Pattern Recognition Letters 33 (2012) 1882–1890

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

M-VIVIE: A multi-thread video indexer via identity extraction

Maria De Marsico [a,*], Gianfranco Doretto [b], Daniel Riccio [c]

[a] Department of Computer Science, Sapienza University of Rome, Italy
[b] Lane Dept. of Computer Science and Electrical Engineering, West Virginia University, United States
[c] Biometric and Image Processing Laboratory, University of Salerno, Italy

ARTICLE INFO

Article history:
Available online 24 March 2012

Keywords:
Video indexing
Face recognition
Identity management

ABSTRACT

In this work we present M-VIVIE, a system for video sequence indexing based on the identity of appearing subjects, which exploits a multithread architecture for fast processing. The system is composed of more conceptual component modules, each performing a specific kind of processing. Each module can possibly be substituted with a different one performing the same task. Special attention is devoted to classification and clustering activities. M-VIVIE presents a number of peculiarities, that distinguish it from existing systems with similar goals. Among these, the account for the concomitant appearance of two identities in the same clip, and the use of such information in the process of identity mapping. M-VIVIE was tested on different kinds of real video clips to assess both its efficacy and efficiency.

1. Introduction

The soaring creation, diffusion and storage of video information is a direct consequence of the reduced costs of technology, as well as of the increase of the available network bandwidth. Moreover, highly demanding security applications often require to organize and handle great amounts of video data. These are possibly accessed on demand on a later time, for example to answer spatial and/or temporal queries related to objects and/or individuals present in a protected area (where and when). In an ideal system, such operations should also be possible online. In a large scale monitoring system (e.g. the surveillance system in a public area) it is extremely important to support the activity of a human operator in detecting specific events. For example, a problem which is often faced is the ability to browse a video according to the subjects contained in different sequences, in order to trace a person walking through different places in different times. This is a typical reidentification problem, requiring to match a person across multiple disjoint fields of view. In general, the object reidentification task is to match an object across space and time. In other words, once we have detected it in one location and at a certain time, we want to identify it again, in a set of possible candidates, when it moves to a different location, at a possible different time. Popular pattern recognition techniques can be exploited, but when the object of reidentification is a person, biometrics play a special role. In particular, face- and gait-based recognition are able to support systems that typically work at a distance. On the other hand, it is often the case that the obtained information must be shared among different organizations to enforce surveillance and security, as well as for investigative needs. In order to navigate a video, divide it into sections containing interesting scenes, or store it according to predetermined criteria (e.g. presence of a person) it is necessary to analyze its content to allow the creation of appropriate indices. Applications which can index video scenes according to the identity of present subjects are particularly interesting, as a number of works suggest (Choi et al., 2010; Torres and Vilà, 2002); such identities can be determined through face recognition (Abate et al., 2007). In practice, a surveillance system is equipped with a biometric module able to discriminate subjects appearing in video clips. It is worth noticing that in this context one is often not much interested in determining the actual people identities, but rather in the pure ability to detect different scenes presenting a same subject. Many present reidentification systems rely on easy-to-compute features, such as color histograms or object silhouettes. These are computationally affordable and able to satisfy real-time requirements, but suffer from low accuracy. On the contrary, the detection and classification of present subjects faces provide more robust indexing cues, but they are also much more complex to perform. Most present face recognition techniques can be hardly applied to poorly controlled settings, due to extreme variability caused, among other factors, by pose and illumination changes (Abate et al., 2007; Becker and Ortiz, 2009). Nevertheless, face represents a biometric trait that is particularly suited for the applications sketched above, since it can be captured at a distance and from unaware subjects. Due to continuous advances in technology, one can presently afford higher resolution and more demanding procedures to address uncontrolled image capture, without spending too much more computational time. In this context, appropriate

* Corresponding author.
E-mail addresses: demarsico@di.uniroma1.it (M. De Marsico), gianfranco.doretto@mail.wvu.edu (G. Doretto), driccio@unisa.it (D. Riccio).

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.03.005

normalization and classification techniques can be an answer to the problem of high intra-class variability of face patterns.

2. System architecture

M-VIVIE is a completely automatic video indexing system which is based on face recognition. It is composed of a number of modules, as shown in Fig. 1, which interact to process and annotate the single frames of an input video sequence according to the present identities. These identities are detected through the face samples associated to them. Pose and light normalization aim to reconstruct a frontal face pose under uniform illumination, to make classification more robust. Each frame in a video sequence undergoes a pipe of operations, which will be detailed in the following sections: (a) face detection, (b) face tracking, (c) face normalization, (d) feature extraction, (e) face classification, (f) face clustering (Fig. 1). Such operations are distributed across the different modules that make up the final architectural design. It is worth noticing that each of them can be substituted with one running a different algorithm, given that interfaces (input/output data) remain fixed. We included detection and tracking in the same module, since in our design the latter only accounts for the subsequent face positions determined by the former. The actual presence of the same person will be verified later by the clustering module. That is why we distinguish between *temporary* and *permanent* identities. The same clustering module will also be in charge of controlling possible dependencies able to improve computation. As we will see, in our case we prune the number of matches to determine a permanent identity for a temporary one, based on spatial/temporal constraints produced by assignments already done (two temporary identities which appear in a same frame cannot belong to the same permanent identity). Modules which perform detection, tracking, normalization and feature extraction can work in multi-threading mode as well, and can be possibly spread over more computers.

The first operation on a frame, after its extraction from input video, is the crucial detection and location of all present faces. These are recorded together with their position, dimension, and quality. Position information for each face is submitted to the tracking module. As long as a face is present in the scene, it is tracked and all its corresponding samples, which are collected according to spatial coherence from a frame to the next one, are processed: they undergo normalization and feature extraction, and are then stored in a temporary list. As soon as a certain face is not detected anymore in the scene during a number of consecutive frames F_{not}, all of its corresponding samples are sent to the classification and clustering modules. The parameter F_{not} is set by the user depending on the characteristics of the video, e.g. processing a clip with frequent and sudden shot changes requires lower values. The clustering module either assigns samples to an existing identity, or, if this is not possible, a new identity is created. The specific implementation of classification, which is performed after each full tracking cycle, is the core of our approach to reidentification in a long video. Reidentification requires to match a person across possibly multiple disjoint fields of view, or across different temporal periods. To this aim, after a face has been detected and tracked at one location, we match it with a feasible set of candidates detected at other locations/in different times. During these phases, an "identity" does not necessarily correspond to a specific recognized person, unless a real-time identification is required, but to a generic subject which is distinguished from the others. When the whole video has been processed, the system returns the set of created identities and the corresponding samples, with the indication of the points in the video where they were detected. At this point, identities may undergo a real identification/recognition process. Let us notice that the outcome provided by M-VIVIE is more articulated and useful for a number of applications, than the mere detection of a certain identity in a video (yes/no answer). For example, it might be used to measure the amount of time a person has been framed during a video sequence, to measure the permanence in a certain area, or results from videos from different cameras may be correlated to detect the path through a series of areas.

Details of each module and of its working are given in the following.

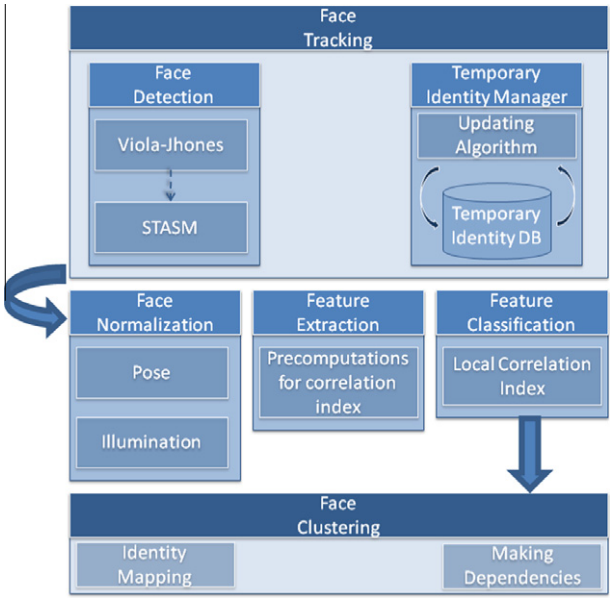

Fig. 1. System architecture. Video is decomposed in frames which are first inputted to the face detection module and then to the face tracking module. Following the pipeline, face samples are normalized and classified to be submitted to the face clustering module, which transforms "temporary" identities in "permanent" ones.

3. Face detection and tracking

Face(s) location is the first operation which is performed by the system and is a crucial step for the following face tracking task. The concerned module analyzes each single video frame and returns all detected faces in it, together with information including position, expressed in cartesian coordinates with respect to the image origin, the dimension, represented as a pair of integer scalars (width and height), and a quality index, expressed by a real scalar (quality indices are detailed in the following). Location is performed by implementing Viola–Jones' algorithm (Viola and Jones, 2004). For each detected face within the frame, the algorithm returns a Region Of Interest (ROI). This represents the input for an Extended Active Shape Model, implemented by STASM software as described in (Milborrow and Nicolls, 2008). STASM applies a generic shape made of 68 points to the input ROI, and performs an iterative refinement process, which is based on reducing a cost function with respect to a previously performed training on a separate set of images. This process readapts the generic shape to the sample at hand by disposing each point on its corresponding zone. Nose tip has a crucial weight in the normalization process; however, since it is a prominent anatomical structure, it is often subject to a higher location error. Such error may hamper the outcome of the following normalization phase. M-VIVIE addresses this problem in a twofold way: it exploits a quality measure based on face symmetry, and discards face samples which are too far from the rest of the gallery during the *identity mapping* process, which will be described later.

Once a face has been detected within a frame, M-VIVIE traces it along the following ones in the sequence. A number of works in literature (Torres and Vilà, 2002) that deal with face indexing from

video sequences, do not implement a real face tracking module: faces that were extracted from the single frames are directly submitted to the classification process. For instance, in (Choi et al., 2010 all faces extracted from a shot (sub-sequence of frames selected by an automatic algorithm) directly undergo a clustering algorithm, which groups them in classes according to a distance (dissimilarity) measure. The two faces with lower distance are included in a single class, if such distance is under a predetermined threshold. The process is iterated until no more pairs are found with a distance lower that the threshold. However, due to inter-class similarity or location errors, it may happen that two faces of different subjects are included in the same class, so invalidating the results of the following steps. Moreover, due to pose or illumination variations, samples of the same subject may be included in different classes (identities). This latter problem is attenuated by our approach, since the presence of a face tracking module in M-VIVIE just aims at avoiding such cases. This module maintains a set V of the faces being currently examined, which is updated after each frame processing. The set is initially empty. Afterwards, for each face detected in the first examined frame, an element $v_i \in V$ is created, which is denoted as a *temporary identity*. Such identities are distinct from *permanent identities*, which will represent the final outcome of the process and whose details will be discussed in Section 5.1. For each *temporary identity,* the coordinates $(x_{\text{nose}}, y_{\text{nose}})$ of the nose tip are recorded. Moreover, each such identity is assigned a list $L(v_i)$, which collects its possible further samples that may appear later in consecutive frames. During the processing of the following frames, when the face tracking module receives a new face sample from the face detector, it tries to assign it to an already existing identity in V. To this aim, it computes the distance between the coordinates $(newx_{\text{nose}}, newy_{\text{nose}})$ of the newly received sample and those of each *temporary identity* already in the set, after normalizing them with respect to frame size. If a pair of coordinates is found for which such distance is lower than a threshold th_{nose}, the new sample is considered as an instance of the corresponding *temporary identity,* and is added to its collection. Otherwise, a new element is added to V. Even the value for th_{nose} is set by the user, according to the characteristics of the video being processed. The quicker the expected movements, the higher the value. The coordinates $(x_{\text{nose}}, y_{\text{nose}})$ of the matching identity are updated with $(newx_{\text{nose}}, newy_{\text{nose}})$. Each *temporary identity* is therefore characterized by the coordinates $(x_{\text{nose}}, y_{\text{nose}})$ of the nose tip in its last detected instance (sample), and this is the relevant element for the following distance matching operations. This is a reasonable choice: since we assume a fluent motion, it is useless to include previous positions in the match. Before continuing, it is worth noticing that in this step the assignment of a detected face sample to an element in the set V does not take into account the identity of the subject, but only the congruence of samples' positions in consecutive frames. If a shot change happens, the same person may appear in a different position, and therefore its face samples may be assigned to different *temporary identities,* or on the contrary a new person may appear in a certain position and its face samples be assigned to the same *temporary identity* of the person formerly occupying that position. The first problem is automatically addressed during our *identity mapping*. The second one is trickier, as it will be more evident from the description of the mapping process. It could be addressed by integrating the process with a robust shot detection method. Once a sudden scene change is detected, e.g. through a sudden background change, this would trigger a reset of the tracking procedure. This reset would map at the same time all temporary identities detected so far, and not yet classified, into permanent ones. We remind that, in normal conditions, each temporary identity is mapped into a permanent one as soon after the corresponding face has not been detected during a number of consecutive frames F_{not}. After this "mass" processing, the tracking would be restarted from scratch. The integration of such a mechanism in our architecture will be object of a future development. Nevertheless, it is to say that the preliminary pre-assignment as it is, is still satisfactory enough in supporting feature extraction and classification procedures, which are performed later. Moreover, shot changes are not expected in surveillance videos, unless we consider videos from PTZ cameras. The described procedure also presents a further positive aspect. If u e v are faces which appear at the same time in a frame, they cannot belong to the same person. This information, though neglected by most works in literature, is of paramount importance, since it gives two crucial advantages: (a) it reduces misclassifications, (b) it reduces the computational cost of the algorithm, since we know in advance that some classes must not be compared among them. This will be detailed later.

4. Face normalization and feature extraction

Face normalization is often exploited in literature to improve classification accuracy. We can mention the algorithm proposed by Blanz and Vetter (2003) among the most efficient ones. It computes a 3D face model, starting from one or more 2D images and modifying a 3D generic model (morphable model). In principle, it allows to simultaneously correct both pose and illumination. The reverse of the medal is a significant computational cost, most of all when processing a high number of faces.

Both normalization of single face samples, and feature extraction and matching are performed in M-VIVIE by exploiting the FACE approach (De Marsico et al., 2010). We summarize it below for readers' convenience.

4.1. Face normalization

Though starting from the set of 68 relevant points that are located by STASM, FACE uses only 13 of them, as shown in Fig. 2 (the red/darker ones). Head rolling is corrected using the centre of the eyes (Fig. 2(a)). The distances between the external corners of the left and right eye and the tip of the nose, are respectively denoted by d_l and d_r in Fig. 2(b). If $d_r \geqslant d_l$ the right half of the face is the better exposed (best) one, and the image is left unchanged; otherwise, it is reflected with respect to the vertical axis (horizontal flip). This trick allows to always consider the right hand side of the face as the best one in the following processing steps. The points in Fig. 2(c) make up the face central line. Notice that the first point

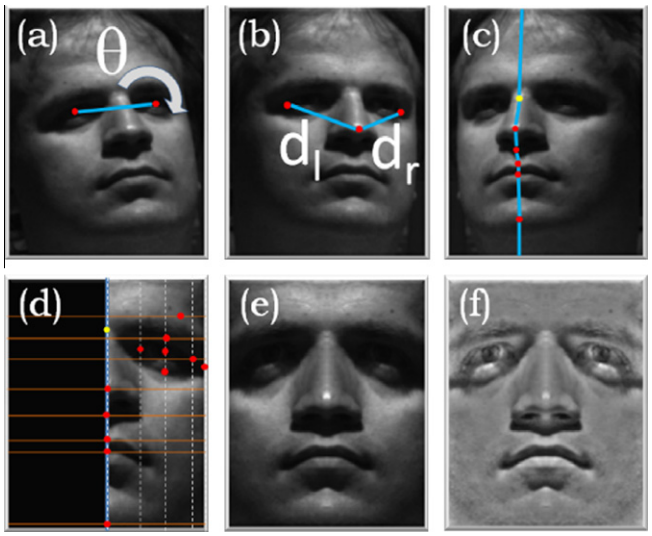

Fig. 2. The six main steps of the pose and illumination correction process.

upward (the light one) is not provided by STASM. It is derived as the median point of the link between the first two points upward on nose sides. The face central line and the image borders delimit the right and left face regions. Only the right (best exposed) one is considered. Its rows are stretched to obtain a constant length. Then the half face is divided into horizontal and vertical bands, which are determined by the lines passing through some of the interest points as shown in Fig. 2(d). The choice of such points privileges the delimitation of bands that are more significant for the normalization process, namely those respectively containing the eyebrow, the eye, the nose and the mouth. The lines are resized, to make the chosen interest points fall in pre-determined positions. Finally, the left half face is reconstructed by reflecting the right half one (Fig. 2(e)). This step allows using FACE normalization procedure even when the following matching program exploits measures related to the whole face (e.g. the distance between eyes).

After pose correction, the face image undergoes illumination correction. This is performed through the Self Quotient Image (SQI) algorithm described in (Wang et al., 2004. The value of each image pixel is divided by the mean of the values in its neighbourhood, represented by a square mask of size $k \times k$ (in our case $k = 8$). The final result of the overall normalization process is shown in Fig. 2(f).

4.2. Feature extraction

A recent work by Becker and Ortiz (2009) mentions a benchmark including some typical image datasets from Facebook. The reported experiments demonstrated that many of the most well-known techniques for face classification (Abate et al., 2007) are still too sensible to image distortions, to be profitably used in partially or totally uncontrolled settings. FACE performs image matching by a localized version of the spatial correlation index. In its global form, given two images A and B and the respective mean values of their pixels $\bar{A}$e $\bar{B}$, it is defined as:

$$s(A,B) = \frac{\sum_{i=0}^{n-1}\sum_{j=0}^{m-1}(A(i,j)-\bar{A})(B(i,j)-\bar{B})}{\sqrt{\sum_{i=0}^{n-1}\sum_{j=0}^{m-1}(A(i,j)-\bar{A})^2\sum_{i=0}^{n-1}\sum_{j=0}^{m-1}(B(i,j)-\bar{B})^2}} \quad (1)$$

In FACE, it is adapted to work locally, over single sub-regions r_A and r_B of fixed size in images A and B. For each sub-region r_A in A, identified by the position of its left-upper corner, we search, moving in a limited window around the same position in B, the region r_B which maximizes the correlation $s(r_A, r_B)$. The global correlation $S(A,B)$ is the sum of the local maxima. This approach is more accurate, but also more computationally expensive, than the global one. However, some tricks, such as the pre-computation of some quantities in the matching formula, allow to perform a considerable amount of matches (hundreds) in less than one second on medium-low band computing equipment. The pre-computed values of elements in (1) are those that do not involve cross-computation. They also represent the features extracted for future matching, making up the biometric key associated to each face.

4.3. Multi-threading

The above operations are quite computationally expensive and therefore hardly lend themselves to fast processing, let alone to online processing. In order to speed up the overall process, the computation can be parallelized by adopting a twofold strategy. First of all, any time a new temporary identity is detected, a new thread is started to track it, possibly on a different connected machine which runs the same tracking module. The same happens for normalization. In a multi-processor (multi-host) system, the host that has to run the new thread can be chosen according to any load balancing algorithm. This kind of architecture lends itself to both single-camera e to multi-camera processing, since the choice of the thread which has to process a specific temporary identity is independent of the source video. When no instance (face sample) can be associated to the *temporary identity* v_i for F_{not} consecutive frames, this processing step for such identity is completed, and v_i must be mapped onto a *permanent identity* I_k. Association among temporary and permanent identities follows a multi-thread protocol as well, which will be described later. Therefore, a mapping thread is identified to which the temporary identity and its samples are transmitted.

5. Face classification and clustering

In Section 3 we introduced the concept of *temporary identity*, anticipating the difference with *permanent identities*. A *temporary identity* is created to temporarily represent a set of samples which are grouped together merely due to their positions within temporally close frames. It only exists until possible instances of the tracked face are detected. When no instance (face sample) can be associated to the *temporary identity* v_i for F_{not} consecutive frames, such identity must be mapped onto a *permanent identity* I_k. This task pertains to the face classification module through an *identity mapping* operation. Identity mapping performs two important activities: a) it analyzes the samples of *temporary identity* v_i and compares them with those of *permanent identities* I_k where $k = 1,2,\ldots,|I|$ and $|I|$ represents the number of existing permanent identities (possibly a new permanent identity I_w is created); b) if no new identity I_w, has been created from v_i, it merges samples associated to v_i with those of the I_k with which v_i has been identified, by selecting the most representative ones from both sets.

Let us consider a generic *temporary identity* $v_i \in V$, for which $|L(v_i)|$ instances (face samples) $v_{i,h} \in L(v_i)$ have been collected. As introduced above, the process of mapping this identity onto a permanent one is performed in two consecutive steps: (a) permanent identity definition, (b) reduction of representative face samples contained in the gallery. It is first necessary to devise a criterion to assess compatibility among face samples of a temporary identity and those of a permanent one, so that they can be possibly associated. Given this, we can face two different situations. In the first one, there already exists a permanent identity for which the temporary one is an instantiation. In other terms, a face was detected and traced that had already been previously detected, and for which a permanent identity had already been created. This case requires more attention, since face samples of the temporary identity must be merged with those already existing in the gallery, and corresponding to the associated permanent identity. The iteration of this process might significantly increase the number of face samples in the gallery, if the number of processed frames is high. To avoid such proliferation, it is necessary to prune the samples to be merged, only preserving the most representative ones. This requires to define suitable rules, allowing to merge the two sample lists (temporary and permanent) and to select from their union a reduced yet significant subset.

In the second case, no existing permanent identity is sufficiently similar to the samples of the temporary identity to be associated with it. Therefore it is necessary to create a new permanent identity, whose gallery of face samples will be that of the temporary one. As a result, the second case is much easier to deal with than the first one, since the gallery of the temporary identity is simply copied into that of the newly created permanent identity.

5.1. Identity Association

Given a *temporary identity* $v_i \in V$ with $|L(v_i)|$ instances (face samples) $v_{i,h} \in L(v_i)$, and a *permanent identity* I_k, with $|L(I_k)|$ face samples $I_{k,j} \in L(I_k)$, it is necessary to test if they are compatible, i.e. if it is

possible to associate the identity v_i to I_k, and to merge the respective face samples in a single set.

Compatibility (similarity) between identities is measured by computing a distance matrix $M(n,m)$, where $n = |L(v_i)|$ and $m = |L(I_k)|$. M is defined such that $M(h,j) = 1 - S(v_{i,h}, I_{k,j})$, i.e. the entry (h,j) contains a measure of distance between the hth face sample of v_i identity and jth face sample of permanent identity I_k, which is in inverse relation to the localized version of image correlation introduced in Section 4.2. Once a threshold $th_C \in [0,1]$ is set, the number C_M of entries such that $M(h,j) < th_C$ is counted. In practice, the threshold th_C represents the distance value below which two face samples can be associated to the same person in a verification process (we experimentally found that $th_C = 0.3$ is a suitable choice, for the classifier which we exploited). Therefore, C_M represents the number of pairs of face samples that can be considered as captures of the same subject; this value is then normalized with respect to the number of considered pairs, so to obtain a value in the interval $[0,1]$. A further threshold th_M, which is set by the user, allows to decide if *temporary identity* v_i can be associated to the *permanent identity* I_k; the association will be performed if $C_M > th_M$ (we experimentally observed that 0.1 is a suitable value). See Fig. 3 for examples of surface representations of the distance matrices for comparisons of incompatible and compatible sets of samples.

If the association is feasible, it is necessary to update the set $L(I_k)$ of the involved *permanent identity* I_k. This is done by analyzing the values in M matrix. First of all, the mean $\overline{M}$ of values in M is computed. A new matrix $\hat{M}$ is obtained from M, by subtracting the value $\overline{M}$ from each of its elements, and setting to zero all those elements producing a negative difference. In practice:

$$\hat{M}(h,j) = \begin{cases} M(h,j) - \overline{M}, & \text{if } M(h,j) > \overline{M} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Given the matrix $\hat{M}$, M-VIVIE defines two vectors *ROWS* and *COLS* with length respectively equal to the number of rows and to the number of columns of the matrix. Vector *ROWS* is defined such that its hth position *ROWS*[h] reports the number of elements (columns) of the hth row in $\hat{M}$ such that $\hat{M}(h,j) > th_{\overline{M}}$, where $th_{\overline{M}}$ is set by the user (we experimentally observed that 0.1 is a suitable value). Vector COLS is defined in a symmetrical way, with respect to columns of $\hat{M}$. In practice, vector *ROWS* (*COLS*) counts the number of permanent samples for each temporary sample (respectively, the number of temporary samples for each permanent sample) which are at least $th_{\overline{M}}$ farther away than the mean. This helps identifying distorted face samples, and therefore items that, though not always being error sources, may disturb the following processing. As a matter of fact, if many "critical" elements (those which satisfy the condition $\hat{M}$ such that $\hat{M}(h,j) > th_{\overline{M}}$) are found in a row of $\hat{M}$, such row must be discarded from future operations. This symmetrically holds for columns. We experimentally fixed the maximum admittable number of critical elements to ¼ of the total number of elements in the row/column.

Samples which survive the pruning phase, because they are not affected by possible distortions, must be further screened. Otherwise, the number of face samples representing a permanent identity would excessively grow at each identity mapping operation. A huge number of face samples per permanent identity would require higher time to compute the distance matrix M. Moreover, this may result useless, since many face samples would be very similar and therefore poorly representative (see Fig. 4). To be able to reduce the number of face samples after merging a permanent and a temporary identity, it is essential to have a selection criterion, which assures a good gallery representativeness. In our specific case we considered two different strategies: (a) Minimum Average Score (MAS); (b) Maximum Score vector Distance (MSD).

5.2. Minimum Average Score

After discarding rows/columns corresponding to distorted face samples, M-VIVIE identifies the row h_{min} with the lowest mean value; the corresponding face sample enters the new set $L'(I_k)$ that we are building for the permanent identity I_k, since it represents the face sample in $L(v_i)$ which is less far on the average from all those in $L(I_k)$. In order to increase intra-class variability, and therefore to make the new set $L'(I_k)$ as much representative as possible, we scan all entries in h_{min} row and, for each j such that $\hat{M}(h_{min},j) < th_s$ we select the corresponding face sample $I_{k,j} \in L(I_k)$ to be included into $L'(I_k)$. In a symmetric way, M-VIVIE identifies the column j_{min} with the lowest mean value and inserts the corresponding face sample into $L'(I_k)$, then scans the column and for each element such that $\hat{M}(h_{min},j) < th_s$ inserts the corresponding face sample $v_{i,h} \in L(v_i)$ in $L'(I_k)$. At the end of this process, the set $L(I_k)$ for the permanent identity I_k is substituted with $L'(I_k)$.

5.3. Maximum Score vector Distance

In this case M-VIVIE scans the distance matrix M to identify the score row (*score vector*) h_{max} with the highest variance. Such row corresponds to the face sample in the gallery of the temporary identity $L(v_i)$, which is the most characterizing with respect to those in permanent identity gallery $L(I_k)$. Once such face sample is selected and inserted into the new set $L'(I_k)$, that we are building, all the face sample which are too close to it are discarded. Formally,

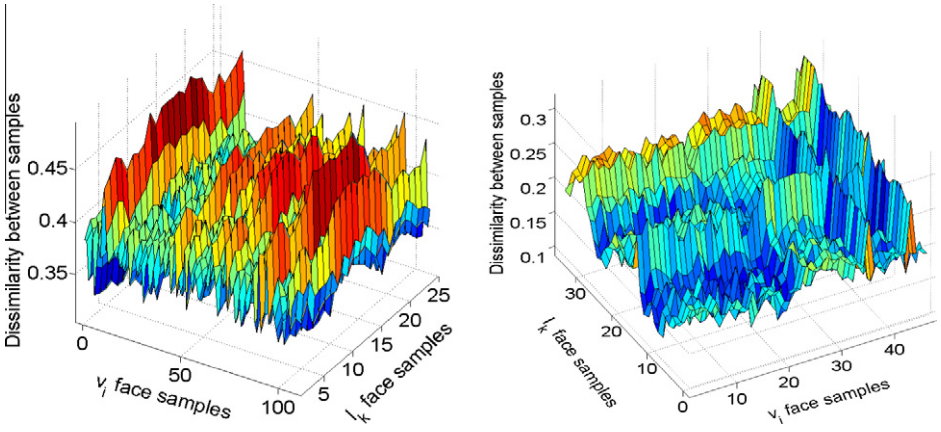

Fig. 3. Surface representation of the distance matrix. Two distinct cases are shown: the temporary identity v_i and permanent identity I_k are not compatible (left) or compatible (right).

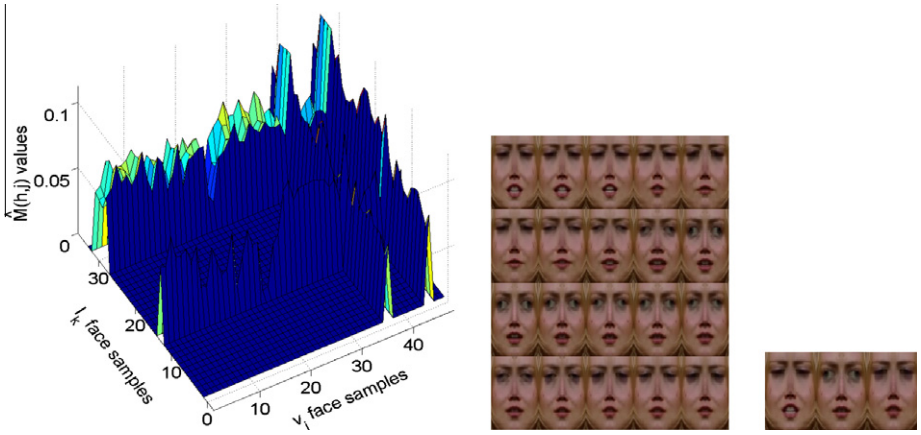

Fig. 4. A surface representation of matrix $\hat{M}$ on the left. On the lower right, three face samples that were extracted from the full set, some of which (only 20) are shown in the upper right.

if h_{max} represents the row for the face sample at hand, we search among all rows h in matrix M which have not been selected yet a new row h_{next} with maximum Euclidean distance $d_{max} = ||h_{next} - h_{max}||_2$ from h_{max}; if such distance is above a predefined threshold th_{max} the corresponding face sample is inserted into in $L'(I_k)$, and the search for a new h_{next} is started, otherwise the search process stops. A symmetrical process is performed for columns in M. The threshold th_{max} allows to regulate the approximate number of face samples that will compose the new gallery $L'(I_k)$; a too high value produces very small face sample sets, so reducing processing times, but also introduces the possibility to be too selective and to generate a gallery which, though extremely varied, contains too few samples per permanent identity to be actually representative. On the other hand, a too low value would hinder an effective sample selection.

5.4. Parallel identity association

Each host which has to run identity association must have a complete, up-to-date list of the present permanent ones, together with their samples. When it receives a temporary identity to map, it starts a mapping thread which performs the operations described in the section above. Once a permanent identity is updated, its samples are sent to a central mapping coordinator, which knows all the mapping hosts and broadcasts the information. If a new permanent identity is created, it is transmitted together with its samples to the central mapping coordinator as well. If the coordinator receives only one new identity, it broadcasts it as is. Otherwise, if it receives more new identities in the same cycle, it checks if they can be mapped one another, by using the same procedure followed to associate temporary to permanent identities. At the end, the coordinator broadcasts the final result with possible merging of received identities. Fig. 5 conceptually depicts the overall described architecture.

5.5. Handling dependencies

Most works in literature, which deal with face classification from video sequences, do not take into account the information given by the concomitant appearance of more subjects within the same scene. However, this information is extremely useful, because it allows to decide with certainty that two identities are not compatible (for compatibility we mean here the concept expressed above, i.e. the fact that they can be associated with a same person). In other words, it allows to decide that two temporary identities cannot be associated with the same permanent one.

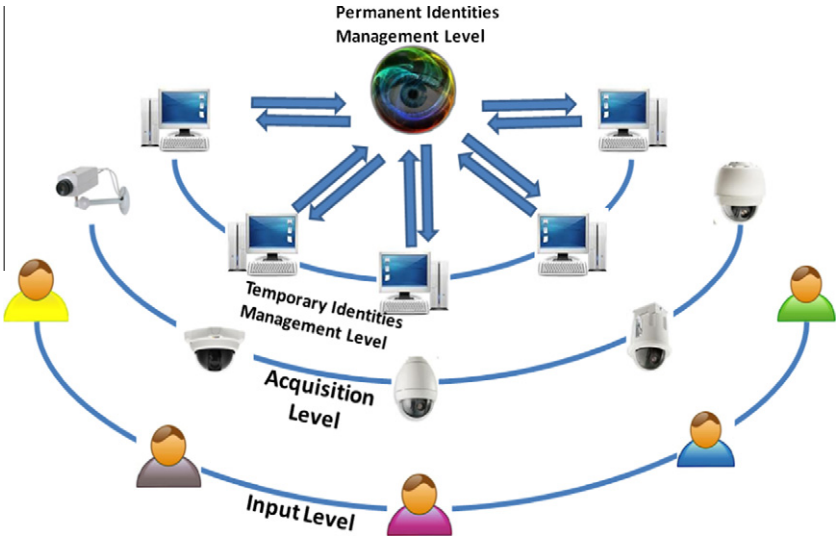

Fig. 5. A conceptual representation of how the identity association overall work is divided between parallel threads which are coordinated with a central management module.

The derived advantage is that a number of *identity mappings* will not be tried, therefore achieving a twofold result of reducing the algorithm computational cost, and increasing the accuracy of the system.

The process of creating dependencies between identities is very tricky, and relies on the specific philosophy that one wants to follow in analyzing video sequences. In particular, two different strategies exist. In the first case, one proceeds off-line, i.e. all temporary identities are extracted from the video sequence, and only at the end of this process they are mapped onto permanent identities. In the second case, one works on-line, therefore temporary identities are mapped onto permanent identities as they are identified. Off-line strategy is much simpler to deal with, since it is sufficient to create a structure, a sparse matrix, where a bit is set to 1 if the two temporary identities (row and column) appear in the same scene. From the moment when a temporary identity is mapped onto a permanent one, mapping with the same permanent identity will not be attempted for all those which are marked as incompatible with it. On the contrary, the on-line strategy requires more care, because we do not have in advance the complete set of temporary identities, as they are identified from time to time and some of them could have already been mapped onto a permanent identity. In this case, one has to track the compatibility between temporary identities even when some of them have already been mapped. To this aim, M-VIVIE associates to each temporary identity v_i two vectors A_{vi} and B_{vi}, of length equal to the expected total number of identities to deal with. If two temporary identities v_i and v_j appear in concomitance in the same scene, this is recorded by setting $A_{vi}(j) = 1$ and $A_{vj}(i) = 1$. When one of the two temporary identities (let us assume for example v_i) is mapped onto permanent identity I_k, this is recorded by setting $B_{vj}(i) = k$. In other words, as for temporary identity v_j, M-VIVIE records that identity v_i, with which it is incompatible, has been mapped onto permanent identity I_k. Such information is exploited during the following *identity mapping* steps; as a matter of fact, when for example v_j must be mapped, only permanent identities which do not appear in B_{vj} are considered. Notice that when a temporary identity is mapped onto a permanent one, it is destroyed together with all related information.

The central mapping coordinator also receives and processes updated versions of the *A* and *B* vectors.

6. Experimental framework and results

Experiments aimed at highlighting the performance of modules making up M-VIVIE system. We present some significant results which are related to face detection and normalization, with more emphasis on the evaluation of the performance of classification and clustering modules, since they represent the most innovative part of this work. Special attention was devoted to the identification of thresholds and constants which are involved in the different processing steps, and to the assessment of their stability and consistence.

M-VIVIE was tested on two different sets of clips, GENERIC and SURVEILLANCE. The set GENERIC was composed of a total of 16 video clips, which are different for type and duration. A set *ClipA* of 8 clips included sequences extracted from 4 different TVNews, 2 for each edition. A second set *ClipB* of 8 out of the 16 clips was extracted from 4 films, 2 for each film, selected from different passages. Clips in *ClipB* are "harder" than those in *ClipA*, due to shot changes, possible variations in illumination, and accentuated pose and expression changes. Moreover, they are harder than clips generally produced in video-surveillance too, since the latter are expected to present much less shot changes, even when we consider videos from PTZ cameras.

The first experiments on GENERIC set verified the performance of face tracking module on clips from *ClipA* and *ClipB*, separately. We evaluated how the variation of threshold th_{nose} affects the performances. A too low threshold might make sudden and emphatic movements to trigger the assignment of face samples from the same subject to different temporary identities. This does not cause any problem in the final classification of samples, since the different temporary identities will be mapped onto the same permanent one with very high probability. This would also demonstrate the robustness of the clustering method. The disadvantage is represented by the increased number of *identity mapping* operations. A suitable choice of such value then affects the rate between effectiveness and efficiency of the system. Fig. 6 shows a plot of the number of *identity mapping* operations while varying threshold th_{nose}.

Fig. 6 highlights how th_{nose} produces an higher increase of identity mapping operations in *ClipB*. In fact, in action scenes faces span longer distances due to sharp movements. Moreover, we can observe that performances settle when th_{nose} approximately equals inter-ocular distance. The latter can substitute th_{nose} to obtain an adaptive system.

The clustering module underwent two different tests using GENERIC: (a) the ability to associate all face samples of the same subject to the same permanent identity, (b) the ability to identify the same subject (by associating the different permanent identities) which is depicted in different clips. Table 1 reports the results of the first experiment: for each clip, it shows the number of permanent identities which were determined versus the real total number of present identities (target identities). Execution time depends on the input video length and on the number of performed identity mappings.

M-VIVIE more often produces a number of permanent identities higher than target identities when processing ClipB. This is due to quantity and quality of face distortions (pose, illumination, make-up). However, some of these identities are characterized by a very low number of samples. This may determine heuristics to identify wrong permanent identities. In a further experiment with GENERIC we measured False Acceptance Rate (FAR) and False Rejection Rate (FRR) achieved by M-VIVIE in associating different identities extracted from different clips, i.e. in recognizing the same person in more clips, extracted from a same video or from different videos. Table 2 reports the mean error rates for the two sets *ClipA* and *ClipB*. As expected, due to lower variations in pose and expression, processing of set *ClipA* achieves better accuracy.

Due to the lack of public repositories of surveillance videos, to use for application benchmarking, we created the SURVEILLANCE set of clips according to the aims of this work. In particular, similarly to the experiments on GENERIC set, we aimed at assessing the performances of M-VIVIE with respect to two crucial aspects:

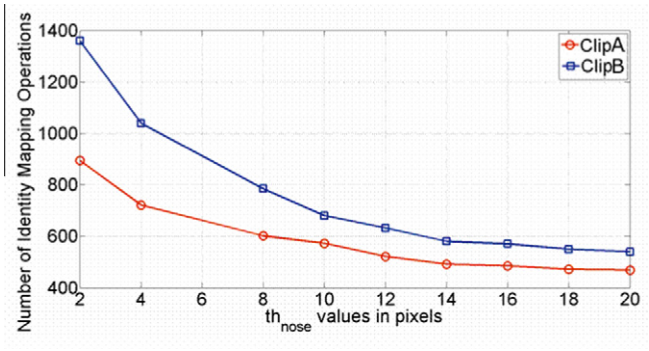

Fig. 6. Plot of the number of *identity mapping* operations versus variations of threshold th_{nose}.

Table 1
Number of permanent identities detected versus the number of identities actually present in each clip.

Properties	ClipA (TV news)								ClipB (movies)							
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Duration (min)	7	8	5	7	7	8	10	7	10	8	11	7	8	10	10	9
Target identities	16	14	8	11	8	10	13	10	4	6	3	5	5	4	5	4
Permanent identities	19	16	11	12	10	12	16	13	6	10	3	8	11	7	6	6

Table 2
Error rate (FAR and FRR) made by M-VIVIE in associating permanent identities of a same subject extracted from different clips.

Clip	FAR	FRR
ClipA	0.21	0.26
ClipB	0.31	0.38

(a) the ability to identify permanent identities within a video, (b) the accuracy of temporary to permanent identity mapping. For the first case, we considered seven videos, all captured by ACTI devices of different models (ICM 1011, ACM 3401, ACM 4001, ACM 5611, ACM 7411). Five out of these seven videos are different and reproduce indoor scenes (office) or outdoor ones (the entrance of a building); one of these five videos (ACM 7411) was processed using two further resolutions, different from its original one. For each such video, we evaluated the number of permanent identities that were detected by M-VIVIE, compared with the number of actual identities in the video. We also checked how many of the detected permanent identities exactly corresponded to the real identities in the video. As a matter of fact, we might have k real different identities in a video, and M-VIVIE might detect k permanent identities as well, but errors might still be made. Some real identity might not be detected, and other ones might be erroneously split in more permanent identities due to location or mapping errors. Moreover, it might also happen that the face detection module identifies as faces, in more consecutive frames, some structures that can be compared to a face shape, but actually have a completely different origin (folds on clothes, groups of small objects in a particular setting). The reason for a more accurate consideration of such further parameter in assessing the system in video-surveillance settings, is the higher criticality of related applications. In these cases errors in exactly determining the number of subjects in a protected space may cause false alarms, or on the contrary a missing alarm. The results of this first experiment are presented in the first four rows of Table 3.

The first row of Table 3 shows that M-VIVIE identified seven permanent identities, three of which were correct. In two cases, permanent identities of two different subjects were not merged, due to poor quality of the face samples, while two permanent identities were due to a false face detection by the related module. Even for video ACTI ACM 4001 two identities were due to face detection errors, while two different identities were erroneously fused. Incorrect identity fusion can be avoided by increasing the similarity threshold value for merging operations, if the system security level requires it. As for the last video, a subject does never look at the camera, so that it was not detected, while the face detection module identified a fictitious face. The two identities that are within actual device reach were correctly detected. The last three rows of Table 3 present the results obtained for a same video, with different resolutions, in order to evaluate how frame resolution can affect M-VIVIE system accuracy.

The first two rows of Table 3 related to such experiment show that one out of the eight present identities in the video was not detected due to the distance from the device (the same happens at lower resolutions), while seven were correctly detected and tracked. As resolution decreased, some identities were not detected due to the reduced size of the faces that were farther from the camera.

The accuracy of identity mapping operations was measured on a subset of SCface image dataset (Grgic et al., 2009). All 130 subjects in the database have been captured at three different distances, with eight devices (cam1,cam2,...,cam8), five of which in visible daylight, two during night in visible light, and one in infrared. In this experiment we only considered visible daylight images from the first four devices (cam1–cam4). For each subject we created two temporary identities. One of them was associated with the six samples from the first two devices (cam1,cam2), while the other one was associated with the samples coming from the remaining two devices (cam3,cam4). For each temporary identity in the first group, we tried a merging with all temporary identities in the second group. Fig. 7 presents the results of these experiments.

Each trial can be considered as a verification operation. Each merging operation that was successfully performed and involved the two temporary identities belonging to the same subject, was considered as a correct match/acceptance. A merging operation that was marked as possible, but involved temporary identities belonging to two different subjects, increased the False Acceptance Rate (FAR). On the contrary if the merging of the two temporary identities of the same subject was marked as impossible, this error increased the False Rejection Rate (FRR). The plot in Fig. 7(a) shows that even with only six samples, from two different cameras, M-VIVIE was particularly accurate in only merging those temporary

Table 3
Permanent identities detected by M-VIVIE versus actual identities in the video at hand, for different videos.

Video	Parameters			
	Frame resolution	Target identities	Permanent identities	Correct identities
ACTI ICM 1011	640 × 480	3	7	3
ACTI ACM 3401	464 × 368	2	2	2
ACTI ACM 4001	640 × 480	5	6	4
ACTI ACM 5611	900 × 720	3	3	2
ACTI ACM 7411	900 × 720	8	8	7
ACTI ACM 7411	640 × 480	8	7	7
ACTI ACM 7411	464 × 368	8	6	6

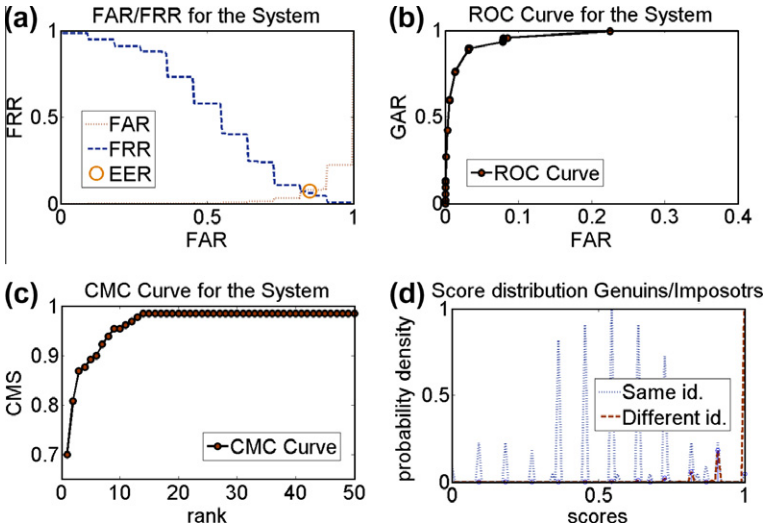

Fig. 7. Performances of M-VIVIE identity reduction process on a subset of SCFace: (a) FAR/FRR plot, with EER (0.073), (b) Receiving Operating Curve (ROC), (c) Cumulative Matching Curve, and (d) probability density of the scores when trying a merge between the same temporary identity or different identities.

identities that belong to the same subject. As a matter of fact, an Equal Error Rate (EER) of 0.073 in such conditions is very low, as it is also confirmed by the related Receiving Operating Curve (ROC) in Fig. 7(b).

If we consider the similarity achieved during each merging trial, we can also use the Figures of Merit (FOM) of idesntification operations. Fig. 7(c) shows that the probability of recovering the correct identity as the most similar in the second group, given a temporary identity in the first group, was above 70% and rapidly grew with the rank. Finally, Fig. 7(d) reports the similarity degree obtained when we tried to merge two identities of the same subject (Same id), compared with that obtained when M-VIVIE tried to merge identities belonging to different subjects (Different id). We can observe that the two distributions are well separated, with the one showing a mean of about 0.5 and the second a mean around 1.0.

7. Conclusions

In this work we presented M-VIVIE system for video sequence indexing based on the identity of appearing subjects. We described the overall system architecture, and detailed the features of the single component modules. Moreover, we presented the strategies which we followed to parallelize the main processing steps. The system presents a number of peculiarities that distinguish it from existing ones. The main one is the handling of dependencies among identities, i.e. the identification of identities which cannot be mapped onto each other. M-VIVIE was tested on 16 generic video clips, as well as on clips resembling video-surveillance settings, with results that fully confirm both its efficacy and efficiency. Thanks to the use of two different sets of videos, experiments provide an analysis of the system behaviour on video-sequences with very different features. We identified the optimal values for a number of parameters and demonstrated their consistency over the full set of clips. A further enhancement factor for the system might be the definition of suitable heuristics to identify and discard wrong permanent identities, and the creation of a suitable interface to browse the information provided by M-VIVIE.

References

Abate, A.F., Nappi, M., Riccio, D., Sabatino, G., 2007. 2D and 3D face recognition: A survey. Pattern Recognition Lett. 28 (14), 1885–1906.

Becker, B., Ortiz, E., 2009. Evaluation of face recognition techniques for application to facebook. In: IEEE Internat. Conf. on Automatic Face & Gesture Recognition, December 2009.

Blanz, V., Vetter, T., 2003. Face recognition based on fitting a 3D morphable model. IEEE Trans. PAMI 25 (9), 1063–1074.

Choi, J.Y., De Neve, W., Ro, Y.M., 2010. towards an automatic face indexing system for actor-based video services in an IPTV environment. IEEE Trans. Consumer Electronics 56 (1), 147–155.

De Marsico, M., Nappi, M., Riccio, D., 2010. FACE: Face Analysis for Commercial Entities. In: Proc. ICIP, 2010 Internat. Conf. on Image Processing, September 26–29, 2010, Honk Kong, pp. 1597–1600.

Grgic, M., Delac, K., Grgic, S., 2009. SCface-surveillance cameras face database. Multimed. Tools Appl. J. 51 (3), 863–879.

Milborrow, S., Nicolls, F., 2008. Locating facial features with an extended active shape model. In: European Conf. Computer Vision, pp. 504–513.

Torres, L., Vilà, J., 2002. Automatic face recognition for video indexing applications. Pattern Recognition 35 (3), 615–625.

Viola, P., Jones, M., 2004. Robust real-time face detection. Internat. J. Computer Vision 57 (2), 137–154.

Wang, H., Li, S.Z., Wang, Y., Zhang, J., 2004. Self quotient image for face recognition. In: Internat. Conf. on Image Processing, pp. 1397–1400.