EFFICIENT NEAR-DUPLICATE IMAGE DETECTION BY LEARNING FROM EXAMPLES

Yang Hu[1]*, *Mingjing Li*[2], *Nenghai Yu*[1]

[1]MOE-Microsoft Key Lab of MCC and Department of EEIS
University of Science and Technology of China, Hefei 230027, China
[2]Microsoft Research Asia, 49 Zhichun Road, Beijing 100190, China

ABSTRACT

In this paper, we propose a novel scheme for near-duplicate image detection, which is an important problem in variety of applications. While in general content based image retrieval, an image could be similar to the query image in infinitely various ways, the ways in which near-duplicate images deviate from the reference image are very limited. Based on this observation, we proposed to use examplar near-duplicate images, which can be obtained automatically, to improve the performance of near-duplicate image retrieval. We first use examplar near-duplicates to learn an effective distance measure and incorporate the learned metric into locality-sensitive hashing to achieve fast retrieval. We then use examplar near-duplicates to automatically expand the query to further improve the retrieval accuracy. The experimental results validate the effectiveness of the proposed algorithms.

Index Terms— Near-duplicate detection, metric learning, LSH, query expansion

1. INTRODUCTION

Near-duplicate image detection has become increasingly important with the proliferation of digital images and their widespread distribution over the Internet. This technique can be applied to detect copyright infringement, as an effective alternative to watermarking. It can also be used to filter spam and illicit content, such as pornography images. Besides, near-duplicate image detection is a vital component for image search engines so that they can discard the duplicate images while indexing to save storage space. It is also reasonable for image search engines to group the near-duplicate images together in the search result so as to improve users' browsing experience.

The wide applications of near-duplicate detection have stimulated much research on this problem in recent years. Both global statistics of the images [1, 2, 3] and local descriptors, such as SIFT [1, 4, 5] have been exploited for this task. Most previous work cast near-duplicate detection into the traditional content-based image retrieval (CBIR) context, which seeks to find relevant or similar images of the query image. While an image could be similar to the query image in infinitely various ways, the ways in which near-duplicate images deviate from the reference image are very limited. The common transformations that could produce near-duplicate images include hue shift, contrast change, resizing, rotation, chopping, framing, format changing, etc. It is this character that differentiates near-duplicate detection from general similarity search problems. And it is also this character that makes additional information available for us to exploit.

The relatively limited transformations of near-duplicate images make it possible for us to actively generate a set of images to imitate existing near-duplicate images for which we intend to search. In this work, we propose to learn from these examplar near-duplicate images and use the model learned to improve near-duplicate retrieval. In most previous systems, in order to realize efficient detection of near-duplicates in a large database, approximate similarity search schemes, such as locality-sensitive hashing (LSH), were employed to index the data. However, the index structures were usually built in the original feature space [1, 3], in which the Euclidean distance between feature vectors may not accurately reflect the true relationships between near-duplicate and non-duplicate images. In this work, we propose to use a set of examplar near-duplicates to learn a more effective similarity measure and incorporate the learned metrics into the approximate similarity search. Our experiments show that among the same number of approximate nearest neighbors retrieved, more near-duplicates are included in this new scheme. Besides learning a global distance function, we also use the examplar near-duplicates in a query sensitive way. In most CBIR systems, relevance feedback is usually introduced to improve retrieval accuracy. However, in order to obtain the relevance judgements, users are usually required to label the initial search results. For near-duplicate detection, we can get relevant examples without human labor. In this paper, we show how some automatically generated near-duplicate images can be used to expand the query and then help the retrieval of other near-duplicates in the database.

The remainder of the paper is organized as follows. We introduce the metric learning algorithm in Section 2. The LSH scheme with the learned metric is presented in Section 3. The

*This work was performed at Microsoft Research Asia.

retrieval with automatic query expansion is described in Section 4. We discuss the experimental results in Section 5. Finally, we conclude the paper in Section 6.

2. DISTANCE LEARNING USING RELEVANT COMPONENT ANALYSIS

The success of a distance-based near-duplicate image detection system depends critically on the quality of the distance metric. A typical distance, such as Euclidean, is affected by all the variations maintained in the data representation. However, to obtain an accurate detection, we would like to ignore the variations between images that are near-duplicate of each other. To achieve this, we could learn, from groups of automatically obtained near-duplicates, the characteristics of these within near-duplicate variations and then modify the distance function to reduce their effect.

Given a set of images that are near-duplicates of each other, which we call as a chunklet, we can use the covariance matrix of their feature vectors to characterize the variations between them. And the average of the covariance matrices of multiple chunklets could characterize the overall feature variations caused by the transformations that generate the near-duplicates. Taking the inverse of the average covariance matrix as the Mahalanobis matrix in the Mahalanobis distance, we would assign low weights to the directions in which the variations are mainly due to the image transforms. Therefore, the distance between near-duplicates would be smaller under this measurement. With this idea, we come to the Relevant Component Analysis (RCA) of Bar-Hillel et al. [6].

Specifically, given a training set $X = \{\mathbf{x}_i\}_{i=1}^{N}$ of k chunklets $\mathcal{C}_j = \{\mathbf{x}_{ji}\}_{i=1}^{n_j}$, which are k sets of near-duplicates obtained by applying a set of common transformations to k reference images, RCA computes the chunklet scatter matrix:

$$\hat{C} = \frac{1}{N}\sum_{j=1}^{k}\sum_{i=1}^{n_j}(\mathbf{x}_{ji} - \mathbf{m}_j)(\mathbf{x}_{ji} - \mathbf{m}_j)^T \quad , \qquad (1)$$

where N is the total number of training images, $\mathbf{x}_{ji}$ is the ith image in the jth chunklet and $\mathbf{m}_j$ denotes the mean of the jth chunklet. Then we measure the Mahalanobis distance between two images by:

$$d_{\hat{C}}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T \hat{C}^{-1}(\mathbf{x}_i - \mathbf{x}_j)} \quad . \qquad (2)$$

If $\hat{C}$ is not full rank, we can precede RCA with Fisher's linear discriminant analysis to reduce the dimension of the feature space.

With this learned metric, we can measure the distance between images more effectively for this task. However, when the database is large, comparing the query image with each image from it is completely impractical. In the following, we show how to incorporate the learned metric with a popular approximate nearest neighbor search scheme, LSH, to achieve both effectiveness and efficiency.

3. LOCALITY-SENSITIVE HASHING WITH LEARNED METRIC

Locality-sensitive hashing (LSH) is a randomized algorithm that allows approximate nearest neighbors in high-dimensional space to be searched in time sublinear in the size of the database. Its main idea is to hash the data points using several hash functions so as to ensure that, for each function, the probability of collision is much higher for points which are close to each other than for those which are far apart. Then, the near neighbors of the query point can be determined by hashing the query point and retrieving points stored in buckets that contain the query [7].

In this work, we use the LSH scheme proposed by Datar et al. in [7], which is based on p-stable distribution and works for l_p norm. To incorporate the distance matric learned as above, we use the following hash function $h_{\mathbf{a},b,A}(\mathbf{x})$ to map a feature vector $\mathbf{x}$ onto the set of integers:

$$h_{\boldsymbol{\alpha},b,A}(\mathbf{x}) = \left\lfloor \frac{\boldsymbol{\alpha}^T A\mathbf{x} + b}{r} \right\rfloor \quad , \qquad (3)$$

where $\boldsymbol{\alpha}$ is a random vector with entries drawn independently from a Gaussian distribution, b is a real number chosen uniformly from the range $[0, r]$ and A is a matrix that satisfies $\hat{C}^{-1} = A^T A$. Then we can construct hash tables from a family of hash functions indexed by $\boldsymbol{\alpha}$ and b and store the feature vectors in the corresponding buckets. Given a query $\mathbf{x}_q$, we search all buckets it belongs to. For each $\mathbf{x}_i$ found in a bucket, the Mahalanobis distance between $\mathbf{x}_q$ and $\mathbf{x}_i$ is computed according to Eq.(2), and only those whose distances from $\mathbf{x}_q$ are less than a pre-specified radius R are returned.

By tuning the random vector $\boldsymbol{\alpha}$ with A, we have integrated metric learning and approximate similarity search seamlessly. Compared with previous work [1, 3], which built the index structure in original Euclidean space, near-duplicate images are more likely to collide in the hash table in this new scheme. Therefore, we can detect near-duplicates more effectively.

4. RETRIEVAL WITH AUTOMATIC QUERY EXPANSION

Besides using the near-duplicates of multiple reference images collectively to learn a global distance function, we can also use the examplar near-duplicate images at query time to improve the near-duplicate detection for a specific query. Given a query image, we first retrieve its approximate nearest neighbors using the LSH scheme presented above. Then, instead of simply ranking the returned images according to their distances to the query, we expand the query by generating a set of near-duplicate images, and examine the returned database images using this query set. The intuition of this process is similar with introducing relevance feedback into CBIR. However, in that scenario, the relevance judgements are usually obtained by asking users to label the initial search

Transforms	Num	Transforms	Num
Colorizing	3	Changing contrast	2
Chopping	4	Despeckling	1
Downsampling	7	Changing format	1
Framing	4	Rotating	3
Scaling	6	Changing saturation	5
Changing intensity	4		

Table 1. Transforms used to generate near-duplicate images.

result. In our task, the relevant images are generated automatically. No additional supervision is needed.

Generally speaking, with the examplar near-duplicate images, we can then apply any algorithm proposed before that uses positive examples to improve retrieval accuracy. For instance, we can even train a one-class SVM classifier for this query. In this work, however, we use a simple criteria to rerank the images. Given a query set Q which is composed of a set of near-duplicate images, the distance between image $\mathbf{x}_i$ and Q is defined by

$$d(\mathbf{x}_i, Q) = \min_{\mathbf{x}_j \in Q} \{d_{\hat{C}}(\mathbf{x}_i, \mathbf{x}_j)\} \ , \tag{4}$$

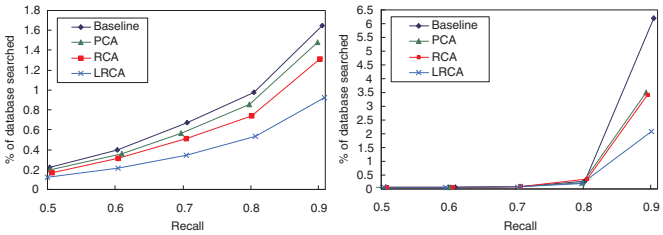
where $d_{\hat{C}}(\mathbf{x}_i, \mathbf{x}_j)$ is the Mahalanobis distance of Eq.(2).

5. EXPERIMENTAL RESULTS

5.1. Dataset

To evaluate the performance of the proposed schemes, we use images from the MM270K dataset, which was used in [4] and is publicly available. It contains about 18,000 images with very diverse content, such as animals, landscapes, peoples etc. The near-duplicate images are obtained by applying the identical transforms that are described in [3, 4]. We list the transforms and the number of near-duplicate images generated by each operation in Table 1. The details of the transforms can be found in [3, 4].

Just like previous works [1, 2, 3], we first use a set of global statistics to describe the images, which include correlograms in HSV color space, color moments and texture moments. A total of 64 dimensional features are extracted for each image. We refer to these features as Combined64. Besides, it has been shown in [8] that the simple gray block feature is very effective for near-duplicate detection. It is obtained by first uniformly dividing the image into $n \times n$ blocks and then taking the average luminance of each block as a component of the feature vector. Each component captures the local statistics of the corresponding region. We use 8×8 gray block, which results in a 64 dimensional feature vector, as our second image descriptor in the experiment. Instead of concatenating these two features into one single feature vector, we examine how the proposed algorithms work under different kinds of features.



(a) Combined64 (b) Gray block

Fig. 1. Comparison of the number of approximate near neighbors that should be returned by LSH so that a certain percentage of near-duplicate images are included.

5.2. Results

We first assess the performance of LSH with learned metric. For this experiment, 2,000 images are randomly selected from the dataset. For each image, we generate 40 near-duplicate images using the transforms in Table 1. Each original image together with its near-duplicates constitute a chunklet. We use these 2,000 chunklets to learn a new distance function. To examine how the learned metric can help the retrieval of near-duplicates for new images, another 500 images are random picked from the dataset as query images. The remaining images that are not used neither for metric learning nor for querying constitute the background dataset. For each query image, we also generate its 40 near-duplicates, yielding 20,000 images in total. These images are combined with the background images to give a set with about 36,000 images, in which we retrieve near-duplicates for each query image.

We compare four different LSH schemes. We take the LSH with Euclidean distance in the original feature space as the baseline, which is the LSH scheme adopted by [1, 3]. Compared with it is LSH with Mahalanobis distance, where the Mahalanobis matrix is learned by applying RCA to the chunklets of near-duplicate images. As the third scheme, we apply PCA to the training images in original feature space and measure Euclidean distance in the reduced space. This scheme was used in [2, 8] for indexing. We preserve 95% of the energy in the principal space, which results in 36 dimensions for Combined64 and 38 dimensions for gray block feature in the principal space respectively. The counterpart of PCA is the scheme that precede RCA with Fisher's linear discriminant analysis, which we referred to as LRCA. The dimension of the reduced space is kept the same as the corresponding PCA.

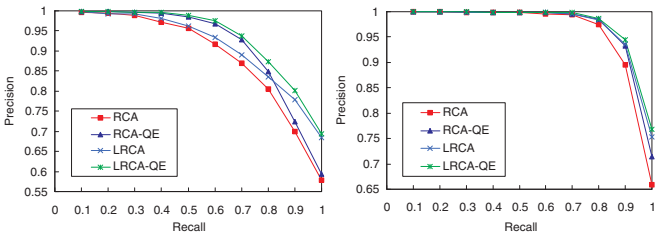
We compare the numbers of approximate nearest neighbors that should be returned by the LSH schemes so that a certain percentage of near-duplicate images are included. The less approximate nearest neighbors that should be retrieved, the more efficient the LSH scheme is. The comparisons are shown in Fig.1. According to the figures, LSH built in original feature space with Euclidean distance measure is least effi-

cient. By applying PCA, some noise are removed and a more compact representation is obtained, which results in more efficient indexing compared with the baseline. However, as an unsupervised dimension reduction method, PCA fails to employ the near-duplicate relations between images. RCA, on the other hand, uses the equivalence relations between near-duplicate images and achieves better performance. By preceding RCA with LDA, we can get the same recall by examining only half the number of near neighbors that are needed to be browsed by the baseline.

The curves in the two figures have different characteristics. While the curves in Fig.1(a) are relatively smooth, there is an abrupt increase in the curves in Fig.1(b). The gray block feature can be regarded as a thumbnail of the image. It is quite robust to many transforms such as scale and color changes. Therefore, we can get 80% of the near-duplicates by browsing a very small amount of images even using the baseline. However, this feature is quite sensitive to some other transforms, such as chopping, rotation and framing operations. To include the near-duplicates under these transforms, a large amount of near neighbors should be returned by the LSH scheme that use Euclidean distance in the original feature space. The learned metric greatly reduced this limitation of gray block feature. The Combined64 and the gray block features are actually quite complementary. We can achieve extremely good performance by combining them together. For example, applying LRCA to the combined 128 dimensional feature, we can find 90% near-duplicate images by browsing 0.1% of the database. Due to the limit of space, we can not show the details of the results using the combined feature. Also, instead of focus on choosing the best features for the task, in this work, we just would like to show the proposed LSH scheme works using different kinds of features.

In the second experiment, we show how the automatic query expansion can further improve the retrieval accuracy. In this experiment, we randomly select 10 near-duplicates of each query, yielding a set of 5000 images. These images are combined with the background images to give a set with about 21,000 images, in which we search near-duplicates for each query image.

In Fig.2, we show the average precision/recall curves of the results. The baseline RCA and LRCA results are obtained by first quering the dataset using the LSH schemes with RCA and LRCA respectively. The images returned by LSH are then ranked according to their Mahalanobis distances to the query image. For LSH, we choose R that achieves 90% overall recall in the first experiment. For the schemes denoted by "QE", after getting the approximate nearest neighbors using LSH, we first expand each query using the 30 near-duplicates that are not included in the target dataset. The images returned by LSH are then ranked according to their distance to the query set, which is defined by Eq.(4). We can see from Fig.2 that we can achieve higher retrieve accuracy by expanding the queries.



(a) Combined64 (b) Gray block

Fig. 2. Comparison of the retrieval accuracy with and without automatic query expansion.

6. CONCLUSION

In this work, based on the characteristics of near-duplicate detection problem, we proposed to use examplar near-duplicate images, which can be obtained automatically, to improve the performance of near-duplicate image retrieval. The examplar near-duplicates have been used in two different ways. First, they are used to learn a more effective distance metric, which is then incorporated into LSH to achieve more efficient indexing. Second, they are used to automatically expand the query image to further improve the retrieval accuracy. The experimental results validate the effectiveness of our algorithms.

ACKNOWLEDGEMENT

The research is supported in part by National Natural Science Foundation of China(60672056) and Specialized Research Fund for the Doctoral Program of Higher Education (20070358040).

7. REFERENCES

[1] O. Chum, J. Philbin, M. Isard, and A. Zisserman, "Scalable near identical image and shot detection," in *CIVR*, 2007.

[2] Y. Maret, S. Nikolopoulos, F. Dufaux, T. Ebrahimi, and N. Nikolaidis, "A novel replica detection system using binary classifiers, r-trees, and pca," in *ICIP*, 2006.

[3] L. Qamra, Y. Meng, and E. Chang, "Enhanced perceptual distance functions and indexing for image replica recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, 2005.

[4] Y. Ke, R. Sukthankar, and L. Huston, "Efficient near-duplicate detection and sub-image retrieval," in *ACM Multimedia*, 2004.

[5] E. Valle, M. Cord, and S. Philipp-Foliguet, "3-way trees: A similarity search method for high-dimensional descriptor matching," in *ICIP*, 2007.

[6] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning distance functions using equivalence relations," in *ICML*, 2003.

[7] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *SCG*, 2004.

[8] B. Wang, Z.-W. Li, M.-J. Li, and W.-Y. Ma, "Large-scale duplicate detection for web image search," in *ICME*, 2006.