VIDEO INDEXING BY FACE OCCURRENCE-BASED SIGNATURES

Costas Cotsaces, Nikos Nikolaidis and Ioannis Pitas

Department of Informatics
Aristotle University of Thessaloniki
Thessaloniki 54124, Greece
(email: pitas@zeus.csd.auth.gr)

ABSTRACT

The extraction of a digital signature from a video segment in order to uniquely identify it, is often a necessary prerequisite for video indexing, copyright protection and other tasks. Semantic video signatures are those that are based on high-level content information rather than on low-level features of the video stream, their major advantage being that they are invariant to nearly all types of distortion. Since a major semantic feature of a video is the appearance of specific people in specific frames, we have developed a method that uses the pre-extracted output of face detection and recognition to perform fast semantic indexing and retrieval of video segments. We give the results of the experimental evaluation of our method on an artificial database created using a probabilistic model of the creation of video.

1. INTRODUCTION

A video *signature* is a reduced dimensionality representation of a video segment that uniquely identifies it. It is almost indispensable for video indexing [1]. In this paper, we will construct such a signature by using semantic information, namely information about the appearance of faces of distinct individuals. We will not concern ourselves with the extraction of face-related information, since ample work has been performed on the subject. Instead we will try to solve the problems of consistency and robustness with regards to face-based indexing, to represent face information with minimal redundancy, and also to find a fast (logarithmic-time) search method.

All works on face-related information for video indexing until now [2, 3, 4, 5, 6] have focused on the extraction of the face-related information and not on its organization and efficient indexing. In effect, they are works on face recognition with a view to application on indexing. In our work we do not propose a face detection and recognition method. Instead, we investigate the effect of different parameters of the face detection and recognition process on the indexing performance of our method. The data used for this purpose is constructed by a probabilistic model describing the appearance of faces in videos and the function of face detectors and recognizers.

The paper is organized as follows: in section 2 we give an overview of our algorithm, in section 3 we describe the data we use for experimental verification, section 4 provides the experimental results, and conclusions are presented in the last section.

This work has been supported by the European Union project MUSCLE: Multimedia Understanding through Semantics, Computation and Learning

2. ALGORITHM DESCRIPTION

2.1. Format of Signature

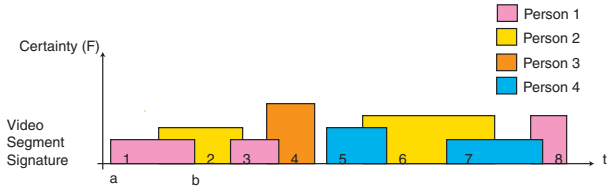

Fig. 1. Example of the characterization of a video segment by quartets. Colors correspond to distinct individuals. Signature elements are represented by numbered rectangles.

Let $\mathbf{V} = \{f_1\ f_2\ \ldots\ f_N\}$ be a video consisting of a number of consecutive frames $f_n, n = 1 \ldots N$. that we wish to characterize through an appropriately constructed signature. Let $\mathbf{S} = \{s_1\ s_2\ \ldots\ s_M\}$ be the set of all the individuals $s_m, m = 1 \ldots M$ that have been imaged in the video. Optionally, with no loss of generality, we can assume $\mathbf{S}$ to contain only the individuals of interest. This can mean, for example, excluding the extras in a motion picture.

Let us then assume a face detector and recognizer F whose output is the certainty:

$$G(n, m) = P\{s_m \text{ is imaged in } f_n\}$$

For each person s_m it is then possible to find all intervals $I_i^m = [a_i^m, b_i^m]$ such that $G(n, m) > 0,\ n \in [a_i^m, b_i^m]$ and $I_i^m \not\subset I_j^m, \forall i \neq j$. Using I_i^m we can define a *face occurrence* $F_i^m = \overline{G(n, m)}\big|_{n=a_i^m}^{b_i^m}$ as the average certainty within the interval I_i^m, that a specific person is imaged. So we can approximate $G(n, m)$ with

$$F(n, m) = \sum_i F_i^m \left[u(n - a_i^m) - u(n - b_i^m)\right] \qquad (1)$$

where $u(n)$ is the unit step function and $[a_i^m, b_i^m]$ is the i-th interval that contains the face of the m-th person.

Therefore, the video $\mathbf{V}$ is characterized by quartets of values $(s_m, F_i^m, a_i^m, b_i^m)$. Each quartet corresponds to a unique face appearance, i.e. it conveys the information that person s_m has been detected from frame a_i^m to frame b_i^m with a confidence of F_i^m. An example is given in Figure 1. The number of quartets in a video is equal to $\sum_{m=1}^{M} g_m \ll N \times M$, where g_m is the number of appearances of person s_m in the video, and N and M are the total numbers of frames and persons in the video.

2.2. Signature Similarity

In order to compare two signatures $F_1(n,m)$ and $F_2(n,m)$ which refer to a common set of faces, we need to find the optimal displacement d between the two sequences, which is the one that gives the highest face co-occurrence. We define co-occurrence as the area of the overlap between the rectangles of the quartets referring to the same person in the two signatures:

$$P_{\textit{co-occurrence}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\min(F_1(n,m), F_2(n+d,m))}{NM}$$

Thus the similarity of two signatures is defined as the maximum value of the co-occurrence, obtained when sliding one signature in relation to the other. Having established a method for computing the similarity between two signatures segments, searching for a specific video in a database entails simply comparing a candidate segment with the whole database and declaring a match when the similarity exceeds a certain threshold. However, doing this exhaustively is computationally unfeasible, and so we have developed an algorithm that does this in near-logarithmic time with respect to the size of the database.

2.3. Algorithm

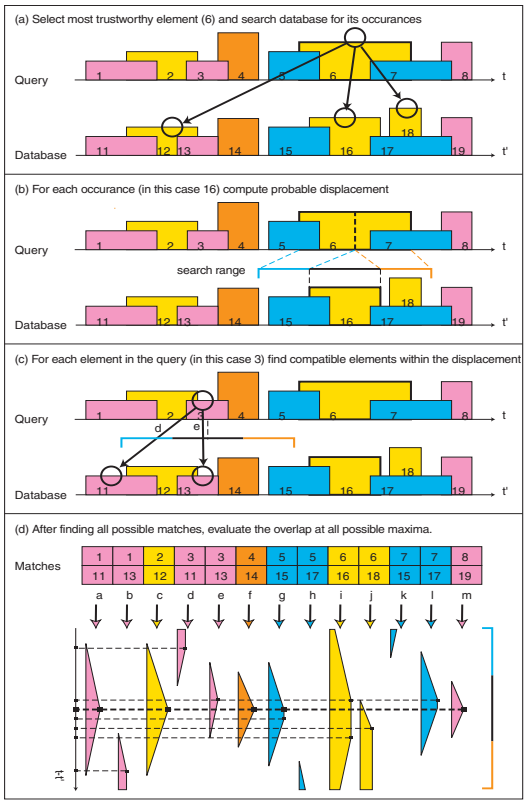

Fig. 2. Graphical overview of the algorithm. Colors correspond to distinct individuals. Signature elements are represented by numbered rectangles.

When the video database is initialized the face detector and recognizer is run to create the signatures, and then indexes on identity and time are created on them. It is assumed that the videos are arranged sequentially in the database.

The following algorithm (illustrated in Figure 2) is proposed for finding segments in the database which match a specific query segment:

1. Find the quartet with the greatest area (duration×certainty) in the query segment, in order to use it as a base for searching, and label it as the *trusted* quartet.
2. Find (through an index) all quartets in the database that refer to the same person as the trusted quartet found in step 1. These will be used as the base for evaluating the segments around them, and be named *base* quartets (Fig 2(a)).
3. For all base quartets found in the above step:
   (a) Add the pair consisting of the current base quartet and the trusted quartet into a new list L.
   (b) Calculate a displacement within which it will be possible to move the current base quartet and have it significantly overlap the trusted quartet. Possible matches may be found in the database using this displacement (Fig 2(b)).
   (c) Then, using the current base quartet, do the following for each quartet in query segment:
      i. Find (through the database indexes) the set of compatible quartets in the database, i.e. quartets that belong to the same person as the one in the query, and which are within the displacement computed in step 3b with respect to the base quartet selected earlier (Fig 2(c)).
      ii. If none are found, increment a counter n. If, for all query quartets examined with this candidate base quartet, $n > T_{\textit{reject}}$ where $T_{\textit{reject}}$ a heuristic threshold, proceed to the next compatible base database quartet.
      iii. Add the pairs consisting of the recovered quartets on one hand, and the current query quartet on the other, into the list L.
   (d) Evaluate the area of overlap of all pairs in the list L, computed for all displacements between the query segment and the candidate segment that correspond to possible maxima of the value of this area. These displacements can be proven to be only those that correspond to at least one quartet in each set having an endpoint equal to that of a quartet in another. As we have seen in Section 2.2, this equates to finding the optimal similarity when using this base quartet.
   Then select the maximum similarity and also keep the corresponding displacement.
   (e) Clear list L.
4. Select the final similarity as the maximum of the similarities computed with respect to each base quartet. If this is above a threshold T_v (which depends on the size of the query segment), then declare a match, otherwise declare no match.
5. Optionally, if no match is found repeat all above for the next most trustworthy quartet. In our experiments we have done so.

Note that if one wishes to continue verifying the retrieved segment, he can repeat 3c for the quartets beyond the initial query segment, keeping the computed displacement but adjusting the signature similarity and checking if it exceeds a modified threshold T'_v.

3. DERIVATION OF TEST DATA

Our focus here is to evaluate the performance of the proposed video indexing and fingerprinting method when applied on large video databases. However, the effort of applying different types of face detectors and recognizers on such a database (typically containing hundreds of hours of video), in order to derive the data required for the experimental performance evaluation of the proposed indexing and fingerprinting method, is extremely high. Thus, we performed the experimental testing of our algorithm on appropriately constructed artificial data. We have formulated a probabilistic model which describes the ground truth of the appearance of faces in videos, and a second probabilistic model which describes the behavior of the face detection and recognition module when used to derive the signature from the query segment. The output of these models is set of video signatures consisting of quartets. This approach has the advantage that we can easily test our algorithm on videos and face detection and recognition methods that have different characteristics, by varying the parameters of the models.

We model the appearance of persons in a video by considering the fact that the video is inherently composed of scenes, which are in turn composed of shots. Since scenes are spatio-temporally continuous in the context of the depicted world, and shots are spatio-temporally continuous in the video domain, we can assume different probabilities of appearance of a specific person for each scene and shot.

In order to construct the above model we needed three sets of information:

1. *The structure of the model*, i.e. the random variables it contains and their interrelations. This was constructed by analyzing the motion picture production process.
2. *The specific probability distributions of the random variables appearing in this model*. To estimate these, we have first manually annotated a moderately large corpus of video data by marking the faces appearing in it, and also the scene and shot boundaries present. We then tried to find appropriate distributions by using a combination of statistical testing (Kolmogorov – Smirnoff fit tests) and analysis of the physical meaning of the variables.
3. *The parameters of the above distributions* (mean, standard deviation etc). These were again computed from the manually annotated video data.

Due to lack of space we will not describe the details of the above model, such as the actual random variables used and their distributions. The model we use for representing the failure of face detection and recognition at query time is described in the next section. The model parameters can be obtained by running the respective algorithms on a sufficient set of data, but in our experiments we have varied them to explore the behavior of our method with respect to algorithms with different characteristics.

4. EXPERIMENTAL RESULTS

Two sets of experiments were performed, one for assessing computational performance, and one assessing robustness with respect to face detection/recognition errors.

Table 1. Average search times

Number of Videos	Search time
100	2 seconds
1000	5 seconds
10000	14 seconds

4.1. Computational Performance

To test the computational performance of our algorithm, we created artificial video databases of different sizes as described in the previous section. Each database consisted of a number of videos, each having a duration of 60 minutes and containing between 1000 and 2000 quartets. The number of different persons for each database was chosen to be 10 times the number of videos. We then selected query segments with an average length of 2.5 minutes and ran our search algorithm on these segments, using a commercial RDBMS system for the implementation. The average times of retrieval are given in table 1.

As it can be seen, the performance of the algorithm is near-logarithmic with respect to the size of the database.

4.2. Retrieval Performance

Given that neither face detection algorithms, nor face recognition algorithms are perfect, we performed a series of experiments to test the behavior of our algorithm in the presence of noise introduced during detection and recognition of the faces in the video. The following types of noise have been considered:

1. Change of quartet bounds. This is one of the most typical errors made by face detectors and trackers. Exponential noise has been added to the start time of a quartet (to simulate a delayed detection), and zero mean gaussian noise to the end time of the quartet (to simulate either early loss of target or false continuation of tracking). It should be noted that when the noise is high it can result in the complete elimination of a quartet. The mean (in the case of a quartet's start frame) and the standard deviation (in the case of a quartet's end frame) of the noise was varied from 1 to 5 seconds.
2. Change of the person's identity in a quartet. This is a typical error made by face recognizers. Here we assumed a chance (between 2.5% and 40%) that a person's identity would be changed to another random one.

This set of experiments was run using an artificial signature database of 1000 videos, each 60 minutes long. From this database we randomly extracted 4 sets of 10 segments each. The segments in the first set was chosen to contain 16 quartets, and had an average duration of 2.5 minutes, those in the second 32 quartets and 5 minutes, those in the third 48 quartets and 7.5 minutes, and those in the fourth 64 quartets and 10 minutes. On each set we added noise representing misbehavior of the face detector and recognizer, as described above, and then proceeded to search for them in the database.

The retrieval performance of our algorithm with respect to query segment size, detector noise and recognizer noise is shown in figure 3. Specifically, we examined if correct segments were found in the database (retrieval), incorrect segments were found (mis-retrieval) or no segments were found (non-retrieval). Results are only shown for 16, 32 and 48 quartet query segments, because for 64 quartets the retrieval is always perfect. We can see that for low noise levels

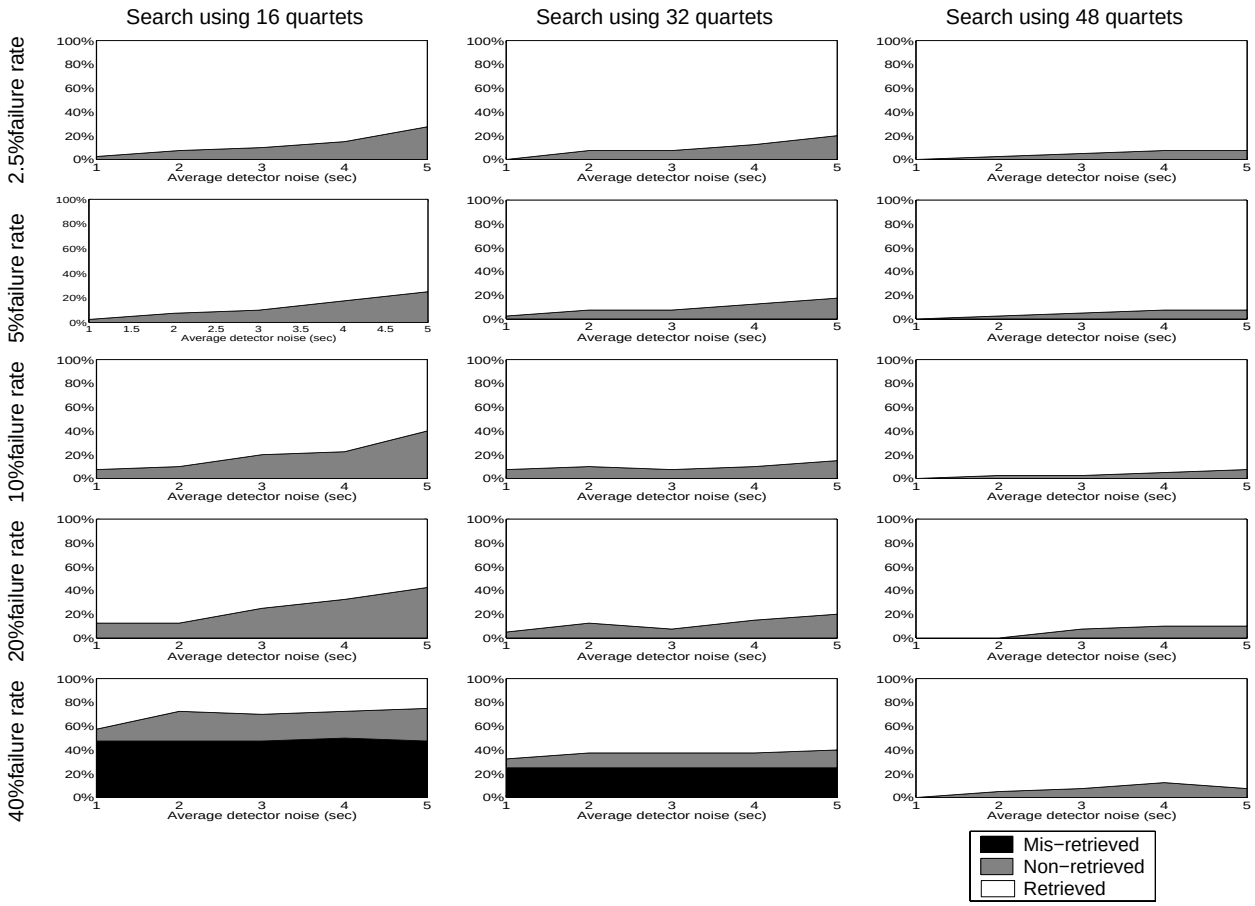

Fig. 3. Retrieval performance of the algorithm with respect to face detector and recognition performance and size of the query segment.

the performance is always satisfactory. For high noise levels, especially for detector noise, the performance drops dramatically when few signature quartets are used. However we should note that the average quartet length (both in real and artificial videos) is less than 4 seconds, while the added noise (change in the start and end frames) had a standard deviation that was as much as 5 seconds. In addition, increasing the length of the query segments greatly diminished the effect of the misbehavior of the face detection and recognition, virtually eliminating it when the length of the query segment reached 64 quertets.

5. CONCLUSIONS

We have presented a novel method for performing fast retrieval of video segments based on the output of face detectors and recognizers, with possible uses to indexing of video databases and fingerprinting of videos. The proposed method is both robust because it is based on a convolution-like similarity computation, and fast because it makes extensive use of database indexes. Testing was performed on artificial data based on models of the appearances of faces in videos and on face detector/recognizer behavior. The results verified that the proposed method performs very satisfactorily, both in terms of computational search efficiency (even in a database of 10000 hours of video), and in terms of recall.

6. REFERENCES

[1] Nevenka Dimitrova, Hong-Jiang Zhang, Behzad Shahraray, Ibrahim Sezan, Thomas Huang, and Avideh Zakhor, "Applications of video-content analysis and retrieval," *IEEE Multimedia Magazine*, vol. 9, no. 3, pp. 42 – 55, July 2002.

[2] Yin Chan, Shang-Hung Lin, Yap-Peng Tan, and S.Y. Kung, "Video shot classification using human faces," in *IEEE International Conference on Image Processing (ICIP)*, 1996, vol. 3, pp. 843–846.

[3] Stefan Eickeler, Frank Wallhoff, Uri Iurgel, and Gerhard Rigoll, "Content-based indexing of images and video using face detection and recognition methods," in *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2001.

[4] Shin'ichi Satoh, "Comparative evaluation of face sequence matching for content-based video access," in *Proc. of the 4th Intl Conf. on Automatic Face and Gesture Recognition(FG2000)*, 2000, pp. 163 – 168.

[5] M. Viswanathan, H.S.M. Beigi, A. Tritschler, and F. Maali, "Information access using speech, speaker and face recognition," in *IEEE International Conference on Multimedia and Expo (ICME)*, July-August 2000, pp. 493–496.

[6] Gang Wei and Ishwar K. Sethi, "Omni-face detection for video/image content description," in *Proceedings of the 2000 ACM workshops on Multimedia*, Nov 2000.