Content-Based Copy Retrieval Using Distortion-Based Probabilistic Similarity Search

Alexis Joly, Olivier Buisson, and Carl Frélicot

***Abstract*—Content-based copy retrieval (CBCR) aims at retrieving in a database all the modified versions or the previous versions of a given candidate object. In this paper, we present a copy-retrieval scheme based on local features that can deal with very large databases both in terms of quality and speed. We first propose a new approximate similarity search technique in which the probabilistic selection of the feature space regions is not based on the distribution in the database but on the distribution of the features distortion. Since our CBCR framework is based on local features, the approximation can be strong and reduce drastically the amount of data to explore. Furthermore, we show how the discrimination of the global retrieval can be enhanced during its post-processing step, by considering only the geometrically consistent matches. This framework is applied to robust video copy retrieval and extensive experiments are presented to study the interactions between the approximate search and the retrieval efficiency. Largest used database contains more than 1 billion local features corresponding to 30 000 h of video.**

I. Introduction

THE principle of content-based copy retrieval (CBCR) is close to the usual content-based image or video retrieval schemes (CBIR) when using the query by example paradigm [1]–[3]. One difference is that the queries are not examples given by a user, but a stream of candidate documents automatically extracted from a particular medium (for example a television stream or a web downloader). The other and main difference is that the objects in demand are not the same. While general CBIR methods try to bridge the semantic gap, CBCR aims at recognizing a given document.

Content-based retrieval methods dedicated to copy detection have emerged in recent years for monitoring and copyright protection issues [4]–[8]. In this context, contrary to the watermarking approach, the identification of a document is not based on previously inserted marks but on content-based extracted signatures. These signatures are searched in an indexed database containing the signatures of all source documents and the retrieval can be performed without accessing the original documents. One of the main advantages of the content-based approach is that copies of already existing materials can be detected even if the original document was not marked or is no more existing.

Copyright and monitoring issues are, however, not the only motivation for CBCR and very promising applications are currently emerging such as database purge, cross-modal divergence detection or content-based web links creation. Automatic annotating is also a major prospect: all the versions of a given document correspond indeed to several utilization contexts and the links between them are highly informative. This could be easily used for semantic descriptions as illustrated by the two following television scenarios.

- A video which is broadcast the same day on several foreign channels refers to an international event.
- A video already having a lot of copies distributed among a large period, refers to a major historical event.

More generally, a CBCR system is able to generate a lot of contextual links (frequency, persistence, geographic dispersion, etc.) that could be exploited by data mining methods.

Section I-A details the CBCR issue and gives basic definitions, while previous work related to CBCR are discussed in Section I-B. In the rest of the paper, we describe a complete CBCR framework based on local features and applied to television monitoring (Section II). The main originality of this work is a new approximate similarity search technique that takes into account the special nature of a copy. The corresponding search algorithm in a multidimensional indexing structure is described and appears to be partially sublinear in database size. The other key point of our work is the study of the interactions between the retrieval and the computational performances. Intensive experiments on large and realistic databases are reported in Section III. They show how the different stages of our framework enable to fullfill both the high discrimination and speed requirements of CBCR.

A. Background

Duplicates and near-duplicates retrieval methods have emerged in recent years for a variety of applications, such as consumer photograph collections organization [9]–[12], multimedia linking [13], copyright infringement detection [4], [5], [7], [8], [14] or forged image detection [6]. In this paper, we focus on the content-based copy retrieval problem which generally consists of monitoring a specific medium to retrieve copies in a large reference database. A copy of an original document is not systematically an exact-duplicate but in most cases a transformed version of the original document. According to the

Manuscript received June 6, 2005; revised August 2, 2006. The associate editor coordinating the review of this paper and approving it for publication was Dr. Chitra Dorai.

A. Joly is with the INRIA Rocquencourt, 78153 Le Chesnay, France (e-mail: alexis.joly@inria.fr).

O. Buisson is with the INA, 94366 Bry-sur-Marne, France (e-mail: obuisson@ina.fr).

C. Frélicot is with the University of La Rochelle, 17042 La Rochelle, France (e-mail: carl.frelicot@univ-lr.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2006.886278

Fig. 1. Two copies (right) and their originals (left).

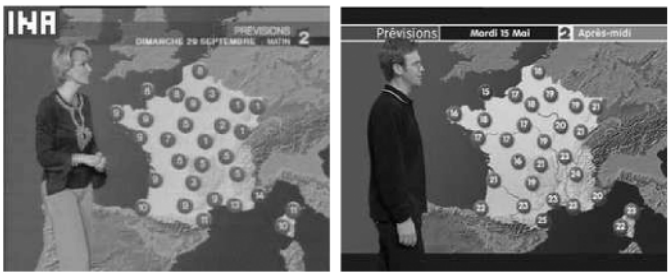

Fig. 2. Although these two images represent two different scenes, they can be considered as two copies of the France map displayed in the background.

literature definitions [9], [13], a copy can be see as a duplicate for which the capturing conditions can not differ (camera view angle, scene lighting conditions, camera parameters, etc.). Two copies are somehow derived from the same original document. Two documents and their copy are displayed on Fig. 1. The first copy is a typical televisual post-production example and the applied transformations are mainly resizing and a frame addition. In the second example, the copy was obtained by a poor kinescope (a film made of a live television broadcast). Fig. 2 presents a more ambiguous case since the documents represent two different scenes and none of them is a copy of the other one. However, both of them could be considered as copies of the France map displayed in the background. On the other hand, although they are similar in a sense, none of the two images displayed on Fig. 3 is a copy of the other one and they do not have a common original document. Note that, in the end, two documents that are not copies could be visually more similar than a copy obtained after a strong image processing. In practice, however, whatever the application is (copyright protection, multimedia linking, etc.), there is a subjective limit to the tolerated transformations and what is meant by *copy* remains a document that is visually similar.

In the end, a copy is a transformed version of a document that remains recognizable. We propose a definition of what a copy is, based on the notion of tolerated transformations.

Fig. 3. Two different scenes of the same video clip that are not copies.

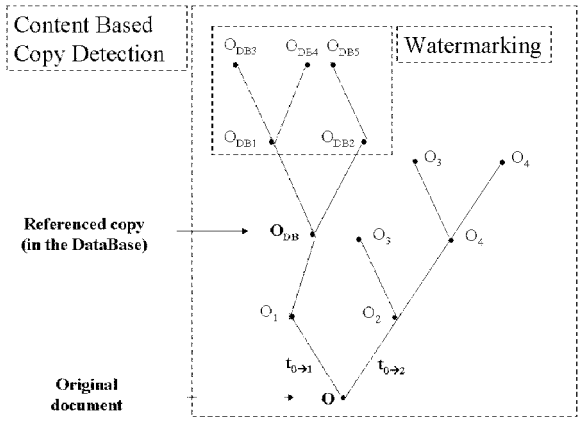

Fig. 4. Tree of all copies of an original document O.

Definition Copy: A document O_1 is a ***copy*** of a document O, if $O_1 = t(O), t \in T$, where T is a set of tolerated transformations. O is called the ***original document***.

Note that T can contain combinations of several transformations and a document $O_3 = t_3 \circ t_2 \circ t_1(O), t_1 \in T, t_2 \in T, t_3 \in T$ can be a copy of O as well as the intermediate documents $O_1 = t_1(O)$ and $O_2 = t_2 \circ t_1(O)$. Given an original document O, it is possible to construct a tree of all its copies as illustrated on Fig. 4. Note that, although all the documents of the tree are copies of the same original document O, they are not necessarily copies of each other, if we strictly respect the formal definition.

The general term *copy retrieval* we use in this paper refers to any document of the tree (all the copies of the original document(s) from which the query is derived from). We notice in Fig. 4 that the use of watermarking for the detection of copyright infringement only allows the detection of the copies of the referenced object. The content-based approach enables the detection of all the copies of the original object.

B. Related Work

The works related to CBCR are not necessarily dedicated to copy retrieval. In a sense, all CBIR systems using the query-by-example paradigm are potentially applicable to copy retrieval. It would be too long to list all of the CBIR works and we let the reader refer to more complete reviews [1]–[3]. However, usual CBIR schemes are generally neither fast enough nor discriminant enough. Most of them are based on color, texture and shape global features that are relevant in a generalization context but not in a copy recognition context

which requires more discrimination. The first system entirely dedicated to image copy retrieval is the RIME system [7], in which the extracted features are Daubechies wavelets coefficients. Unfortunately, no efficient similarity search strategy is proposed and the tolerated transformations are very light, excluding cropping, shifting, or compositing. Several works are dedicated to the search of duplicates or near-duplicates in consumer photograph databases [9]–[13]. Two different scenarios can be distinguished: duplicate retrieval and duplicate detection [13]. Duplicate retrieval aims at finding all images that are duplicate to an input query image, internal or external to the source image database. Duplicate detection aims at finding all duplicate image pairs given all possible pairs from the source image database. The second scenario is a more challenging task since the number of possible pairs increases in a quadratic speed [12]. The first scenario is very close to the copy retrieval problem. However, in practice, most of the proposed methods use direct and complex image to image comparisons whereas we are interested in finding documents only thanks to the similarity search of their signature(s) (and eventually the post-processing of associated data). Jaimes *et al.* [11], for example, use block-based correlation computed after global alignment of image pairs. In [13], Zhang proposed a near-duplicate detection by stochastic attributed relational graph matching. Using such image-to-image comparisons provide very good results in terms of robustness and discrimination but prevent the use of large databases. From a storage point of view, the monitoring of a specific medium can not be processed faced to the signature database only. And from the computational point of view, efficient similarity query algorithm [15] cannot be used to speed up the search.

In [16], Yuan give a complete list of all *query-by-video-clip* methods developed in the last decade. They classify them in three main categories depending on their ability to detect high-level similarity [17], [18], copies [5], [19] or near-exact copies [8], [20]–[23]. The idea behind the last category is that the low robustness requirements enable the use of very high compression rates and no multidimensional indexing structure is needed. The typical application of such methods is the detection of commercials or sponsoring and they are not applicable to more general copy retrieval applications for which the transformations are stronger. Furthermore, the database size is usually much smaller. Some of them are even dedicated to the detection of replicated sequences in a video stream and the problem is quite different from retrieving copies in a large static reference set. In [19], Cheung *et al.* propose a randomized algorithm called *ViSig*, aimed at measuring the fraction of visually similar frames shared between two sequences. The similarity search is accelerated by a new similarity search technique combining triangle inequality based pruning and a classical principal component analysis approach to reduce the dimension of the feature vectors. The technique enables fast and robust video copy retrieval on the world-wide-web with a large database including more than 1000 h of video.

Recently, Meng *et al.* [10] have used multiscale color and texture features to characterize images and employ the dynamic partial function (DPF) to measure the perceptual similarity of images [10]. Although the DPF outperforms traditional distance metrics, the global image feature they use limits the resistance to cropping, shifting or compositing. Furthermore, the DPF is not a metric and prevents the use of most similarity search techniques [15].

Contrary to the approaches based on blocks [8], local features based approaches [4]–[6], [12]–[14] give the best results in terms of robustness to cropping, shifting or compositing. In [14], the authors propose a mesh-based image copy retrieval method which is robust to severe image transformations. Computational performances are, however, not tackled in their study. Copy retrieval based on interest points and indexed local signatures has been proposed in [4] for still images, and in our previous work for video [5]. Such approaches enable to deal with both robustness and speed even with very large databases. The computationally critical step consisting of finding similar local features can be speed up by using multidimensional indexing structures and efficient similarity query algorithm. The final similarity measure between two documents is postponed to the post- processing of the partial results which can eventually contain associated data such as interest points position, orientation, or characteristic scale [12], [24]. Recently, this approach has been used again for image copy retrieval in [6], but with more recent distinctive points detector and local descriptors [24]. The experiments reported in this work show again that this strategy outperforms the others in terms of discrimination, robustness, and speed.

II. Content-Based Copy Retrieval Framework

An overview of the proposed CBCR framework is given in Fig. 5. Although it is dedicated to video copy retrieval, it can be easily adapted to still images. The retrieval stage itself includes three main steps that will be discussed in this section: a local features extraction (Section II-A), a new approximate similarity search technique (Section II-B), and a post-processing step based on a registration algorithm and a vote (Section II-C). The global principle of the retrieval can be summarized as follows: once the local features have been extracted from a candidate video sequence, they are individually searched in the database via the probabilistic similarity search technique. The search of each local feature provides a partial result consisting in a set of similar local features (called *neighbors*). The partial results obtained for all the candidate local features are then merged by the *post-processing* which consists of counting the number of geometrically-consistent local matches between the candidate sequence and the retrieved sequences (i.e.,the reference video sequences having at least one of their local features represented in the partial results).

A. Local Features

The local features used in our video CBCR framework are those described in [5]. They are based on an improved version of the Harris interest point detector [25] and a differential description of the local region around each interest point. To increase the compression, the features are not extracted in every frame of the video but only in key-frames corresponding to extrema of the global intensity of motion [26]. The final local features are 20-dimensional (20-D) vectors in $[0, 255]^{D=20}$ and the mean rate is about 17 local features per second of video (1000

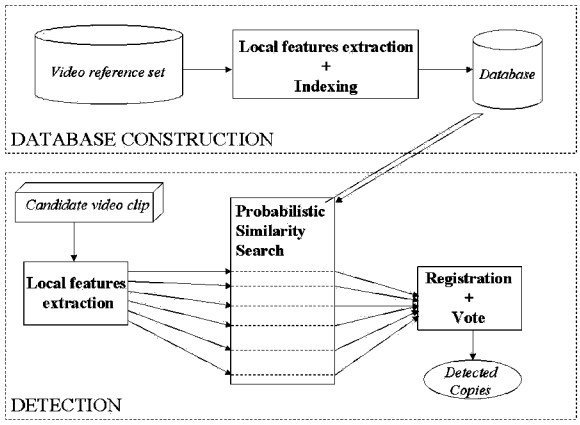

Fig. 5. Overview of the proposed framework.

h of video are represented by about 60 millions features). Let $\boldsymbol{S}$ be one of the local features, defined as

$$\boldsymbol{S} = \left(\frac{\boldsymbol{s^1}}{\|\boldsymbol{s^1}\|}, \frac{\boldsymbol{s^2}}{\|\boldsymbol{s^2}\|}, \frac{\boldsymbol{s^3}}{\|\boldsymbol{s^3}\|}, \frac{\boldsymbol{s^4}}{\|\boldsymbol{s^4}\|} \right)$$

where the $\boldsymbol{s^i}$ are five-dimensional (5-D) subvectors computed at four different spatio-temporal positions distributed around the interest point. Each $\boldsymbol{s^i}$ is the differential decomposition of the gray-level two-dimensional (2-D) signal $\boldsymbol{I}(x, y)$ up to the second order

$$\boldsymbol{s^i} = \left(\frac{\partial \boldsymbol{I}}{\partial x}, \frac{\partial \boldsymbol{I}}{\partial y}, \frac{\partial^2 \boldsymbol{I}}{\partial x \partial y}, \frac{\partial^2 \boldsymbol{I}}{\partial x^2}, \frac{\partial^2 \boldsymbol{I}}{\partial y^2} \right).$$

B. Distortion-Based Probabilistic Similarity Search (DPS^2)

Once a local feature $\boldsymbol{S}$ has been extracted from the candidate video sequence, the similar local features are searched in the database via the DPS^2 technique. After discussing the previous work related to the similarity search issue (Section II-B), we will present the DPS^2 paradigm (Section II-B2) and then briefly describe the indexing structure and the search algorithm we have developed (Section II-B3).

1) Related Work: Efficient similarity search in large databases is an important issue in all content-based retrieval schemes. In its essence, the *similarity query* paradigm [15] is to find similar documents by searching similar features in a database. A distance between the features is generally used to perform K*-nearest neighbors queries* or ϵ*-range queries* in the features database.

To solve this problem, multidimensional index structures such as R-tree family techniques [27]–[29] have been proposed, but their performances are known to degrade seriously when the dimension increases [30]. To overcome this *dimensionality curse*, other index structures have been proposed, e.g., the pyramid tree [31] or dimension-reduction techniques [32]. Sometimes, a simple sequential scan or other sequential techniques such as the VA-file are even more useful than all other structures [30]. However, the search time remains too high for many of the emerging multimedia applications and especially those using local features [33]. During the last few years, researchers have been interested in trading quality for time and the paradigm of *approximate similarity search* has emerged [34]–[42]. The principle is to speed up the search by returning only an approximation of the exact query results, according to an accuracy measure. Some of the first proposed approximate solutions are simply extensions of exact methods to the search of $\epsilon - KNN$ [35]–[37]; a $\epsilon - KNN$ being an object whose distance to the query is lower than $(1 + \epsilon)$ times the distance of the true kth nearest neighbor. ϵ represents the maximal relative error between the true nearest neighbors and the retrieved neighbors. In [37], Weber *et al.* propose an approximate version of the VA-file which is about five times faster than the exact version when 20% of the exact KNN are lost. The main drawback of the VA-file is that it is strictly linear in database size and that it is profitable only for a disk storage. In [35], Zezula *et al.* deal with $\epsilon - KNN$ in a M-tree, a convenient indexing structure for general metric distance measures. The performance gain is around 20 for a recall of 50% compared to exact results. However, they remarked that the real relative error was in practice seriously lower than the theoretical value ϵ. The accuracy of the search is therefore not controlled. To solve this, Ciaccia *et al.* [36] propose another M-tree based approximate method whose accuracy is enhanced thanks to an analyse of the distances distribution between the objects of the database. However, the a ssumption that the global distribution is a good estimator for the local query-to-element distribution presupposes that the distributions from all queries are similar, which is often not the case.

Clustering-based approximate methods have also been proposed to achieve substantial speed-ups over sequential scan [38], [40], [43]. As a pre-processing step, these algorithms partition the data into clusters. To handle a query, the clusters are ranked according to their similarity with the query vector and only the more relevant clusters are visited. These methods are efficient only if the data are well partitionable into clusters and their performances are therefore highly dependent of the data distribution. Another common drawback is that the clusters preprocessing are time consuming algorithms which can be prohibitive for very large databases. The approach of Ferhatosmanoglu *et al.* uses the well-known K-means heuristic to generate a large number of small clusters. The queries are handled by an original iterative selection of the clusters based only on the first few coordinates of the vector representation. It can potentially speed-up the query processing by an order of magnitude or more. The main drawback of the K-means algorithm is that it produces poor-quality clusters degrading the efficiency of the data filtering. The CLINDEX method, proposed by Li *et al.* [38], is based on a very different clustering scheme. The partition is processed via an efficient bottom-up cluster-growing technique, relative to a grid partition of the domain. The search algorithm makes use of a fast index structure over the set of clusters and achieves speedups of roughly 21 times over sequential search at the 70% recall level. The accuracy of the search is however not controlled since the stopping criterion is only the number of visited clusters. At the opposite, the method of Berrani *et al.* [40] allows an accurate control of the query results thanks to a probabilistic criterion preprocessed for each cluster. The adaptation of the BIRCH algorithm [44] to form the clusters also provides a better

clustering in an acceptable computation time. A comparison with CLINDEX shows that this technique is 10–70 times faster than CLINDEX. The search time is however linear in database size which prevents the use of extremely large databases. The method of Benett *et al.* [39] also use a probabilistic criterion to select the more relevant clusters but it is directly issued from the clustering process: A mixture of Gaussian distributions is estimated from the dataset thanks to the Expectation Maximisation algorithm and this probability density function is used to assess the probability that a given cluster contains a closer point than the neighbors already found. The main drawback is that the maximum number of clusters is strongly limited. The method is therefore limited to very specific distributions.

Some approximate methods are based on a binary representation of the vectors and the use of a simple Hamming distance to compare them [21], [34], [45]. Kalker *et al.* proposed the use of an inverted list where the entries are sub-vectors of the complete binary vectors. A neighbor can then be retrieved only if one if its sub-vector remains unchanged which is a strong hypothesis. The technique is therefore very fast but the quality of the results is very poor and not controlled. In the binary scheme proposed by Miller *et al.* [45], the search in a binary tree is guided by estimating bit errors probabilities. This allows to have a control of the query results accuracy but only according to the Hamming distance. Furthermore, the bit errors are supposed to be uniformly distributed along the binary vectors which is a strong assumption. Indyk *et al.* [34] developed a randomized *locally-sensitive hashing* method for vector data. Contrary to other binary schemes, the unary representation of the vectors, used to form the binary vectors, makes the Hamming distance in the binary transformed space equivalent to the L_1 distance in the original space. A set of hash functions is applied to the binary vectors and the similarity search issue is translated in terms of collision probability. The main advantages of the technique is that it is sub-linear in database size and that it tolerates high dimensions. However, like other binary schemes, the quality of the query results is poor [42] and their accuracy is not controlled.

In [42], Houle *et al.* developed a practical index called $SASH$ for approximate similarity queries in extremely high-dimensional data without any assumption regarding the representation of the data. $SASH$ is a multilevel structure of random samples connected to some of their neighbors. Queries are processed by first locating approximate neighbors within the sample, and then using the pre-established connections to discover neighbors within the remainder of the data set. The technique can return a large proportion of the true neighbors roughly two orders of magnitude faster than sequential scan for moderate dimensions and less than one order of magnitude for extremely high dimensions. Main drawbacks are the cost of the pre-processing step and the absence of query accuracy control.

In the following subsection, we present our new similarity search technique. Contrary to the previous methods, it is not based on the approximation of an exact geometric query but directly on a distortion-based probabilistic query in the multidimensional vector space. The technique has two main advantages.

- Searching most probable signatures instead of signatures respecting a geometric criterion is less restricting and more relevant. Furthermore, no explicit metric is needed to select the relevant domain regions.
- As the queries are supposed to be entirely independent of the database distribution, the selection of the domain regions we need to visit can be determined without accessing the database. This makes the method easily distributable and very quick since no tree is needed.

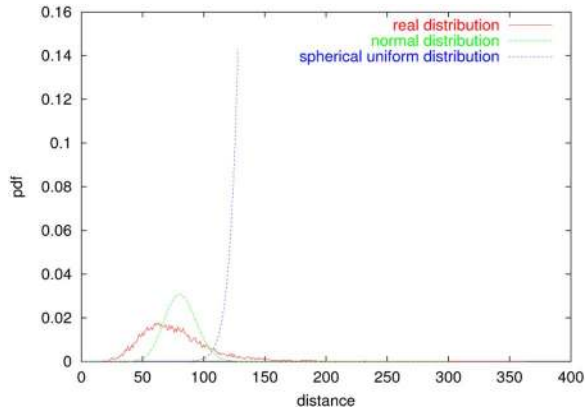

Fig. 6. Distribution of the distance between a feature and a distorted feature after transformation of a video sequence (small resizing, noise addition, gamma, and contrast modification).

2) Distortion-Based Probabilistic Queries: Our distortion-based probabilistic queries can be introduced in terms of approximate range queries. Intuitively, the principle of an approximate range query would be the following: by excluding several regions of the query, having a too small intersection with the bounding regions of the index structure, it is possible to speed up the search without significantly degrading the results. However, it is not possible to directly take the volume as an error measure because it would be equivalent to consider that the relevant similar features are uniformly distributed inside the range query. When the dimension increases, the features following such a distribution become closer and closer to the surface of the hyper-sphere and this is not true in reality, as illustrated in Fig. 6. The solid curve (left) is the real distribution of the distance between referenced and distorted features obtained by a representative combination of image transformations. The two other dotted curves represent the estimated probability density function for two probabilistic models: an uniform spherical distribution (right) which would be obtained if we took the volume as an error measure and a zero mean normal distribution under the independence assumption (center). The figure shows that the normal distribution is much closer to the real distribution than the uniform distribution.

The proposed *distortion-based probabilistic queries* rely on the distribution of the relevant similar features for finding a transformed document. Let the *distortion* vector ΔS be defined as

$$\boldsymbol{\Delta S} = \boldsymbol{S}(\boldsymbol{M}) - \boldsymbol{S}\left(t_M(\boldsymbol{M})\right)$$

where $\boldsymbol{S}(\boldsymbol{M})$ is the feature of an image local region M and $\boldsymbol{S}(t_M(\boldsymbol{M}))$ the distorted feature after transformation t_M of the

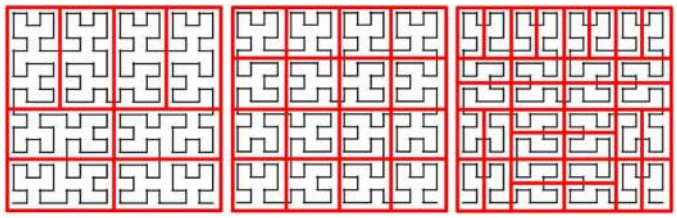

Fig. 7. Illustration of the space partition induced by the Hilbert space-filling curve at different depths $p = 3, 4, 5$ in two dimensions.

image. We define a *distortion-based probabilistic query*, associated to a probability equal to α, as the retrieval of all the database features contained in a region V_α of the feature space satisfying

$$\int_{V_\alpha} p_{\boldsymbol{\Delta S}}(\boldsymbol{X}-\boldsymbol{S})d\boldsymbol{X} \geq \alpha \tag{1}$$

where $\boldsymbol{S}$ is the query (i.e., the candidate feature) and $p_{\boldsymbol{\Delta}}\boldsymbol{S}(.)$ is the probability density function of the distortion. Intuitively, the probabilistic query selects only the regions of the feature space for which the probability of finding a distorted signature is high in order to reduce the number of signatures to scan during the search: usually, the first step of a search in a multidimensional indexing structure is a set of geometric filtering rules that quickly exclude most of the bounding regions of the index partition that do not intersect with the query [15]. To compute the probabilistic queries, we propose to replace the geometric rules by probabilistic rules, according to the distortion model. The main advantage of this strategy is that a probabilistic query has no intrinsic shape constraint. Thus, the region V_α can be chosen so that it minimizes the number of bounding regions that need to be explored.

3) Indexing Structure and Search Algorithm: The indexing structure we use to process our distortion-based probabilistic queries is described in [46]. It is a space-partition based and a static method (dynamic insertions or deletions are not possible). The partition is induced by the regular split of a Hilbert space-filling curve as illustrated in Fig. 7. It results in a set of 2^p non overlapping and hyper-rectangular bounding regions, called *p-blocks* [46], which are well-suited to compute quickly the integral of (1). The depth p of the partition is equal to the number of bits of the Hilbert derived keys used to access the data pages corresponding to each block.

The probabilistic search algorithm [46] is composed of two steps: a filtering step that selects the relevant p-blocks and a refinement step that exhaustively processes all the features belonging to the selected blocks. For computational efficiency, the probabilistic filtering step of our search algorithm relies on the assumption that the components of the distortion are independent:

$$p_{\boldsymbol{\Delta S}} = \prod_{j=1}^{D} p_{\Delta Sj}$$

The distortion distribution is simply modeled by a zero-mean normal distribution with the same standard deviation σ, whatever the component is

$$p_{\Delta Sj}(x) = f_{\mathcal{N}(0,\sigma)}(x). \tag{2}$$

The unique parameter σ of this isotropic distribution makes the probabilistic query more intuitive and closer to the approximate range query paradigm, σ replacing the usual radius. The probability of a p-block b can then be computed as

$$\int_b p_{\boldsymbol{\Delta S}}(\boldsymbol{X}-\boldsymbol{S})d\boldsymbol{X} = \prod_{j=1}^{D} \int_{u_j}^{v_j} f_{\mathcal{N}(0,\sigma_q)}(x_j - s_j)dx_j$$

where u_j and v_j are the lower and upper bounds of the p-block b along the j^{th} axis, s_j and x_j are the j^{th} component of respectively a candidate feature $\boldsymbol{S}$ and any vector $\boldsymbol{X}$ of the feature space.

For a p-depth partitioned space and a candidate feature $\boldsymbol{S}$, the probabilistic query inequality (1) may be satisfied by finding a set B_α of p-blocks, such as

$$\sum_{i=1}^{card(B_\alpha)} \int_{b^i} p_{\boldsymbol{\Delta S}}(\boldsymbol{X}-\boldsymbol{S})d\boldsymbol{X} \geq \alpha \quad b^i \subset B_\alpha, \forall i \tag{3}$$

where $card(B_\alpha) \leq 2^p$ is the number of blocks in B_α.

In practice, $card(B_\alpha)$ should be minimum to limit the cost of the search. We refer to this particular solution as B_α^{min}. Its computation is not trivial because sorting the 2^p blocks according to their probability is not affordable. Nevertheless, it is possible to quickly identify the set $B(\tau)$ containing all the blocks having a probability greater than a fixed threshold τ

$$B(\tau) = \left\{ \{b^i\} / \int_{b^i} p_{\boldsymbol{\Delta S}}(\boldsymbol{X}-\boldsymbol{S})d\boldsymbol{X} > \tau \right\}.$$

The total probability of $B(\tau)$ is given by:

$$P_\Sigma(\tau) = \sum_{i=1}^{card(B(\tau))} \int_{b^i} p_{\boldsymbol{\Delta S}}(\boldsymbol{X}-\boldsymbol{S})d\boldsymbol{X}$$

$B(\tau)$ and $P_\Sigma(\tau)$ are computed thanks to a simple hierarchical algorithm based on the iterative increase of the partition depth (from $p_1 = 1$ to $p_p = p$). At each iteration, only the blocks having a probability higher than α are kept in a priority queue. Since $card(B(\tau))$ decreases with τ, finding B_α^{min} is equivalent to finding τ_{min}, verifying

$$\begin{cases} P_\Sigma(\tau_{min}) \geq \alpha \\ \forall \tau > \tau_{min}, P_\Sigma(\tau_{min}) < \alpha \end{cases} \tag{4}$$

As $P_\Sigma(\tau)$ also decreases with τ, τ_{min} can be easily approximated by a method inspired by Newton-Raphson technique (the hierarchical algorithm is applied several times).

The refinement step of the DPS^2 technique computes the L^2 distance between the query and all the vectors belonging to the selected blocks. The final results can be selected either by a K-nearest neighbor search or by a range search. As explained before, we generally prefer a range search strategy and the radius r_σ is set in order to guarantee a final probability α_f higher than $\alpha - 0.1\%$ (see Appendix I). This range search allows to exclude a large part of the features selected by the probabilistic filtering step while preserving a final probability very close to α.

The partition depth p is of major importance since it directly influences the search time t_s of the DPS^2 technique

$$t_s(p) = t_f(p) + t_r(p)$$

The time of the filtering step $t_f(p)$ is strictly increasing with p because the number of p-blocks in B_α^{min} and thus the computation time increase with p. The refinement time $t_r(p)$ is decreasing because the *selectivity* of the filtering step increases, i.e the number of features belonging to the selected blocks decreases with p. The search time $t_s(p)$ has generally only one minimum at p_{min} which can be set at the start of the system in order to obtain the best average response time. In practice, p_{min} depends particularly on the database size and the storage support. Storing on disk makes the filtering step depending mainly on the disk access time whereas it depends mainly on CPU time when running in main memory. These costs differ from three orders of magnitude, whereas the costs of the refinement step differ only from one order of magnitude. Therefore, the optimal depth is lower for a disk storage (from about seven unity) and the search time is about 100 times slower.

4) Comparison to Exact Range Queries: Here, we do not aim at testing the relevance of the distorsion model, but only at showing the advantage of a statistical query compared to an exact range query when the distortion model [see (2)] is supposed to be exact. We randomly selected 1000 signatures in a real database and computed the filtering step of our similarity search technique for both a probabilistic query and an exact range query. The radius ϵ of the range query is set in order to have the same probability α than the probabilistic query. For varying values of α, we measured the average number of p-blocks intercepted by both queries (Fig. 8). The figure clearly show that the probabilistic query is widely more profitable than the exact range query: the same probability can be expected with 30 to 100 times fewer blocks to visit, depending on α. The default of the exact range query is its rigid shape which intercepts a lot of unlikely blocks. On the contrary, the probabilistic query selects the optimal set of blocks for the desired probability.

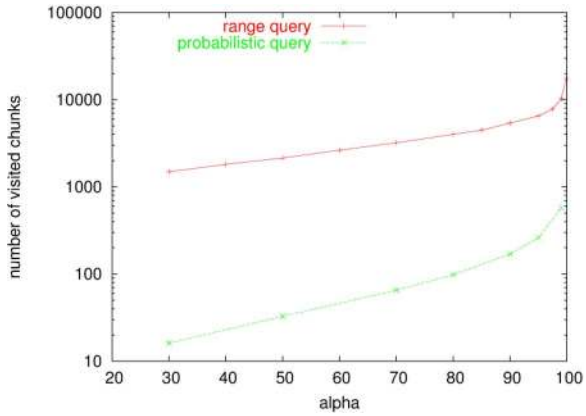

Fig. 8. Average number of blocks intercepted by a probabilistic query and a range query of same probability α.

C. Registration and Vote

Once the local features have been searched in the database, the partial results must be post-processed to compute a global similarity measure and to decide if the more similar documents are copies of the candidate document. Usually, this step is performed by a vote on the document identifier provided with each retrieved local feature [4], [25]. Thus, the similarity between the candidate document and the retrieved documents is measured by the number of local matches. This method is robust to strong geometric transformations since it does not care about the points relative position. However, as the geometry of the image is ignored, it can induce confusion between two images having similar local features but which are not copies from each other.

Another method consists in keeping, for each retrieved document, only the matches which are geometrically-consistent with a global transform model [6], [11], [47]–[49]. The principle is to use the associated points position of the retrieved local features to estimate the parameters of the model. The vote is then applied by counting only the matches that respect the model (registration + vote strategy). The choice of the model characterizes the tolerated transformations. In this paper, we consider only resize, rotation, and translation for the spatial transformations and also slow/accelerated motion from the temporal point of view

$$\begin{pmatrix} x' \\ y' \\ t'_c \end{pmatrix} = \begin{pmatrix} rcos\theta & -rsin\theta & 0 \\ rsin\theta & rcos\theta & 0 \\ 0 & 0 & a_t \end{pmatrix} \begin{pmatrix} x \\ y \\ t_c \end{pmatrix} + \begin{pmatrix} b_x \\ b_y \\ b_t \end{pmatrix} \quad (5)$$

where (x', y', t'_c) and (x, y, t_c) are the spatio-temporal coordinates of two matching points. A candidate video sequence is defined as a set of n_f successive key-frames (typically $n_f = 9$, i.e about 11 s of video), containing n_c local features that will be searched in the database. The similarity search technique provides for each of the n_c candidate local features a set of matches characterized by their spatio-temporal position and an identifier V_h defining the referenced video clip to which the feature belongs. All the matches of the candidate sequence are first sorted according to the value of their identifier and the transformation model parameters are estimated for each retrieved video clip V_h thanks to a random sample consensus (RANSAC [50]) algorithm. For each candidate local feature, only the best match in each retrieved video clip V_h is kept for the estimation (according to the L_2 distance between signatures). Once the transformation model has been estimated, the final similarity measure $m(V_h)$ related to a retrieved video clip V_h consists in counting the number of matching points that respects the model according to a small spatio-temporal precision. At the end, the similarity measure $m(V_h)$ is thresholded to decide whether the document V_h is a copy or not. m is lower or equal than n_c, the number of

local features in the candidate video sequence and its expected value $E(m)$ can be ideally expressed as:

$$E(m) = n_c R \alpha$$

where R is the repeatability of the interest points detector (rate of points that remain stable after image transformation) and α the query probability.

III. Experimental Evaluation

Section III-A describes the common experimental setup. Section III-B discusses the influence and the settings of the DPS^2 parameters. Section III-C presents a comparison between exact range queries and probabilistic queries. Section III-D aims at studying the influence of the database size on both the computational performances and the retrieval performances. Section III-E presents some experiments on a real ground truth and some results obtained from a TV monitoring.

A. Experimental Setup

The video sequences used to construct the signatures databases come from the so-called *SNC* database stored at the French *Institut National de l'Audiovisuel* (INA), whose main tasks include collecting and exploiting French television broadcasts. The *SNC* video sequences are stored in $MPEG1$ format with an image size of 352 × 288. They contain all kinds of TV broadcasts from the Forties to the present: news, sport, shows, variety, films, reports, black and white archives, advertisements, etc. They also contain noise, black sequences, and test cards which potentially degrade any experimental assessment. The databases used in the following experiments are randomly selected subparts of the SNC database and we note S_H a randomly selected subpart containing H h of video (the smallest used database is S_{100} containing 100 h of video and the largest S_{30000} containing 30 000 h of video). Experiments were carried out on a Pentium IV (CPU 2.5 GHz, cache size 512 kb, RAM 1.5 Gb) and the response times were obtained with unix `getrusage()` command. Recall, precision and false alarm probability metrics are defined as

$$\begin{aligned} &Recall\ r_c \\ &\quad = \frac{number\ of\ true\ positives}{total\ number\ of\ true} \\ &Precision\ p_r \\ &\quad = \frac{number\ of\ true\ positives}{total\ number\ of\ positives} \\ &False\ alarm\ probability\ p_{fa} \\ &\quad = \frac{number\ of\ false\ positives}{total\ number\ of\ false} \end{aligned} .$$

Except for Section III-E, where a specific ground-truth data is used, the experiments are based on five kinds of synthetic transformations illustrated in Fig. 9.

1) Resizing: resize factor w_{scale}.
2) Shifting: parameter w_{shift} in % of the image height.
3) Contrast modification: $I'(x,y) = I(x,y) \times w_{\text{contrast}}$.
4) Gamma modification: $I'(x,y) = 255(I(x,y)/25^{w_{gamma}}$.
5) Gaussian noise addition: standard deviation w_{noise}.

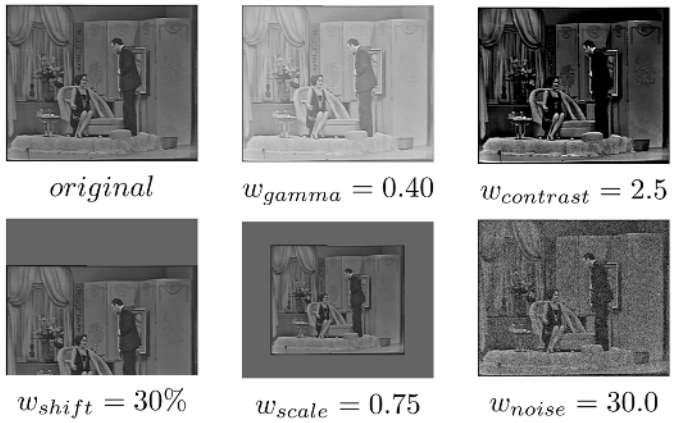

Fig. 9. Five kinds of transformations studied in the experiments: resize, shift, contrast, gamma, noise addition.

TABLE I
Range Values for the Randomly Selected Transformations

	n_T	w_{scale}	w_{shift}	$w_{contrast}$	w_{gamma}	w_{noise}
min	0	0.7	0%	0.4	0.4	0.0
max	5	1.5	35%	2.5	2.5	35.0

The default assessment methodology is as follows: 100 video clips of 15 s are randomly selected in the database S_H and then corrupted with a combination of previous transformations. This query video set is referred as Q_H. The number of transformations n_T and the transformation parameters are randomly selected **for each video clip**. The range values of each parameter are given in Table I. These 100 transformed video clips represent the true probes and a true positive occurs when the original video clip in S_H is well retrieved with a temporal precision of two images. The false probes come from a foreign TV channel capture that is supposed to never broadcast archives belonging to the french SNC database (the most confident matches were manually controlled to confirm this hypothesis). This query video set is referred as Q_F and it is ten times longer than Q_H, in order to have a more realistic precision measure. The total length of all the queries $(Q_H + Q_F)$ is 16 500 s (4 h 34 min).

B. DPS^2 Parameters Discussion

1) Estimation of σ: We applied four kind of transformations with increasing parameters value to 30 randomly selected video sequences of 1-min length (from S_{1000}). For each transformation, we measured the repeatability R of the Harris detector as the pourcentage of stable points with a tolerated error of 1 pixel. We also estimated the parameter σ of the distortion model (see (2)) by considering only the stable points

$$\overline{\sigma}^2 = \frac{1}{N_s}\frac{1}{D}\sum_{i=1}^{N_s}\sum_{j=1}^{D}\left(\Delta S_j^i\right)^2$$

where N_s is the number of stable points and ΔS_j^i is the j-th component of the distortion vector of the i-th stable point. The results are summarized in Table II. It shows that the value of $\overline{\sigma}$ increases with the severity of the transformations. On the other side the repeatability of the Harris detector decreases with the severity of the transformations. When this repeatability is to low, the retrieval will fail in all cases and it is useless to try to find signatures distorted by so severe transformations. The value of

TABLE II
HARRIS DETECTOR REPEATABILITY AND VALUE OF $\overline{\sigma}$ FOR FOUR TRANSFORMATIONS WITH VARIABLE PARAMETERS

w_{scale}	0.5	0.6	0.7	0.8	0.9	1.0	1.1	1.25	1.4	1.6	2.0
R	5%	14%	31%	62%	83%	100%	90%	77%	71%	**34%**	11%
$\overline{\sigma}$	37.4	31.74	24.20	15.85	9.22	0.0	8.15	13.20	20.82	**24.4**	33.9
w_{gamma}	0.4	0.61	0.82	1.03	1.24	1.45	1.66	1.87	2.08	2.29	2.50
R	76%	84%	92%	98%	92%	85%	77%	72%	70%	66%	62%
$\overline{\sigma}$	10.80	9.82	9.23	8.96	9.19	9.60	10.31	11.05	12.17	13.10	13.98
$w_{contrast}$	0.4	0.61	0.82	1.03	1.24	1.45	1.66	1.87	2.08	2.29	2.50
R	87%	95%	98%	100%	91%	82%	76%	69%	67%	63%	50%
$\overline{\sigma}$	9.26	9.15	9.13	8.99	9.64	10.97	12.74	15.17	17.26	18.92	20.18
w_{noise}	5.0	10.0	15.0	20.0	25.0	30.0	35.0	40.0			
R	92%	82%	78%	69%	64%	55%	49%	40%			
$\overline{\sigma}$	10.28	11.35	12.25	13.35	14.42	15.52	16.64	17.58			

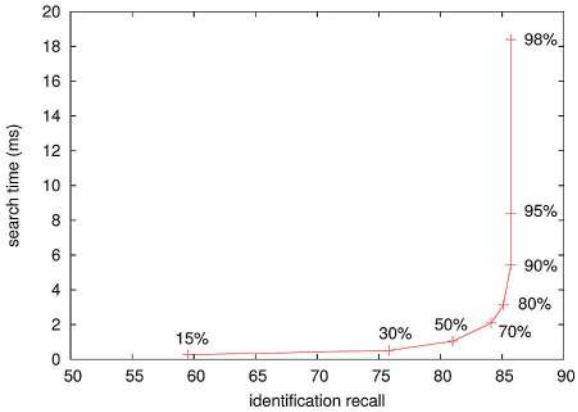

Fig. 10. Search time of the DPS^2 technique with respect to recognition recall, at constant precision ($p_r = 90\%$), for different query probabilities α

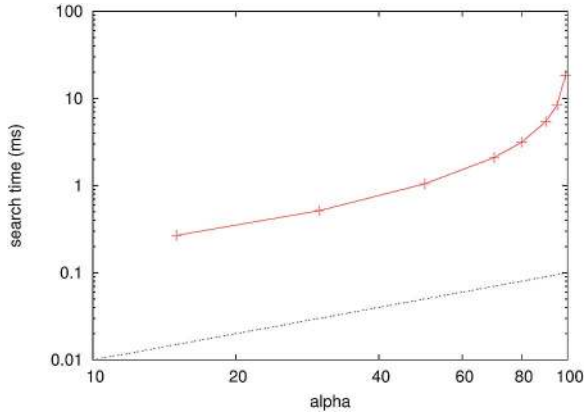

Fig. 11. Search time of the DPS^2 technique with respect to the query probability α.

$\overline{\sigma}$ in bold in the table corresponds to the maximum value for which the Harris detector repeatability is higher than 15% and we propose to use this criterion as a selection of σ. In all the following experiments the value of σ is set to

$$\sigma = 24.4$$

2) Influence of α: Fig. 10 represents the average search time t_s of one single local query with respect to the recall of the complete copy retrieval scheme (database S_{1000} and query set $Q_{1000} + Q_F$). Each point corresponds to a value of the query probability which varies from $\alpha = 15\%$ to $\alpha = 98\%$. The recall values were determined at constant precision $p_r = 90\%$ (i.e., a ROC curve has been built for each point). The curve show the relevance of the approximate search paradigm: the recall remains almost constant when the probability of the query decreases from $\alpha = 98\%$ to $\alpha = 70\%$, whereas the search is more than three times faster. For smaller values of the probability, the recall starts to degrade more significantly. It is, however, interesting to see that the recall is still 60% when only 15% of the signatures are expected to be retrieved.

Let us emphasize that the distortion-based probabilistic search paradigm is always more efficient than reducing the number of queries. In other words, searching a fixed rate of the candidate local features is always slower than searching all the features with the appropriate approximation (for the same quality). It is indeed easy to show (see Appendix II) that this proposition is true if

$$\frac{ln\left(t_s(\alpha_2)\right) - ln\left(t_s(\alpha_1)\right)}{ln(\alpha_2) - ln(\alpha_1)} < 1, \quad \forall(\alpha_1, \alpha_2) \qquad (6)$$

TABLE III
SEARCH TIME AT CONSTANT PRECISION ($p_r = 90\%$) AND RECALL ($r_c = 80\%$)

algorithm	time (ms)
probabilistic queries	9 min 35 sec
range queries	7 h 24 min 23 sec

where $t_s(.)$ is the average search time of one single feature and α_1 and α_2 two values of the query probability. The experimental curve $t_s(\alpha)$ is plotted in logarithmic coordinates on Fig. 11 and shows that (6) is always true, since the slope is always increasing and higher than one (A reference line with a slope equal to one has also been plotted on the figure). This proves the relevance of the approximate search paradigm for the DPS^2 technique.

In most of the following experiments, the value of α is set to $\alpha = 75\%$ in order to benefit from the approximate search paradigm without degrading the retrieval performances.

C. Comparison to Exact Range Queries

To validate the approximate search paradigm induced by the distortion-based probabilistic similarity search, we have compared it to an exact range query strategy. For this purpose, we developed a hierarchical algorithm using the same indexing structure as the probabilistic queries, but with geometric filtering rules. Fig. 12 represents the average search time of one single local feature with respect to the recall of the whole retrieval process (database S_{1000} and query set $Q_{1000} + Q_F$). The curve corresponding to the probabilistic queries was obtained with $\sigma = 24$ and increasing values of α. The curve corresponding to the range queries was obtained with increasing values of the

TABLE IV
TOTAL SEARCH TIME OF THE SIGNATURES EXTRACTED FROM THE 4 H 34 MIN OF CANDIDATE VIDEO MATERIALS

DB size (hours)	250	2,500	25,000
DB size	14,098,729	126,562,273	1,286,585,349
Search time DPS^2 in memory	4 min 32 s	22 min 24 s	2 h 48 min 42 s
Search time DPS^2 on disk	5 h 3 min 23 s	25 h 12 min 54 s	
Search time sequential scan in memory	38 h 36 min	346 h (interpolated)	3519 h (interpolated)
Search time sequential scan on disk	262 h (interpolated)	2352 h (interpolated)	23911 h (interpolated)

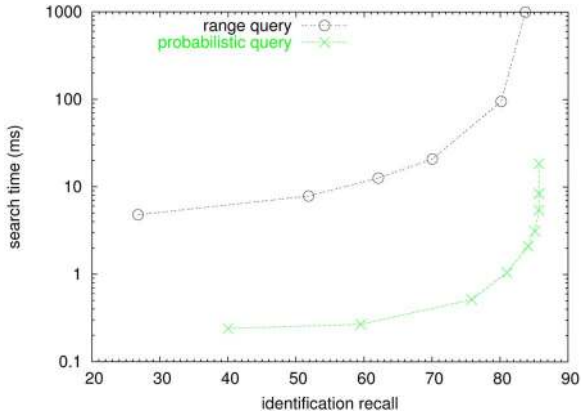

Fig. 12. Comparison of distortion-based probabilistic queries and exact range queries. Search time with respect to recognition recall, at constant precision ($p_r = 90\%$).

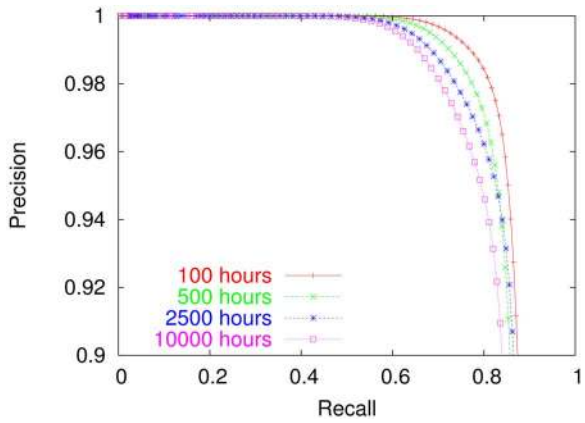

Fig. 14. ROC curves for different DB sizes.

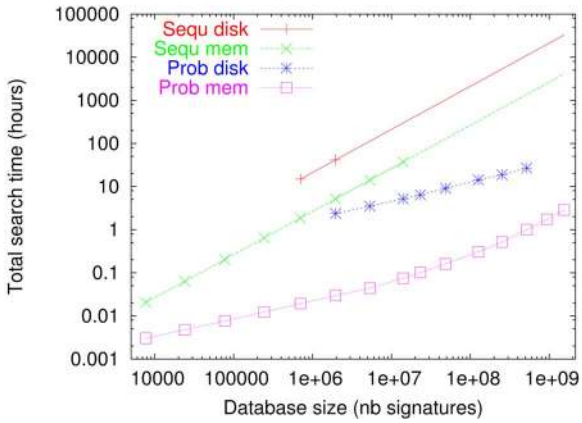

Fig. 13. Total search time with respect to databased size.

query radius. Recall values are measured at constant precision $p_r = 90\%$. Table III gives the total search time for all the queries (extracted from the 4 h 34 min of candidate video materials) at constant precision $p_r = 90\%$ and consta nt recall $r_c = 80\%$. These results clearly show that the approximate search paradigm is widely more advantageous than exact range queries. At identical recall and precision, the search is 30–500 times faster than exact range queries.

D. Influence of the Database Size

1) Computational Performances of the DPS^2 *Technique:* Fig. 13 and Table IV deal with the computational performances of the DPS^2 technique (databases S_H and query sets $Q_H + Q_F$). Probabilistic query parameters are set to $\alpha = 75\%$ and $\sigma = 24.4$. The DPS^2 technique is compared to a classical sequential scan which is a reference method. Each method has been implemented both on disk and in main memory. For the main memory case, a specific strategy is used when the database size exceeds the memory size.

1) The filtering step of the DPS^2 is first processed for all the queries ($Q_H + Q_F$). Notice that this step doesn't need to access the database.
2) The database is then split into pages that can fit in main memory.
3) The pages are loaded successively in main memory and the refinement step of all queries is processed for each page.

The search times refer to the search of all the signatures extracted from the 4 h 34 min contained in $Q_H + Q_F$. As shown on the figure, plotted in logarithmic coordinates, the search time of the DPS^2 is first sublinear in database size and then asymptotically linear for very large databases. The search time remains however very low even in the case of huge databases including more than 1.5 billion local features (30 000 h of video). Compared to a sequential scan, which is a reference method, the technique is asymptotically 2500 times faster.

2) Retrieval Performances: Figs. 14 and 15 deal with the quality of the retrieval when the database size is strongly increasing (databases S_H). Fig. 14 represents the precision/recall curves of the system for the default assessment methodology (randomly selected transformations) whereas Fig. 15 represents the recall at constant precision $p_r = 90\%$ for varying parameters of the five studied transformations. Probabilistic query parameters were set to $\alpha = 75\%$ and $\sigma = 24.4$.

Despite of the two orders of magnitude between the smallest and the largest database, the recall and the precision of the system do not significantly degrade. This robustness to database size is closely linked to the use of local features and to the discrimination of voting strategies. Even if the number of nonrelevant features provided by the search for each local

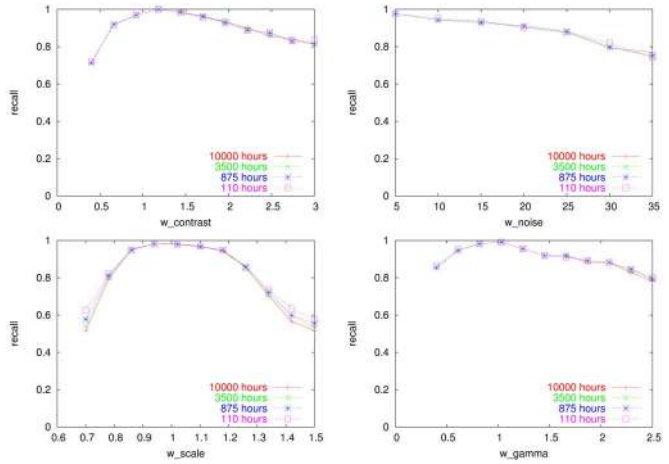

Fig. 15. Recall at constant precision $p_r = 90\%$ for several transformations and several database sizes.

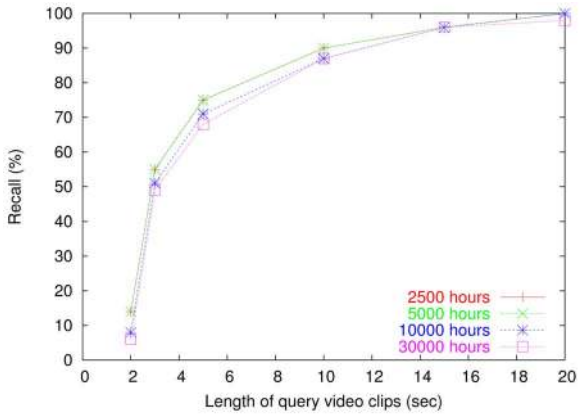

Fig. 16. Recall versus maximal length of.

feature is linear in database size, the impact on the final similarity measure is very limited because the number of consistent matches remains very low.

E. Real World Experiments

1) Ground Truth: A ground truth has been built from a 3-h suspicious program that was stored on a video tape. It is a variety program containing 23 archives video clip of various lengths ($min = 6\,\text{s}, max = 350\,\text{s}, average = 59\,\text{s}$), the rest of the program being set scenes. The 23 archives were manually retrieved in the INA database by a keyword search in a dedicated search engine and were manually registered to determine precisely the beginning and the end of each match. They were then inserted into four randomly selected databases (S_{2500}, S_{5000}, S_{10000}, S_{30000}). The transformations that could be visually identified include resizing ($\pm 15\%$), frames and texts addition and contrast enhancement with overload of several regions. The program was then entirely searched by our content-based copy retrieval system in each database. Fig. 16 displays the recall of these searches for various temporal resolution, i.e the step of the temporal splitting defining what are the researched objects. A temporal segment is considered to be retrieved if at least one correct detection occurs among all its key images (with a temporal precision of two frames). The curve shows that the retrieval is very efficient when the temporal granularity is higher than 5 s.

Fig. 17. Two good detections (up) and two false alarms (bottom) of the TV monitoring system: the left image of each match is the broacasted video (captured in black and white) and the right image of each match is the retrieved video.

TABLE V
TIME COST OF EACH STEP

Local features extraction	$0.086\ s$
Search (DPS^2)	$0.397\ s$ (16.88 feat./s $\times\ t_s = 23.52ms$)
Registration + Vote	$0.203\ s$

Shorter video clips have a high probability to be missed which is not surprising since the average number of local signatures per second is only 17.

2) Television Channel Monitoring: A television monitoring system based on the proposed framework has been developed for copyright protection. A French TV channel has been continuously monitored for 4 months faced with a reference video set containing 30 023 h of SNC video materials (1 813 902 051 local features). The average number of false alarms is about 30 per day and the average number of correct matches is about 320 per day (including trailers, channel events, start and end video clips of programs, etc.). The detection examples presented in the beginning of the paper (Fig. 1 and 2), as well as the results presented on Fig. 17, were obtained in that context. Fig. 17 also presents typical false alarms of the systems. The average time costs of each step, required to monitor 1 s of video, are detailed in Table V.

IV. CONCLUSION AND PERSPECTIVES

As discussed in this paper, CBCR is a challenging CBIR issue. The lack of discrimination and speed of the usual retrieval systems disables their use, whereas on the other hand, dedicated schemes are often not robust enough for all CBCR applications. Previous work did not attempt to take into account the special nature of the similarity existing between two copies. A copy is not only similar to the original document, but was obtained from the original by some specific operations. According to this property, we propose a distortion-based probabilistic approximate similarity search technique that is not based on features distribution in the database but rather on the distribution of the feature distortions. When employed in a global CBCR framework using local features, this technique enables a high speed-up compared to classical range queries and to the sequential scan method. It is asymptotically 2500 times faster than a sequential scan and, in practice, a TV channel can be monitored with a database containing 30 000 h of video. We think that investigations in the modeling of the feature distortions would allow to increase the

performances of the DPS^2 technique. More precision during the search is indeed the best way to reduce the amount of explored data. The construction of learning databases containing real and relevant local matches will be a first step. New algorithms should also be designed to compute quickly the probability of multidimensional blocks for more complex probabilistic models. Extending our grid-structure strategy to multidimensional trees is also a project. This will allow dynamic insertions and will also better distribute the signatures in the chunks reducing the asymptotic increase of the search cost against database size. The hierarchical pruning of the chunks in such structures could be based both on the distribution in the database and a distortion model.

Future work will also focus on local features extraction and databases statistical post-processing. Local features redundancy in the database is problematic both for the speed of the search and the probability to have false alarms. So, we will attempt to reduce it. The detection of spatio-temporal interest points instead of spatial points in key images could be very efficient in this way. A statistical purge of the database is also possible if it is constraint by the number of signatures per image and their geometrical distribution. Labelling interest points with motion categories during the extraction is another perspective. It will allow the post-construction of specific databases dedicated to various applications (copy retrieval, background detection, logo detection, etc.).

Another important issue is the estimation of the transformations that occured between two documents. This could be widely profitable for the challenging tasks of distinguishing a copy and a original. We think that it could be used to enhance the quality of CBCR schemes and also to reconstruct the copies tree of a document which is a high level information (where, when and how a document is used). All of the copies of the tree correspond indeed to several utilization contexts and the links between them are highly informative.

Appendix I
Radius of the Range Search

Let $V_{B\cap R}$ be the intersection between the set of selected p-blocks $B(\tau_{min})$ and the range query of radius r_σ. Without loss of generality, the probability $\alpha_f = \alpha_{B\cap R}$ to find a distorted feature in $V_{B\cap R}$ is equal to

$$\alpha_{B\cap R} = \alpha_B + \alpha_R - \alpha_{B\cup R}$$

where α_B is the probability to find a distorted feature in $B(\tau_{\min})$, α_R the probability to find a distorted feature in the range query and $\alpha_{B\cup R}$ the probability to find a distorted feature in $V_{B\cup R}$, i.e., the union of $B(\tau_{min})$ and the range query. As $\alpha_{B\cup R} \leq 1$ and $\alpha_B \geq \alpha$, then

$$\alpha_{B\cap R} \geq \alpha + \alpha_R - 1.$$

Thus, in order to order to have $\alpha_{B\cap R} \geq \alpha - 0.1\%$, α_R should be equal to $\alpha_R = 0.999$.

For the given normal distortion model, the L^2 norm of the distortion has the following probability density function:

$$p_{\|\Delta S\|}(r) = \frac{f_{\mathcal{N}(0,\sigma)}(r)}{(2\pi\sigma)^{\frac{D-1}{2}}} \frac{\pi^{\frac{D}{2}} D}{\Gamma\left(\frac{D}{2}+1\right)} r^{D-1}$$

where Γ is the gamma function and D is the dimension of the feature space. Thus, the probability α_R of the range query is equal to

$$\alpha_R = \int_0^{r_\sigma} p_{\|\Delta S\|}(r)dr \tag{7}$$

and the radius r_σ for which $\alpha_R = 0.999$ can be estimated at the start of the system by a dichotomy on (7). In practice, r_σ is typically equal to $r_\sigma \approx 6.0\sigma$.

Appendix II
Proof of Equation (6)

Let n_c be the total number of candidate local features. The total search time T_1 of all local features is equal to

$$T_1(\alpha_1) = n_c t_s(\alpha_1)$$

where $t_s(\alpha_1)$ is the average search time of one single local feature when the query probability is set to α_1.

The total search time T_2 of a part only of the candidate features, say $\alpha_3\%$ of the candidate features, is equal to

$$T_2(\alpha_2, \alpha_3) = \alpha_3 n_c t_s(\alpha_2)$$

where $t_s(\alpha_2)$ is the average search time of one single local feature when the query probability is set to α_2.

To be of equivalent quality, the expected final number of retrieved features must be the same in both cases

$$R n_c \alpha_1 = R n_c \alpha_2 \alpha_3$$

where R is the repeatability of the interest points detector. This is equivalent to

$$\alpha_1 = \alpha_2 \alpha_3$$

The probabilistic search of all candidate features will be always faster than searching a part only of them if

$$T_1(\alpha_1) < T_2(\alpha_2, \alpha_3) \forall (\alpha_1, \alpha_2, \alpha_3) | \alpha_1 = \alpha_2 \alpha_3$$

or

$$t_s(\alpha_1) < \alpha_3 t_s(\alpha_2) \forall (\alpha_1, \alpha_2, \alpha_3) | \alpha_1 = \alpha_2 \alpha_3$$

or

$$\frac{t_s(\alpha_1)}{t_s(\alpha_2)} < \frac{\alpha_1}{\alpha_2} \forall (\alpha_1, \alpha_2)$$

which is equivalent to (6).

References

[1] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, Dec 2000.

[2] N. Boujemaa, J. Fauqueur, and V. Gouet, "What's beyond query by exemple?," in *Trends and Advances in Content-Based Image and Video Retrieval LNCS*. New York: Springer-Verlag, 2004.

[3] K. Vu, K. A. Hua, and W. Tavanapong, "Image retrieval based on regions of interest," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 4, pp. 1045–1049, Jul.–Aug. 2003.

[4] S.-A. Berrani, L. Amsaleg, and P. Gros, "Robust content-based image searches for copyright protection," in *Proc. ACM Int. Workshop on Multimedia Databases*, 2003, pp. 70–77.

[5] A. Joly, C. Frélicot, and O. Buisson, "Robust content-based video copy identification in a large reference database," in *Proc. Int. Conf. Image and Video Retrieval*, 2003, pp. 414–424.
[6] Y. Ke, R. Sukthankar, and L. Huston, "Efficient near-duplicate detection and sub-image retrieval," in *Proc. ACM Int. Conf. Multimedia*, New York, 2004.
[7] E. Chang, J. Wang, C. Li, and G. Wilderhold, "Rime—a replicated image detector for the world-wide web," in *Proc. SPIE Symp. Voice, Video, and Data Communications*, 1998, pp. 58–67.
[8] A. Hampapur and R. Bolle, "Comparison of sequence matching techniques for video copy detection," in *Proc. Conf. Storage and Retrieval for Media Databases*, 2002, pp. 194–201.
[9] A. Jaimes, S.-F. Chang, and A. C. Loui, "Duplicate detection inconsumer photography and news video," in *Proc. ACM Int. Conf. Multimedia*, Juan-les-Pins, France, 2002, pp. 423–424.
[10] Y. Meng, E. Y. Chang, and B. Li, "Enhancing dpf for near-replica imagerecognition," in *Proc. Int. Conf. Pattern Recognition*, 2003, pp. 416–423.
[11] A. Jaimes, S.-F. Chang, and A. Loui, "Detection of non-identical duplicate consumer photographs," in *Proc. Pacific Rim Conf. Multimedia*, Singapore, 2003, pp. 16–20.
[12] F. Schaffalitzky and A. Zisserman, "Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?"," in *Proc. European Conf. Computer Vision*, 2002, pp. 414–431.
[13] D.-Q. Zhang and S.-F. Chang, "Detecting image near-duplicate by stochastic attributed relational graph matching with learning," in *ACM Int. Conf. Multimedia*, New York, 2004.
[14] C.-Y. Hsu and C.-S. Lu, "Geometric distortion-resilient image hashing system and its application scalability," in *Proc. Workshop on Multimedia and Security*, Magdeburg, Germany, 2004.
[15] C. Böhm, S. Berchtold, and D. A. Keim, "Searching in high-dimensional spaces: index structures for improving the performance of multimedia databases," *ACM Comput. Surv.*, vol. 33, no. 3, pp. 322–373, 2001.
[16] J. Yuan, L.-Y. Duan, Q. Tian, and C. Xu, "Fast and robust short video clip search using an index structure," in *Proc. Int. Workshop Multimedia Information Retrieval*, 2004, pp. 61–68.
[17] A. K. Jain, A. Vailaya, and W. Xiong, "Query by video clip," *Multimedia Syst.*, vol. 7, no. 5, pp. 369–384, 1999.
[18] A. Ferman, M. Tekalp, and R. Mehrotra, "Robust color histogram descriptors for video segment retrieval and identification," *IEEE Trans. Image Process.*, vol. 11, no. 5, pp. 497–508, May 2002.
[19] S.-C. Cheung and A. Zakhor, "Fast similarity search and clustering of video sequences on the world-wide-web," *IEEE Trans. Multimedia*, vol. 7, no. 3, pp. 524–537, Jun. 2004.
[20] K. Kashino, T. Kurozumi, and H. Murase, "A quick search method for audio and video signals based on histogram pruning," *IEEE Trans. Multimedia*, vol. 5, no. 3, pp. 348–357, Sep. 2003.
[21] J. Oostveen, T. Kalker, and J. Haitsma, "Feature extraction and a database strategy for video fingerprinting," in *Proc. Int. Conf. Visual Information and Information Systems*, 2002, pp. 117–128.
[22] R. Lienhart, C. Kuhmunch, and W. Effelsberg, "On the detection and recognition of television commercials," in *Proc. Int. Conf. Multimedia Computing and Systems*, 1997, pp. 509–516.
[23] K. M. Pua, M. J. Gauch, S. E. Gauch, and J. Z. Miadowicz, "Real time repeated video sequence identification," *Comput. Vis. Image Understand.*, vol. 93, no. 3, pp. 310–327, 2004.
[24] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[25] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 5, pp. 530–535, May 1997.
[26] S. Eickeler and S. Müller, "Content-based video indexing of tv broadcast news using hidden markov models," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, 1999, pp. 2997–3000.
[27] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, "The r *-tree: an efficient and robust access method for points and rectangles," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1990, pp. 322–331.
[28] S. Berchtold, D. A. Keim, and H.-P. Kriegel, "The x-tree: An index structure for high-dimensional data," in *Proc. Int. Conf. Very Large Data Bases*, 1996, pp. 28–39.
[29] N. Katayama and S. Satoh, "The sr-tree: An index structure for high-dimensional nearest neighbor queries," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1997, pp. 369–380.
[30] R. Weber, H. J. Schek, and S. Blott, "A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces," in *Proc. Int. Conf. Very Large Data Bases*, 1998, pp. 194–205.
[31] S. Berchtold, C. Böhm, and H. P. Kriegel, "The pyramid-tree: breaking the curse of dimensionality," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1998, pp. 142–153.
[32] C. Faloutsos and K.-I. Lin, "Fastmap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1995, pp. 163–174.
[33] L. Amsaleg, P. Gros, and S.-A. Bernini, "Robust object recognition in images and the related database problems," *J. Multimed. Tools and Applic. (Special Issue)*, vol. 23, pp. 221–235, 2003.
[34] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," in *Proc. ACM Symp. Theory of Computing*, 1998, pp. 604–613.
[35] P. Zezula, P. Savino, G. Amato, and F. Rabitti, "Approximate similarity retrieval with m-trees," *Very Large Data Bases J.*, vol. 7, no. 4, pp. 275-–293, 1998.
[36] P. Ciaccia and M. Patella, "PAC nearest neighbor queries: approximate and controlled search in high-dimensional and metric spaces," in *Proc. Int. Conf. Data Engineering*, 2000, pp. 244–255.
[37] R. Weber and K. Böhm, "Trading quality for time with nearest neighbor search," in *Proc. Int. Conf. Extending Database Technology*, 2000, pp. 21–35.
[38] C. Li, E. Chang, M. Garcia-Molina, and G. Wiederhold, "Clustering for approximate similarity search in high-dimensional spaces," *IEEE Trans. Knowl. Data Eng.*, vol. 14, no. 4, pp. 792–808, Aug. 2002.
[39] K. P. Bennett, U. Fayyad, and D. Geiger, "Density-based indexing for approximate nearest-neighbor queries," in *Proc. Conf. Knowledge Discovery in Data*, 1999, pp. 233–243.
[40] S.-A. Berrani, L. Amsaleg, and P. Gros, "Approximate searches: k-neighbors + precision," in *Proc. Int. Conf. Information and Knowledge Management*, 2003, pp. 24–31.
[41] E. Tuncel, H. Ferhatosmanoglu, and K. Rose, "Vq-index: An index structure for similarity searching in multimedia databases," in *Proc. ACM Int. Conf. Multimedia*, 2002, pp. 543–552.
[42] M. E. Houle and J. Sakuma, "Fast approximate similarity search in extremely high-dimensional data sets," in *Proc. Int. Conf. Data Engineering*, 2005, pp. 619–630.
[43] H. Ferhatosmanoglu, E. Tuncel, D. Agrawal, and A. Abbadi, "Approximate nearest neighbor searching in multimedia databases," in *Proc. Int. Conf. Data Engineering (ICDE)*, 2001, pp. 503–511.
[44] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: an efficient data clustering method for very large databases," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1996, pp. 103–114.
[45] M. L. Miller, M. A. Rodriguez, and I. J. Cox, "Audio fingerprinting: nearest neighbor search in high dimensional binary spaces," in *IEEE Workshop on Multimedia Signal Processing*, 2002, pp. 182–185.
[46] A. Joly, C. Frélicot, and O. Buisson, "Feature statistical retrieval applied to content-based copy identification," in *Proc. Int. Conf. Image Processing*, 2004.
[47] K. Mikolajczyk, T. Tuylelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A comparison of affine region detectors," *Int. J. Comput. Vis.*, 2004.
[48] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. Int. Conf. Computer Vision*, 1999, pp. 1150–1157.
[49] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce, "Object modeling and recognition using local affine-invariant image descriptors and multi-view spatial contraints," *Int. J. Comput. Vis.*, 2004.
[50] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

Alexis Joly received the Eng. degree in telecommunications from the National Institute of Applicative Sciences (INSA), Lyon, France, in 2001 and the Ph.D. degree in computer science from the University of La Rochelle, La Rochelle, France, in 2005.

During his Ph.D. work, he collaborated with the French National Institute of Audiovisual (INA) to develop a powerful TV monitoring system. In 2005, he spent few months at Tokyo National Institute of Informatics and then joined the IMEDIA team at INRIA Rocquencourt, Le Chesnay Cedex, France, where he is currently a Junior Research Scientist. He is a member of the MUSCLE European Network of Eexcellence and participates in several European and national research projects. His research interests are linked to image and video content description, multimedia search engines, similarity searches, and indexing techniques in very large databases.

Olivier Buisson received the Ph.D. degree in computer science from the University of La Rochelle, La Rochelle, France, in 1997.

From 1998 to 1999, he developed color movie restoration software for the Ex-Machina Company. Since 1999, he has been with the research group of the French National Institute of Audiovisual (INA). From 1999 to 2002, he worked on BRAVA, a European research project, with the aim to automatically restore video films in real time. Since 2002, he has led a team of researchers in video content indexing. His research focuses on visual descriptors for images and videos, visual search engines for very large databases of videos and images, and similarity measures of visual descriptors. His goal is to build bridges between these three different research activities in order to have scalable content search engines.

Carl Frélicot received the Ph.D. degree in system control in 1992 from the Compiegne University of Technology, Compiegne, France.

He joined the University of La Rochelle, La Rochelle, France, as an Assistant Professor in 1993, where he has been the Head of the Pattern Recognition Group of the L3i laboratory for two years. He is currently a Full Professor and the Head of the Computer Science Department, University of La Rochelle. His research interest include algebraic operators applied to information fusion, statistical and fuzzy pattern recognition, and image and video analysis.