)}$ is minimal compared to all prototypes, distance to which is less than R. If such area doesn't exist, a new area $N := N+1$ is formed with prototype $\zeta^{(N)} := y_{\max}$.

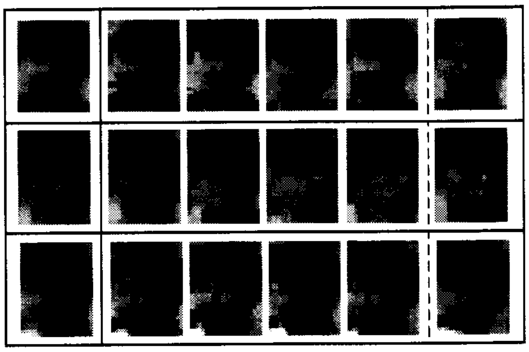

Fig.3 Three examples of VQ-faces coding for different views are shown in the leftmost column. To the right of them are shown several of the samples (each from different sequence) from which they were computed, while in the rightmost column are shown "noise" samples which usually occupy an insignificant fraction of all samples in the same area.

As a result of the algorithm above, input face-image space is divided into N areas, each area being represented by its prototype (VQ-face). Several examples of such VQ-faces, together with some of the sample-faces in their area used to calculate them are shown in Fig. 3. The tessellation algorithm will build a "view-book" (similar to the codebook in the VQ algorithm) from the input sequences, with different prototypes coding face views of different angles, and if the assumptions made in the previous section are valid, each area will contain predominantly samples of similar view angle from different sequences but from the same face category, plus (hopefully) an insignificant level of face samples from different views and different categories, which can be considered as noise.

2.3. Unsupervised recognition of face sequences by combinatorial optimization

Now, the temporal-constraint is used to group the unlabelled face sequences into categories. From the tessellation obtained with the algorithm in the previous section, all areas (together with their prototypes) which contain samples from a single image sequence only are eliminated, thus the number of areas is reduced from N to M (where $M<N$). The following two matrices, **A**=$\{a_{lm}\}$ and **V**=$\{v_{lm}\}$, information in which will be necessary in order to obtain the final grouping results, are calculated from the remaining M areas:

$$a_{lm} = \sum_i \exp(-\alpha \left\| \zeta^{(m)} - x_{li}^{(m)} \right\|^2) \qquad (1)$$

$$v_{lm} = \begin{cases} 1: & if\ l = \arg\max_j (a_{jm}) \\ 0: & otherwise \end{cases} \qquad (2)$$

where, $x_{li}^{(m)}$ are all vector-faces of sequence l in area m. Initially, each available face-image sequence l (l: 1...S) defines a separate set, a singleton $L(l)=\{l\}$. At each successive step t of the combinatorial optimization algorithm, the sets $L(i^*)$ and $L(j^*)$ corresponding to those two sequences (i^*, j^*), whose merge would optimize the representation error, as defined in (3) below, are merged into a single set $L(i^*) \cup L(j^*)$:

$$\begin{aligned}(i^*, j^*) &= \arg\max_{(i,j)} \Delta E^t_{(i,j)} \qquad (3)\\ &= \arg\max_{(i,j)} \{(\varphi_{ij} - C) T_\delta (\sum_{m=1}^{M} [\theta(\sum_{r \in L(j)} v_{rm}) \sum_{p \in L(i)} (1 - v_{pm}) a_{pm} \\ &\quad + \theta(\sum_{p \in L(i)} v_{pm}) \sum_{r \in L(j)} (1 - v_{rm}) a_{rm} \\ &\quad + \prod_{k \in L(i) \cup L(j)} (1 - v_{km}) T_0 (\sum_{k \in L(i) \cup L(j)} a_{km} - \sum_{l \notin L(i) \cup L(j)} v_{lm} a_{lm})])\};\end{aligned}$$

$$\begin{aligned}Z_{rp} = \theta\{\sum_{m=1}^{M} [& v_{rm}(1 - v_{pm}) a_{pm} + v_{pm}(1 - v_{rm}) a_{rm} \qquad (4)\\ & + (1 - v_{rm})(1 - v_{pm}) \theta(\sum_{k \in L(i) \cup L(j)} a_{km} - \sum_{l \notin L(i) \cup L(j)} v_{lm} a_{lm})]\};\end{aligned}$$

$$\varphi_{ij} = \frac{\sum_{r \in L(i)} \sum_{p \in L(j)} z_{rp}}{\eta(L(i))\ \eta(L(j))} \quad (5)$$

$$\theta(x) = \begin{cases} 1: & x > 0 \\ 0: & otherwise \end{cases} ; \quad T_{\delta}(x) = \begin{cases} x: & x > \delta \\ 0: & otherwise \end{cases} \quad (6)$$

where, $\eta(L(i))$ is the cardinality of set $L(i)$, C is a constant (between 0 and 1), and φ_{ij} in (5) is a *merge factor* which can take values in the range [0,...,1], providing a measure of how *unlikely* it is that the merge of the sets corresponding to the chosen pair (i^*, j^*) would produce a "chain-effect" [4], thus erroneously leading to the merge of sets belonging to different categories.

At each optimization step, after the sequence pair (i^*, j^*) in (3) is selected, matrix V is updated in order to reflect the merge (i.e. add a new sequence to be represented by the prototype in the corresponding areas) in the following manner: all v_{qm} (q : p, r, or k) which have produced non-zero values $(1 - v_{qm})\,a_{qm}$ respectively in the first, second and third terms in (3) for the maximal value of $\Delta E'_{(i^*,j^*)}$ are set to '1', while all v_{lm} which have contributed non-zero values $v_{lm}\ a_{lm}$ in the third term are set to '0'. Thus, the sets $L(i)$, obtained after the optimization algorithm converges, will contain the final clustering result.

3. EXPERIMENTS

In order to evaluate the performance of the proposed method we used a dataset of 377 face image sequences obtained during a period of several months from 17 different subjects. A typical example of the experimental setting can be seen in Fig.1, and the experiment itself was described in section 2.1. Illumination conditions were very demanding and varied significantly with the time of the day. Samples with and without glasses were included and hairstyles changed with time. Resolution of the original images was 320x240 pixels, and 18x22 pixels for the normalized face-only images. A recognition rate of 85.4% was achieved on the dataset of 377 sequences, using the following formula for calculating the self-organization (recognition) rate ρ :

$$\rho = (1.0 - \frac{E_{AB} + E_{O}}{S}) \times 100\% . \quad (7)$$

In (7), S is the total number of sequences to be grouped, E_{AB} is the number of sequences which are mistakenly grouped into the cluster for certain category A, although in reality they come from category B, and E_O is the number of samples gathered in clusters in which no single category occupies more than 50% of the nodes inside them.

Because of the large volume of data, we were unable to manually inspect all the face sequences output from the preprocessing module, but from the few inspected ones it was obvious that the dataset on which the system had to perform contained many instances of noisy data, in the form of erroneous face croppings and misalignments, large variations in illumination with face shadows, and so on, as would be expected in a real-world situation.

4. CONCLUSION AND FURTHER WORK

In this paper we have proposed a novel method for unsupervised face recognition from video sequences, which offers the following major advantages: (a) the learning process implemented by the method does not rely on category-specific information provided by human teachers in advance, but rather lets the system self-organize the sensory input. This allows all stages of the resulting system to be completely automated, avoiding the need for manual segmentation and labeling of the input stream, which might be impractical and sometimes impossible, e.g. in on-line video surveillance systems or with large databases; (b) both frontal and side view faces can be handled by the method, thus the problem of finding view-invariant features, or the necessity to build view-invariant models in a supervised manner (which also can be labor-consuming or impractical) can be avoided. A preliminary test of the method on a dataset containing both frontal and side-view face sequences obtained under demanding real-world conditions achieved a self-organization rate of 85.4%, which can be considered encouraging, having in mind the difficulty of the task and the bottleneck of an unreliable preprocessor and sub-optimal face-similarity measures used.

Acknowledgment

The authors are grateful to Dr. K. Ishii and Dr. N. Sugamura of NTT CS Laboratories for their help and encouragement.

REFERENCES

[1] H. Wechsler, P. J. Philips, V. Bruce, F. F. Soulie, and T. S. Huang, (eds.), *Face Recognition: From Theory to Applications*, Springer-Verlag, 1998.

[2] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, Englewood Cliffs, NJ, 1988.

[3] N.M. Nasrabadi and R.A. King, "Image Coding Using Vector Quantization: A Review," IEEE Transactions on Communications 36, pp. 957-971, 1988.

[4] B. S. Everitt, *Cluster Analysis*, Wiley, 1993.