Duplicate Image Detection in Large Scale Databases

4 authors, including:

Elisa Drelie Gelasca
Italian National Research Council
32 PUBLICATIONS **1,022** CITATIONS

Kalpathi Ramakrishnan
Indian Institute of Science
167 PUBLICATIONS **3,294** CITATIONS

B. S. Manjunath
University of California, Santa Barbara
516 PUBLICATIONS **30,711** CITATIONS

Chapter 1

Duplicate Image Detection in
Large Scale Databases

Pratim Ghosh, E. Drelie Gelasca, K.R. Ramakrishnan† and B. S. Manjunath

Vision Research Lab., Electrical and Computer Engineering Department,
University of California, Santa Barbara 93106-9560.
†Indian Institute of Science, Bangalore, 560 012,India.

We propose an image duplicate detection method for identifying modified copies of the same image in a very large database. Modifications that we consider include rotation, scaling and cropping. A compact 12 dimensional descriptor based on Fourier Mellin Transform is introduced. The compactness of this descriptor allows efficient indexing over the entire database. Results are presented on a 10 million image database that demonstrates the effectiveness and the efficiency of this descriptor. In addition, we also propose extension to arbitrary shape representations and similar scene detection and preliminary results are also included.

1.1. Introduction

Automated robust methods for duplicate detection of images/videos is getting more attention recently due to the exponential growth of multimedia content on the web. The large quantity of multimedia data makes it infeasible to monitor them manually. In addition, copyright violations and data piracy are significant issues in many areas including digital rights management and in the entertainment industry. In this chapter, our main aim is to propose a system that can detect *duplicate* images in very large image databases. We specifically focus on the scalability issue. Our proposed approach results in a very compact image signature that is robust to many image processing operations, can be indexed to efficiently search large databases (we show results on a 10 million image database), and is quite effective (about 82% precision). Our method can also be extended for similar image detection and "region of interest" duplicate detection.

In many practical scenarios, the duplicates are not identical replicas of the images in the database, but are digitally processed versions of the original images in the database. In these cases, standard hashing methods will not work. Here, "duplicate" refer to digitally modified versions of the image after manipulations such as those shown in Figure 1.1. Duplicate detection of exact copy using hashing techniques has been already addressed in the literature.[1,2] Figure 1.1 (a) shows the original image and Figures 1.1 (b)-(p) are obtained after digital processing such as

2   *Pratim Ghosh, E. Drelie Gelasca, K.R. Ramakrishnan and B. S. Manjunath*

scaling, rotation and cropping. One can consider duplicate detection as a subset of similarity search and retrieval, see for example .[3–6]   Real time retrieval from
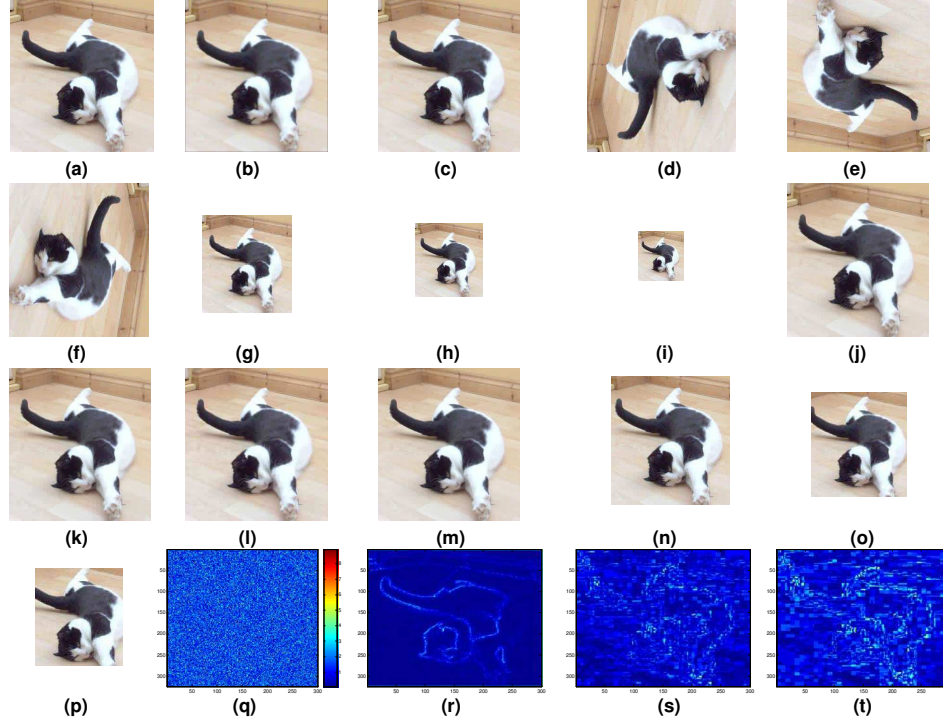

Fig. 1.1.   (a) original image; (b) gaussian noise added image; (c) blurred image; (d)-(f) rotated images: $90^o, 180^o, 270^o$; (g)-(i) scaled images: $75\%, 50\%, 25\%$; (j)-(m) JPEG compressed images: 90, 70, 50, 30; (n)-(p) cropped images: $10\%, 20\%, 30\%$; (q)-(t) difference images with respect to the original one for (b),(c),(l) and (m). The compact signatures of all these images are summarized in the Table 1.1 sequentially.

a large image archive such as the World Wide Web (WWW) necessarily demands robust systems in terms of

- *efficiency*, time performance;
- *accuracy*, precision and recall;
- *scalability*, the property of accommodating significant changes in data volume without affecting the system performance.

Many of the results reported in the literature are on small databases, ranging from a few thousand (e.g.,[7–9]) to 1.4 million images in the case of Wang *et al.*[10]

    The key steps in our duplicate detection includes the computation of the Fourier Mellin Transform (FMT)[11] followed by a dimensionality reduction resulting in a 12 dimensional quantized vector. These quantized vectors are represented using unsigned characters (total of 96 bits). This compact representation allows us to

Table 1.1.   Operations corresponding 4 bytes
signatures for images in Figure 1.1.

Operations performed	Compact Signature
Original Image	122 197 157 73
Gaussian Noise addition	122 197 156 73
Gaussian Blurring (2)	122 196 158 73
Rotation (90)	122 197 155 73
Rotation (180)	122 197 155 73
Rotation (270)	121 197 157 73
Scaling down (x1.3)	115 196 166 74
Scaling down (x2)	103 197 178 80
Scaling down (x4)	93 194 181 95
JPEG Compressed (90)	122 197 156 73
JPEG Compressed (70)	122 196 156 74
JPEG Compressed (50)	122 197 156 72
JPEG Compressed (30)	122 197 156 72
Cropping (by 10%)	129 201 146 72
Cropping (by 20%)	127 203 154 79
Cropping (by 30%)	126 205 165 86

build an efficient indexing tree, such as a k-d tree, that can search the database in 0.03 seconds on an Intel Xeon with CPU 2.33GHz. The accuracy is evaluated using a query data set of 100 images. We are also exploring the use of clustering methods for approximate search and retrieval and results are presented.

The rest of the chapter is organized as follows. Section 1.2 gives an overview of related work. In Section 1.3, we present the details of the proposed duplicate detection method. Extensions of the algorithm for sub image retrieval are also proposed. Section 1.4 discusses the performance of compact signature on a very large database. The results for sub image duplicate detection and detection of similar images taken with slightly different illumination conditions, different point of views, rotations and occlusions are also demonstrated. Finally, we conclude in Section 1.5 with some discussions.

1.2. Related Work

Many duplicate detection[8,9] and sub-image retrieval [5,7,12,13] schemes have been proposed in the literature. Maret *et. al*[8] proposed duplicate detection based on support vector classifier. Different image features such as color and texture are first extracted from the image. Distances are then computed in the respective feature space and finally the dissimilarity between two images is given by the summation of these partial distances. A 138 dimensional feature is computed for each image, on a 18 thousand image database. The high dimensionality of the feature vector is a limiting factor in scaling this approach to large databases.

Another method, RAM (Resolving Ambiguity by Modification),[9] was proposed for duplicate detection using Analytical Fourier Mellin Transform (AFMT). First,

in the pre-processing stage, the feature vectors are extracted from the database. A modified version of the original feature space is obtained by increasing the mutual distances among the features maintaining the semantic content of the image. Second, the algorithm searches through the modified database for a given query. A constrained optimization problem was solved in order to generate the modified feature space. This optimization problem has $d \times n$ variables and a minimum of n^2 constraints where d and n are the dimensions and number of points considered respectively (specifically, $d = 400$ was used in their case). This method also suffers from the scalability issue and some ad hoc post processing steps were suggested in the paper to address this.

There are also methods that deal with sub image retrieval.[5,7,12,13] The main idea of all these approaches is based on extracting a large number of local features and then using sophisticated algorithms for their efficient matching. These methods are computationally very expensive and require significant amount of storage/memory. In the above mentioned sub image retrieval methods, the database size ranges from few hundreds to thousands and their scalability is not demonstrated.

Few web image search engines for large scale duplicate detection have been also proposed.[3,10] RIME (Replicated IMage dEtector)[3] detects duplicate images by representing them with feature vectors (wavelet co-efficients) and employing an indexing scheme (multidimensional extensible hashing) to index the high dimensional feature vectors. Experimental results on a database of about 30000 images are provided. In,[10] each image was compactly represented (≤ 32 *bits*) by a hash code. These compact hash codes are then compared for duplicate image retrieval yielding a high precision recall ratio (more than 90%) on 1.4 million images considering only the simple manipulations such as minor scale changes, image storage format (PNG, JPEG, GIF) conversions and color/grayscale conversion.

Our proposed method makes the following contributions:

- a compact signature that can be used to detect duplicates when the original image is modified significantly is proposed;
- the compactness of the signature allows efficient indexing tree to be built;
- the scheme shows to be scalable for large image database containing over 10 million images;
- possible extensions to similar scene and region of interest identification are also shown.

In the following section, we discuss the system level issues in more detail.

1.3. System Overview

The overall block diagram of the web duplicate image retrieval system is depicted in Figure 1.2. The current version of our system contains about 10 million images. The database used in these experiments can be found at `http://cortina.ece.ucsb.`
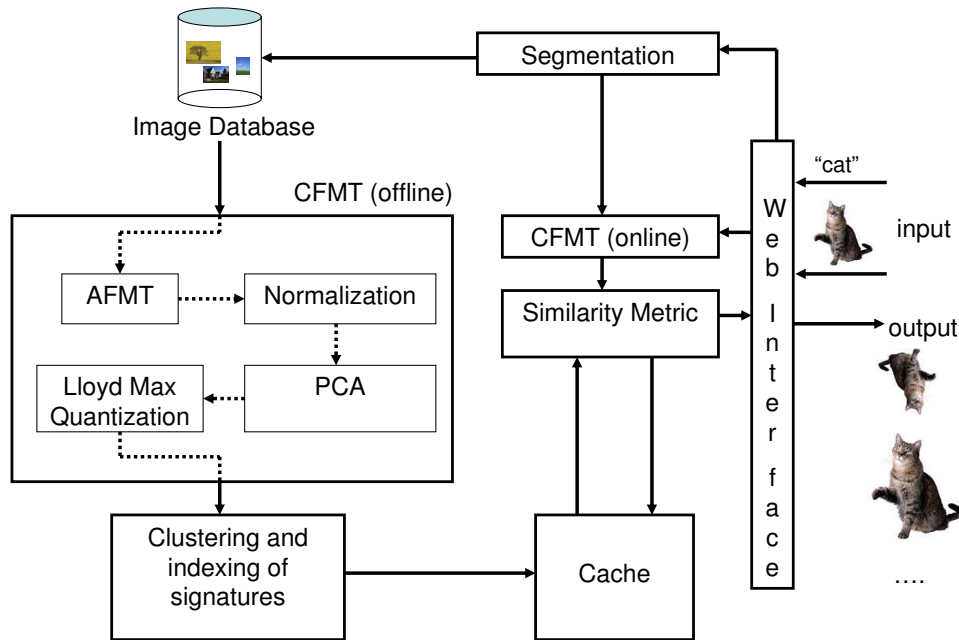
Fig. 1.2.   System Architecture.

edu/. These images are downloaded from the web using a web crawler and stored into the image database along with the associated meta-data (text and keyword).

CFMT block. The CFMT (Compact Fourier Mellin Transform) is computed for each image in the database. It takes approximately 50 msec in our current C implementation to compute this descriptor. The details of the CFMT algorithm are discussed in details in Section 1.3.1.

K-d tree indexing. A k-d tree indexing scheme is also implemented to structurally range the signatures for fast search and retrieval. It takes around 30 msec to retrieve the 20 nearest neighbors for a given query from the entire database. Indexing performance is discussed in Section 1.4.

Similarity metric. Both L_1 and L_2 distance measure have been implemented for comparing the feature vectors. The L_2 distance measure was found to improve the results marginally.

Arbitrarily shaped region based CFMT. On a smaller dataset (MM270K with about 18000 images) we have tested an adaptation of CFMT algorithm for arbitrarily shaped regions. Firstly, GPAC (Graph Partitioning Active contours), a recently proposed segmentation scheme[14] is applied to segment foreground regions within the image. The GPAC method was selected after exploring different foreground/background segmentation methods (e.g. active contour model by Chan and Vese[15] and Geodesic Active Contour[16]) since it gives better results overall.

Then, the CFMT is extracted on the foreground region instead of the whole image. The adaptation of CFMT algorithm for arbitrarily shaped regions is presented in Section 1.3.2 and preliminary results in Section 1.4.4.

1.3.1. *CFMT Descriptor for Images*

Fourier-Mellin transform (FMT) has been studied extensively in the context of watermarking[17,18] and invariant object recognition.[19–21] All these methods exploit the fact that this transform generates a rotation, translation and scale invariant representation of the images. The FMT was first introduced in[11] and our implementation is based on the fast approximation described in.[19]

The classical FMT of a 2D function f, $T_f(k, v)$ is defined as:

$$T_f(k, v) = \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} f(r, \theta) r^{-iv} e^{-ik\theta} d\theta \frac{dr}{r} \qquad (1.1)$$

where (k, v) and (r, θ) are respectively the variables in Fourier Mellin and polar domain representation of the function f. Ghorbel [22] suggested the AFMT, an important modification to the problem associated with the existence of standard FM integral (the presence of $\frac{1}{r}$ term in the definition necessarily requires f to be proportional to r around the origin such that when $r \to 0$ then $f \to 0$). The AFMT, $T_{f\sigma}(k, v)$, is defined as:

$$T_{f\sigma}(k, v) = \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} f(r, \theta) r^{\sigma-iv} e^{-ik\theta} d\theta \frac{dr}{r} \qquad (1.2)$$

where σ, a strictly positive parameter, determines the rate at which f tends toward zero near the origin.

Let $f_1(x, y)$ be an image and its rotated, scaled, translated version $f_2(x, y)$ be related by the equation:

$$f_2(x, y) = f_1(\alpha(x \cos\beta + y \sin\beta) - x_o, \alpha(-x \sin\beta + y \cos\beta) - y_o) \qquad (1.3)$$

where the rotation and scale parameters are β and α respectively, and $[x_o, y_o]$ is the translation. It can be shown that for rotated and scaled images, the magnitudes of the AFM transforms, $|T_{f_1\sigma}|$ and $|T_{f_2\sigma}|$, (corresponding to f_1 and f_2) are related by the equation:

$$|T_{f_2\sigma}(k, v)| = \alpha^{-\sigma} |T_{f_1\sigma}(k, v)| \qquad (1.4)$$

Concisely, an AFMT leads to a scale and rotation invariant representation after proper normalization by $1/\alpha^{-\sigma}$. Finally, the CFMT representation can be made translation invariant by computing the AFMT on the Fourier transformed image (considering only the magnitude part).

Once the AFM coefficients are extracted, Principal Component Analysis (PCA)[23] and Lloyd Max non uniform scalar quantization[24] are applied to obtain a compact representation, the CFMT descriptor. Each dimension of the CMFT descriptor is quantized to 256 levels. After extensive experimentation, we choose

the 12 dimensional CFMT descriptor for our duplicate detection since it provided a good trade off between accuracy and efficiency.

1.3.2. *CFMT Extraction for Arbitrarily Shaped Regions*

Here we extend the CFMT computation for arbitrarily shaped regions (SA-CFMT, Shape Adaptive CFMT). This is useful in many applications where one is looking for specific objects or regions of interest within a larger image. A schematic of this region of interest CFMT computation is shown in Figure 1.3. We first applied the
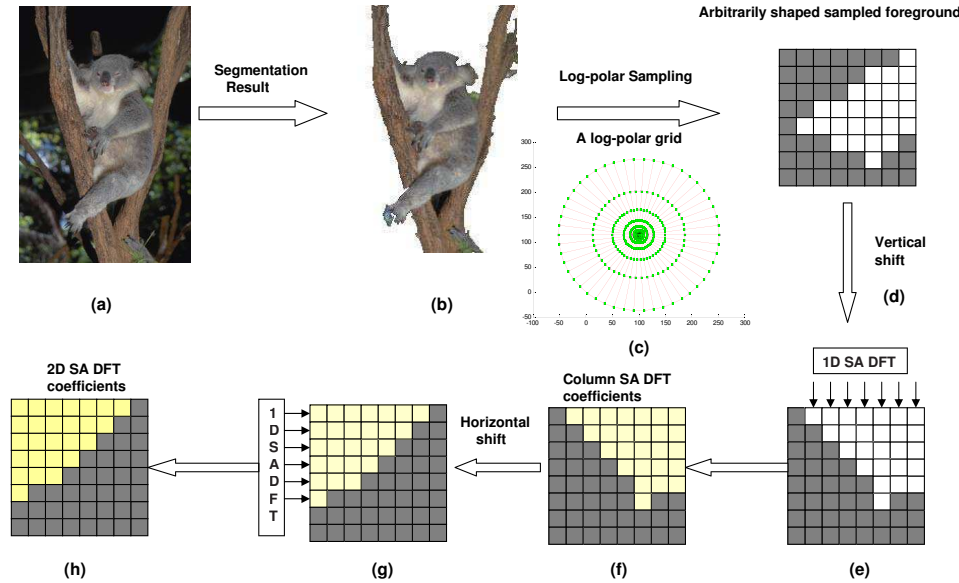

Fig. 1.3.   A typical 2D SA DFT work flow: (a) original image, (b) segmented Region of Interest(ROI), (c)-(d) sampled foreground using the log-polar grid, (e) up-shifting and 1D SA DFT on each column, (f) column SA DFT coefficients, (g) left-shifting and 1D SA DFT on each row, (h) final 2D SA DFT coefficients. Darker and brighter regions correspond to background and foreground respectively in all these matrices.

GPAC[14] segmentation on a given image to extract the foreground region. Then a log-polar transform is computed with the center of the coordinate system for the transformation being the centroid of the region of interest. The pixel values inside the foreground are mapped to a log-polar sampling grid and the rest of the positions in the grid are filled with zeroes. Since all the grid positions do not correspond to foreground, normal 2D FFT can not be employed on the sampled values directly. Instead, we use the Shape Adaptive Discrete Fourier Transform (SA-DFT).[25] SA-DFT was first proposed for coding of arbitrary shaped image segments in the MPEG-4 image compression standard.

The SA DFT coefficients of a vector $x[n]$ where $n = 0, 1, 2, ..., N - 1$ are com-

puted in a two step approach:

(1) Let $x[n]$ has N_s samples belonging to the foreground and the rest to the background samples. Also consider a new sequence $x'[n]$ to be constructed using only the foreground samples of $x[n]$. Two cases can occur. In the first case, the foreground samples can form a contiguous cluster:

$$x[n] = \{0, 0., ., 0, a_1, a_2, a_3, .., a_{N_s}, 0, ..0, 0\}$$

where $\{a_i\}_{i=1,2,...,N_s}$ denotes the foreground and the zeros are the background samples. In this case, $x'[n]$ is obtained by taking the contiguous block from $x[n]$ e.g. $x'[n] = \{a_1, a_2, a_3, ..., a_{N_s}\}$. In the second case, the foreground samples in $x[n]$ can be separated by the background samples like:

$$x[n] = \{0, 0.., a_1, a_2, 0, 0, 0, a_3, a_4, a_5, 0, 0, .., a_{N_s}, 0, 0\}$$

Therefore, in this case, $x'[n]$ is constructed by replacing the background ones with following foreground samples e.g. $x'[n] = \{a_1, a_2, a_3, ..., a_{N_s}\}$ ($x'[n]$ is the condensed version of $x[n]$ without any background samples). Also the relative positions of the foreground samples in $x[n]$ are maintained in $x'[n]$.

(2) Then, a N_s point DFT is applied to $x'[n]$, followed by a scaling of $1/\sqrt{N_s}$ which preserves the orthogonality property of the DFT 2D transform. Let us define, $X'[k]$ where $k = 0, 1, 2, ..., N_s - 1$ be the DFT of $x'[n]$. The required number of zeros are padded at the end of the sequence $X'[k]$ to have the same length as the input vector $x[n]$. Thus, $X'[k]$ gives the SA DFT of $x[n]$.

Like other separable transforms SA DFT is also applicable to two dimensional matrices. Firstly, each column is processed using above mentioned 1D algorithm and secondly the same is applied to each row of the results. Given the 2D SA DFT representation for an image we extract the CFMT signature in the same way as described in Section 1.3.1 and finally obtain the SA-CFMT.

1.4. Experimental Results

We now describe the evaluation metric used to asses the performance of the proposed CFMT signature. Then, we proceed to present experimental results on duplicate detection for both whole and segmented image. Time performance is also discussed.

1.4.1. *Performance Evaluation*

Precision-recall value has been used to measure the performance of our signature. Let $A(H, \Gamma)$ be the set of H retrievals based on the smallest distances from the query image, Γ, in the signature space and $C(\Gamma)$ be the set of D images in the database relevant to the query Γ. Then, precision P is defined by the number of images retrieved relevant to query image divided by the set of retrievals, H.

$$P(H, \Gamma) \stackrel{\text{def}}{=} \frac{|A(H, \Gamma) \bigcap C(\Gamma)|}{H}$$

Fig. 1.4.   Original image, log-polar transformed image and reconstructed image (from left to right) using only ~ 50 % of the total A.C. energy. Overall shape remains unchanged in the reconstructed image.

Recall which is defined as

$$R(H, \Gamma) \stackrel{\text{def}}{=} \frac{|A(H, \Gamma) \bigcap C(\Gamma)|}{D}$$

is the proportion of relevant images retrieved from $C(\Gamma)$. A precision-recall curve is usually obtained by averaging precision and recall values over a large number of queries Γ to obtain a good estimate.

1.4.2. *Results on Web Image Database*

In our implementation of AFMT, the image is first mapped to a log-polar domain and a 2D FFT is then computed on that domain. A 71×71 grid has been found to be adequate for the log-polar mapping. We extract all Fourier Mellin (FM) coefficients lying within a fixed radius, the *target radius*, from the center. We choose the *target radius* in such a way so that the energy of the AFM coefficients within it corresponds to 50% of the total AFM coefficients energy. Within the *target radius* (which in our implementation is 8 pixels) there are 96 independent AFM coefficients. The AFM coefficients are normalized by the central FM harmonic to get rid of the $\alpha^{-\sigma}$ term (see Eq. 1.4). Figure 1.4 shows the original image and the reconstructed image using the AFM coefficients which correspond to 50% of the total A.C energy.

A set of 100 random images are chosen as queries and for each of the query images 15 duplicates are generated by performing the operations described in Table 1.1.

Varying sizes of CFMT signature include: 4, 6, 8 and 12 dimensions with one byte per dimension. To give an idea of how much the signatures varies among duplicate images, the 4 dimensional CFMT representations for the images shown in Figure 1.1 are reported in Table 1.1.

Figure 1.5 shows the retrieval results for various sizes of CFMT signatures. Note that for the 12 dimensional CFMT signature for $H=15$ (at the knee point) the corresponding precision and recall are $P = 0.82$, $C = 0.81$.   In Figure 1.6, a comparative study is obtained to show the scalability of CFMT signatures as the
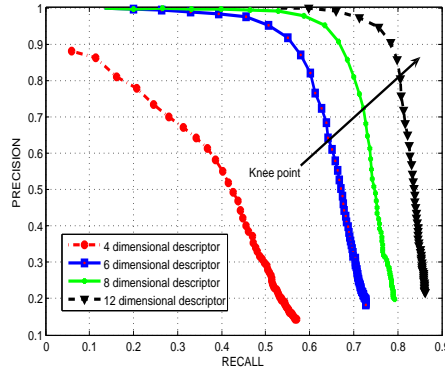
Fig. 1.5.   Precision Recall curve on close to a 10 million image database averaged on 100 queries, each with 15 duplicates.
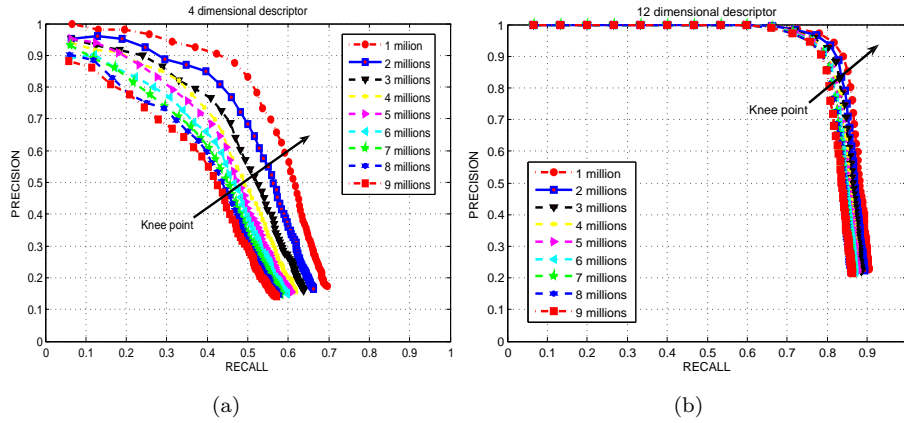

Fig. 1.6.   Scalability performance of various signatures: (a) performance of 4 dimensional descriptor; (b) performance of 12 dimensional descriptor.

size of the database increases starting from 1 million up to 9 millions. It is clear from the figure that the 12 dimensional descriptor scales quite well with the size of the database.

1.4.3.   *Time Performance*

We investigated different approaches to improve the run time performance of our system. A naive sequential search over the 10 million image database takes approximately 3 seconds for retrieving the 20 nearest neighbors. A k-d tree indexing data structure is also implemented. The k-d tree index structure built on a 12 dimensional feature space takes only about 0.03 seconds to retrieve the 20 nearest neighbors. It takes about 3 minutes to build this data structure and requires 1.5

GB of main memory to keep the data structure. Note that the entire k-d tree needs to be kept in memory during the query-retrieval time. Such high memory requirement might be crucial. In fact, if we increase our database size by 50% the k-d tree structure would require more than 2 GB of main memory. This motivated us to investigate clustering based methods for approximate nearest neighbor search and retrieval. The performance of the simple K-means clustering is summarized in Table 1.2. For the 10 million images with 64 clusters one can get about 65.6% accurate results with the search time of about 1.8 seconds. These clustering results are preliminary and suggest a trade off between accuracy and computations.

Table 1.2.   Speed and accuracy using sequential search and K-means clustering.

#clusters	none	32	64
# points	11033927	1085509	583381
search (sec)	3.014	2.841	1.826
accuracy	82%	77.9%	65.6%

1.4.4. *Results on MM270K Image Database*

Preliminary results have also been obtained for region and similar scene retrieval on a smaller dataset. The MM270K database used in these experiments can be downloaded from `http://www.cs.cmu.edu/~yke/retrieval`.

Similar Scene Retrieval. In this case, the duplicates correspond to images of the same scene acquired under different imaging conditions. For example, these images are captured at different time, from different view point and may have occluded regions. See Figure 1.7 for some examples. The CFMT descriptor in its current form is not translation invariant and needs further modifications to address this issue. One simple solution is to construct the CFMT descriptor on top of the Fourier Transform of the image. Performance can be further improved by increasing the dimensionality of the descriptor. The precision recall curve obtained for the whole MM270K dataset is depicted in Figure 1.8 for the case of 12 dimensional and 36 dimensional modified descriptor. In this graph, the results are averaged over 14 queries with each having 4 similar scenes in the database. As can be seen from the graph, these preliminary results are quite promising.

Arbitrarily shaped region retrieval. The GPAC[14] segmentation method was used to automatically compute the foreground and background segmentation from the MM270K database for this experiment. We constructed 40 query examples, each having 12 duplicates. These duplicates correspond to the modifications (b)-(m) in Figure 1.1. GPAC segmentation was applied to the MM270K database, to the originals and its duplicates. Some results are shown in Figure 1.9. There was no manual parameter tuning on these results. The SA-CFMT descriptors was then computed on these segmented region as discussed in Section 1.3.2. We also

12   *Pratim Ghosh, E. Drelie Gelasca, K.R. Ramakrishnan and B. S. Manjunath*
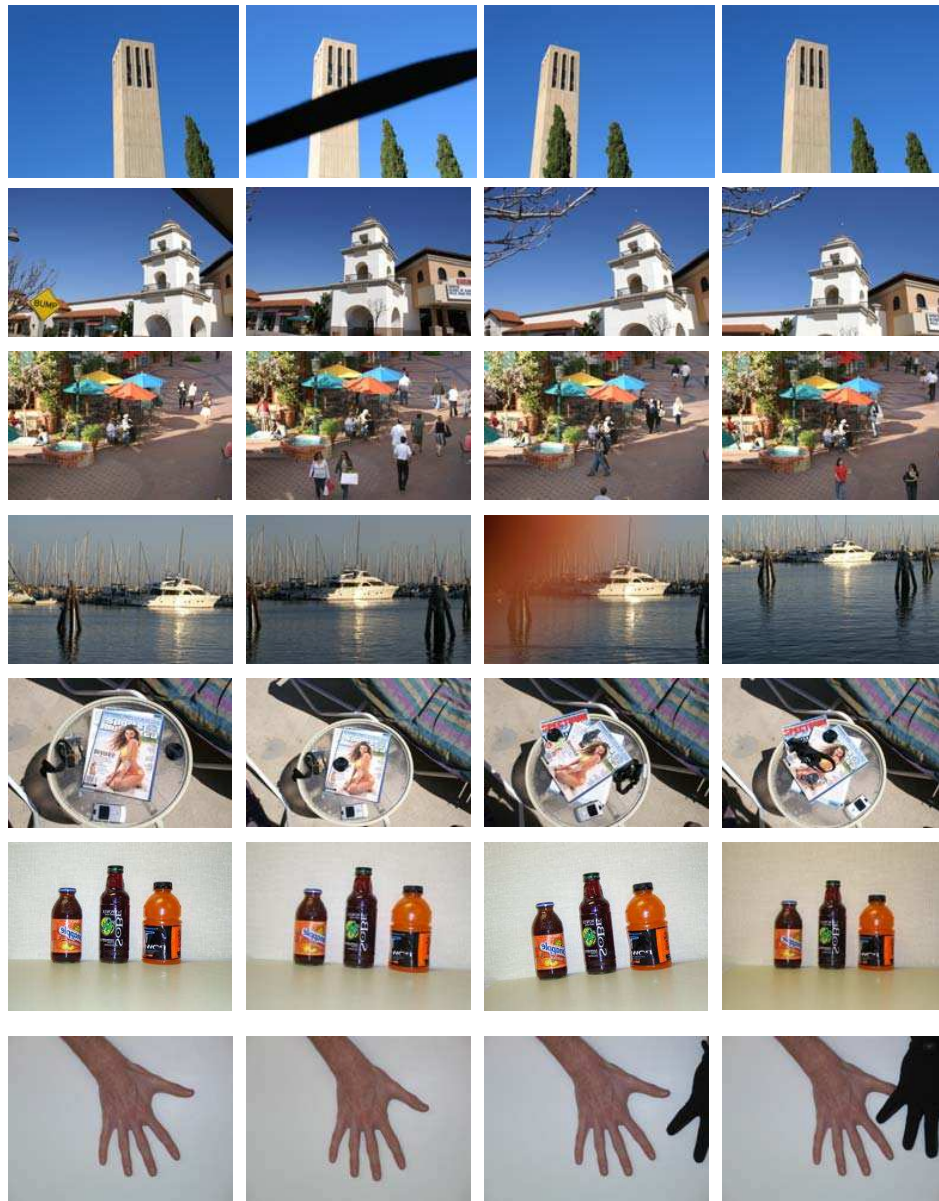

Fig. 1.7.   Tested scene changes in the similar scene retrieval experiments. Similar images are taken with: slightly different view points, camera setting, occlusions, rotation and photometric changes.

computed the CFMT for the whole image with different kind of backgrounds as shown in Figure 1.10. Figure 1.11 shows the precision recall curve for MM270K database with CFMT (whole image) compared to GPAC plus SA-CFMT (region
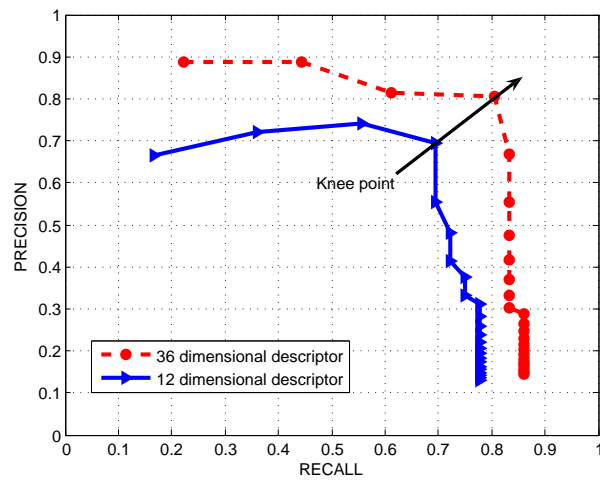
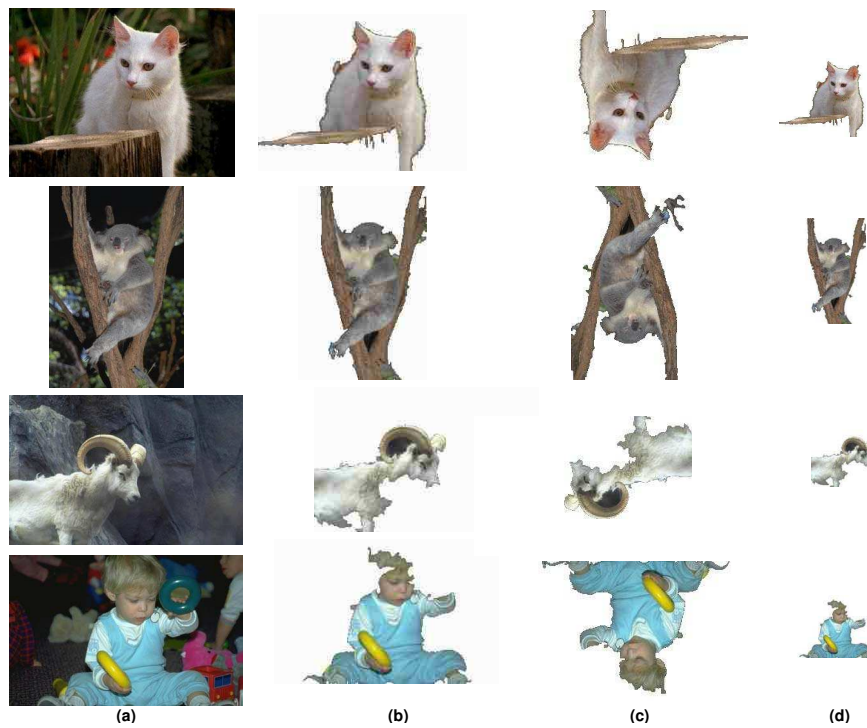Fig. 1.8.   Precision Recall curve on MM270K averaged on 14 queries, each with 4 similar scenes.

Fig. 1.9.   (a) original images; (b)-(d) segmentation results on: original images, 180° rotated version of the original images, 25% scaled version of the original images.
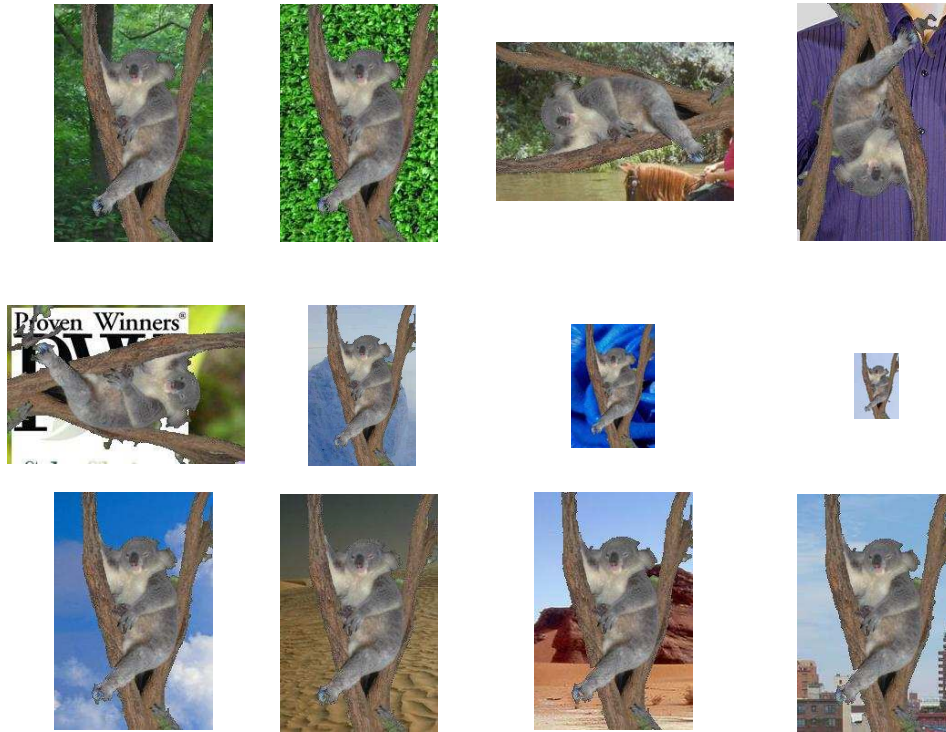
Fig. 1.10.   Backgrounds used for testing CFMT and SA-CFMT in MM270K.

based). Note that a precision of 61% is achieved with a recall rate of 60% at the knee point for $H = 12$ for GPAC plus SA-CFMT and very low precision values are obtained by using only CFMT on the whole image for any size of signature.

1.5. Conclusion and future work

In this chapter we have presented a scalable duplicate detection method. The scalability of the 12 dimensional CFMT signature has been demonstrated for a web image database containing about 10 million images. We have provided detailed experimental results demonstrating the accuracy and efficiency of the proposed approach. On the 10 million image database we get about 82% accuracy with a search time of about 30 msec on a standard desktop. Preliminary results for arbitrarily shaped similar region retrieval as well as similar scene detection are very promising.
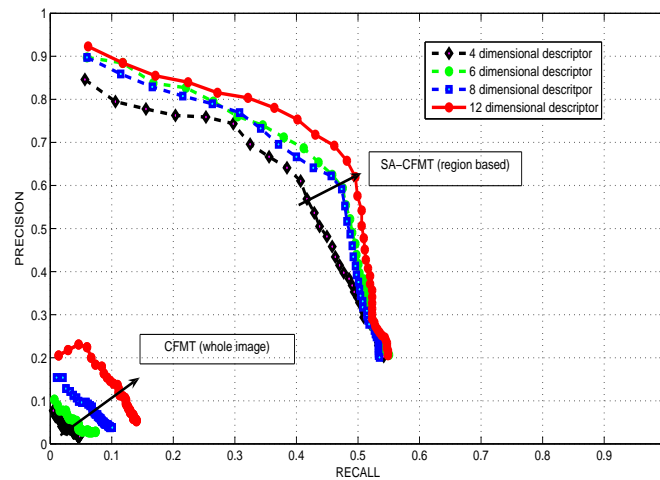
Fig. 1.11.   Precision Recall curve on MM270K averaged on 40 queries, each with 12 duplicates.

1.6. Acknowledgments

References

1. C. S. Lu, C. Y. Hsu, S. W. Sun, and P. C. Chang. Robust mesh-based hashing for copy detection and tracing of images. In *ICME*, vol. 1, pp. 731–734 (June, 2004).
2. R. Venkatesan, M. H. J. S. M. Koon, and P. Moulin. Robust image hashing. In *Int. Conf. Image Processing*, vol. 3, pp. 664–666 (Sept, 2000).
3. E. Chang, J. Wang, C. Li, and G. Wiederhold. RIME: A replicated image detector for the world-wide web. In *SPIE Multimedia Storage and Archiving Systems* (November, 1998).
4. J. Fridrich, D. Soukal, and J. Lukas. Detection of copy-move forgery in digital images. In *Digital Forensic Research Workshop* (August, 2003).
5. J. Luo and M. Nascimento. Content based sub-image retrieval via hierarchical tree matching. In *ACM Workshop on Multimedia Databases*, (2003).
6. Y. Meng, E. Chang, and B. Li. Enhancing dpf for near-replica image recognition. In *IEEE Computer Vision and Pattern Recognition*, (2003).
7. Y. Ke and R. Suthankar. Efficient near duplicate detection and sub image retrieval. In *ACM Multimedia* (August, 2004).
8. Y. Maret, F. Dufaux, and T. Ebrahimi. Image replica detection based on support vector classifier. In *Optical Information System III, SPIE*, vol. 5909, pp. 173–181, (2005).
9. S. Roy and E. C. Chang. A unified framework for resolving ambiguity in copy detection. In *ACM Multimedia*, pp. 648–655, (2005).

16   *Pratim Ghosh, E. Drelie Gelasca, K.R. Ramakrishnan and B. S. Manjunath*

10. B. Wang, Z. Li, M. Li, and W. Y. Ma. Large-scale duplicate detection for web image search. In *ICME*, pp. 353–356 (July, 2006).

11. D. Casasent and D. Psaltis, Scale invariant optical transform, *Opt.Eng.* **15**(3), 258–261, (1976).

12. N. Sebe, M. S. Lew, and D. P. Huijsmans. Multi-scale sub-image search. In *ACM Multimedia (2)*, pp. 79–82, (1999).

13. D. Zhang and S. F. Chang. Detecting image near-duplicate by stochastic attributed relational graph matching with learning. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pp. 877–884 (October, 2004).

14. B. Sumengen and B. S. Manjunath, Graph partitioning active contours (GPAC) for image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI).* **28**(4), 509–521 (Apr, 2006).

15. T. F. Chan and L. A. Vese, Active contours without edges, *IEEE Transactions on Image Processing.* **10**(2), 266–277 (February, 2001).

16. V. Casellas, R. Kimmel, and G. Sapiro, Geodesic active contours, *International Journal of Computer Vision.* **22**(1), 61–79, (1997).

17. C. Y. Lin, M. Yu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, Rotation scale and translation resilient watermarking for images, *IEEE Transaction on Image Processing.* **10**, 767–782 (May, 2001).

18. D. Zheng and J. Zhao. LPM-based RST invariant digital image watermarking. In *Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on*, vol. 3, pp. 1951–1954 (May, 2003).

19. S. Derrode and F. Ghorbel, Robust and efficient Fourier-Mellin transform approximations for gray-level image reconstruction and complete invariant description, *Computer Vision and Image Understanding: CVIU.* **83**(1), 57–78, (2001).

20. N. Gotze, S. Drue, and G. Hartmann. Invariant object recognition with discriminant features based on local fast-fourier mellin transform. In *International Conference on Pattern Recognition*, vol. 1, (2000).

21. S. Raman and U. Desai. 2-d object recognition using Fourier Mellin transform and a MLPnetwork. In *IEEE International Conference on Neural Networks 1995 Proceedings*, vol. 4, pp. 2154–2156 (May, 1995).

22. F. Ghorbel. A complete invariant description for gray-level images by the harmonic analysis approach. In *Pattern Recognition Letters*, vol. 15, pp. 1043–1051 (October, 1994).

23. I. T. Jolliffe, *Principal Component Analysis.* 2002.

24. R. C. Gonzales and R. E. Woods, *Digital Image Processing.* 1992.

25. R. Stasinski and J. Konrad, A new class of fast shape-adaptive orthogonal transforms and their application to region-based image compression, *IEEE Trans. Circuits Syst. Video Technol.* **9**, 16–34 (February, 1999).

Index