Improving Similarity Search in Face-Images Data

Pedro Chambel[1], Fernanda Barbosa[1]

[1] FCT-Universidade Nova de Lisboa, Portugal
{pms17299@fct.unl.pt, fb}@di.fct.unl.pt

Abstract. Similarity search involves finding all the face-images in a database, which are similar to a desired face-image, based on some distance measure. Comparing the desired face-image to all the face-images in a large dataset is prohibitively slow. If face-images can be placed in a metric space, search can be sped up by using a metric data structure. In this work, we evaluate the performance of range queries with metric data structures (LAESA, VPtree, DSAT, HDSAT1, HDSAT2, LC, RLC and GNAT) when the metric spaces are face-images data with the Euclidean distance. The experimental results show that all data structures reduce the ratio between the number of distances computed and the database size. Moreover, the LAESA has the best performance in the majority of the experimental cases, but the RLC competes with the other metric data structures, and has the best results when compared with the other dynamic metric data structures.

Keywords: similarity search; metric data structures; image processing

1 Introduction

With the rapid increase in the use of digital technology, large amounts of image data sets will soon be accumulated. Face-images browsing is based on the concept of similarity between face-images, i.e. searching face-images which are very similar or close to a given face. Each face-image is represented as a vector of numeric properties (features) extracted from the contend-based image. The similarity between two images is associated with a function, which measures the distance between their feature vectors. When this function is metric, the set of face-images defines a metric space.

Most of the actual work in face-images is about face detection and face recognition. In these works, the similar searching for a given face-image leads to an exhaustive search in the dataset, so the response time will be very long and the search will became ineffective. For this reason, it is necessary to introduce new techniques that can deal with this problem efficiently.

In order to have efficient similar searching in metric spaces, several metric data structures have been proposed [1,2]. These data structures partition the database based on the distances between a set of selected objects and the remaining objects. Space partitions seek to minimize the exhaustive search, i.e. at search time, some subsets are discarded and others are exhaustively searched. The distance-based indexing method

may be pivot-based or cluster-based [1]. Some of the data structures using the pivot-based method are the VP-Tree [3] and the MVP-Tree [4]. There are variants of the pivot-based method, used in LAESA [5]. Some of the data structures using cluster-based method are the GNAT [6], the HDSAT [7], the LC [8] and the RLC [9]. The RLC was already evaluated in different application domains [10, 11, 12].

Our main goal is to evaluate the use and the efficiency of similar searching with metric data structures in face-images databases with Euclidean Distance. This study involves four databases of face-images: Faces94 [13], Jaffe [14], Yalefaces [15] and AT&T [16], and comprises 8 metric data structures: LAESA, VPtree, DSAT, HDSAT1, HDSAT2, LC, RLC and GNAT.

The rest of the paper is structured as follows. In Section II, we recall some basic notions on range query in metric spaces. Section III is devoted to the characterization of the metric space over face-images data. Then Section IV reports on the experimental results. Conclusions and future work are drawn in Section V.

2 Range Queries in Metric Spaces

A metric space is a pair (U,d), where U is a set of objects, called the universe, and d: U x U $\Rightarrow \Re^+_o$ is a function, called distance, that satisfies the three following properties: (1) strict positiveness: $d(x,y) \geq 0$ and $d(x,y) = 0 \Leftrightarrow x = y$; (2) symmetry : $d(x,y) = d(y,x)$ and (3) triangle inequality: $d(x,y) \leq d(x,z) + d(z,y)$.

A database over a metric space (U,d) is a finite set $B \subseteq U$. The problem raised by range queries is to yield the set of all database objects whose distance to a given object does not exceed a certain amount. Formally, given a database B over a metric space (U,d), a query point $q \in U$, and a query radius $r \in \Re^+$, the answer to the range query (q,r) is the set $\{x \in B \mid d(x,q) \leq r\}$.

Metric data structures seek to minimize the number of distance computations performed in range queries. During the computation of a range query (q,r) in a database over a metric space (U,d), triangle inequality and symmetry are used to discard elements of the database without computing the associated distance to the query object. Given a query element q and a query radius r, an element x may be left out without the need for evaluating d(q,x). This will arise if there is an object o where $|d(q,o) - d(x,o)| > r$. In this case, it is not necessary to compute d(q,x) since we know that $d(q,x) > r$, based on the triangle inequality.

It is important to remark that the range queries are hard to compute in high dimension metric space [2]. It is well known that the metric space dimension grows with the mean and decreases with the variance.

3 Face-Images Metric Space

Our experiments involve four face-images databases, which were already used in related works [17, 18, 19]. The databases used are:

- Faces94 – This database is available in [13]. The images are stored in 24 bit RGB, JPEG format and the size of each image is 180 x 200 pixels.

- Jaffe – This database is available in [14]. The images are stored in TIFF format and the size of each image is 256 x 256 pixels.
- Yalefaces- This database is available in [15]. The images are stored in GIF format and the size of each image is 320 x 243 pixels.
- AT&T- This database is available in [16]. The images are stored in PGM and the size of each image is 92x112 pixels, with 256 grey levels per pixel.

In these databases, all the images are frontal face-images of different individuals in different facial expressions. In table 1 we present the size of each database.

3.1 Face-Image Representation

Each face-image is represented by a feature vector, which describes the face according to a training set of face-images. The method used to extract this feature vector was based on the method of Eigenfaces. The steps involved in creating a set of eigenfaces are:

- Define a training set: each image is seen as one vector, simply by concatenating the rows of pixels in the image. So an image with r rows and c columns is therefore represented as a vector with r x c elements. All images in the training set are stored in a single matrix T, where each row is an image;
- Calculate the average image and subtract it from each image in T;
- Calculate the eigenvectors and eigenvalues of the covariance matrix S. The eigenvectors of this covariance matrix are called eigenfaces;
- Choose the principal components by keeping the eigenvectors with the largest associated eigenvalue.

The fundamental idea of this method is to project the face-images on the eigenfaces created. So, each face-image is a vector of features $S = <f_1, \ldots, f_n>$, where n is less than or equal to the size of the training database. In our experiment, the size of the training set is 25 face-images. For further details of the method of Eigenfaces, the reader is referred to [20].

3.2 Similarity between Face-Images

In our experiment, the similarity between two face-images is based on the similarity between the associated feature vectors, which is computed with the Euclidean distance. Let S = <f1s, …, fns> and T = <f1t, …, fnt)> be feature vectors associated with two face-images S and T. The Euclidean distance between S and T, denoted by ED(S,T), is defined by:

$$ED(S,T) = \sqrt{\sum_{i=1..n} (f_{is} - f_{it})^2}. \quad (1)$$

In order to study the metric spaces, we have computed the histogram of distances between any two face-images of each database. In Table 1, we present the mean and the variance of the histogram of distances for each database.

An immediate conclusion is that our metric spaces have different dimensions. The dimension is highest for the Jaffe database, where the quotient between the mean and the variance is 2.7710. The other metric spaces have lowest dimension.

Table 1. Database Size and Mean and Variance of the Histogram of Distances

Database	Size(images)	Mean	Variance	Mean/Variance
Faces94	3040	9137.49	7936650.53	0.0012
JAFFE	213	10749.96	15050638.74	2.7710
Yalefaces	165	19293.42	37809562.91	0.0005
AT&T	400	3008.78	617807.58	0.0049

3.3 Evaluation of Euclidean Distance in the Databases

The Euclidean distance was evaluated in the 4 databases with the ROC (Receiver Operating Characteristics) analysis, based on the true positive rate and the false positive rate. In all the databases, the results show that the false positive rate (irrelevant results) is less than 0.1 and the true positive rate (relevant results) is bigger than 0.5. The Euclidean distance shows better results with the Faces94 database and worse results with the Yalefaces database. For lack of space, this evaluation can not be presented here.

4 Evaluation of the Metric Data Structures

The goal of this section is to evaluate the behavior of range queries with metric data structures (LAESA, VPTree, DSAT, HDSAT1, HDSAT2, LC, RLC and GNAT) over face-images data with the Euclidean distance.

For each database, four files were generated. The smallest was used as the query set of faces and is composed by random faces from the database. In order to maintain the same number of different individuals in this set, we choose randomly 25% of the face-images associated to each individual. Table 2 presents the size of the query set for each database. The other three files are random permutations of the database. The justification for making use of three equal sets lies on the fact that the final shape of some data structures depends on the order in which the objects occur in the input of the construction algorithm.

In order to compute the range queries with the metric data structures we choose two different query radii for each database. The first one is 25% and the second is 50% of the database distances mean (see Table 2).

Table 2. Size of the Query Set and Query Radii for each Database

Database	Size(query set)	1st Query Radius	2nd Query Radius
Faces94	760	2284	4569
JAFFE	50	2687	5374
AT&T	80	752	1504
Yalefaces	30	4823	9646

For each database, we submitted the set of query faces associated with the database with the two radii. In Table 3, we present the average number of face-images retrieved in range queries for each query, and the associated percentage of the database size.

Table 3. Average number of face-images retrieved in range queries, and the associated percentage of the database size

Database	1st Query Radius		2nd Query Radius	
	Num	Percent	Num	Percent
Faces94	15.94	0.52%	109.79	3.61%
JAFFE	6.76	3.17%	21.94	10.3%
AT&T	2.35	0.59%	12.83	3.21%
Yalefaces	3.7	2.24%	11.43	6.93%

In each experimental case (a database, a query set and a query radius), we compute the average number of distance computations done for each face-image query. So, the results presented are the mean of the results obtained to query the three sets associated to the database.

4.1 Metric Data Structures Parameterizations

The eight metric data structures were parameterized in order to obtain the best results for each database:

- LAESA: The Linear Approximating and Eliminating Search Algorithm was parameterized with 44 prototypes for Faces94, 8 prototypes for JAFFE, 22 prototypes for AT&T and 23 prototypes for Yalefaces.
- VP-tree: This data structure does not have parameters.
- DSAT: The Dynamic Spatial Approximation Tree was parameterized with arity 3 for Faces94, Jaffe and Yalefaces, and with arity 5 for AT&T.
- HDSAT1: The Hybrid Dynamic Approximation Tree 1 was parameterized with arity 5 for Faces94, 6 for JAFFE and AT&T, and 9 for Yalefaces.
- HDSAT2: The Hybrid Dynamic Approximation Tree 2 was parameterized with arity 8 for Faces94, 6 for JAFFE and AT&T and 9 for the Yalefaces.
- LC: The List of Clusters we was parameterized with cluster radius 2995 for Faces94, 7130 for JAFFE, 1400 for AT&T and 8795 for Yalefaces.
- RLC: The Recursive List of Clusters was parameterized with cluster radius 3575 and array capacity 20 for Faces94, cluster radius 6750 and array capacity 11 for JAFFE, cluster radius 1840 and array capacity 9 for AT&T and cluster radius 9450 and array capacity 11 for Yalefaces.

4.2 Experimental Results

Figures 1 and 2 show, for each database, the average number of distance computations done with each query with the two query radii.

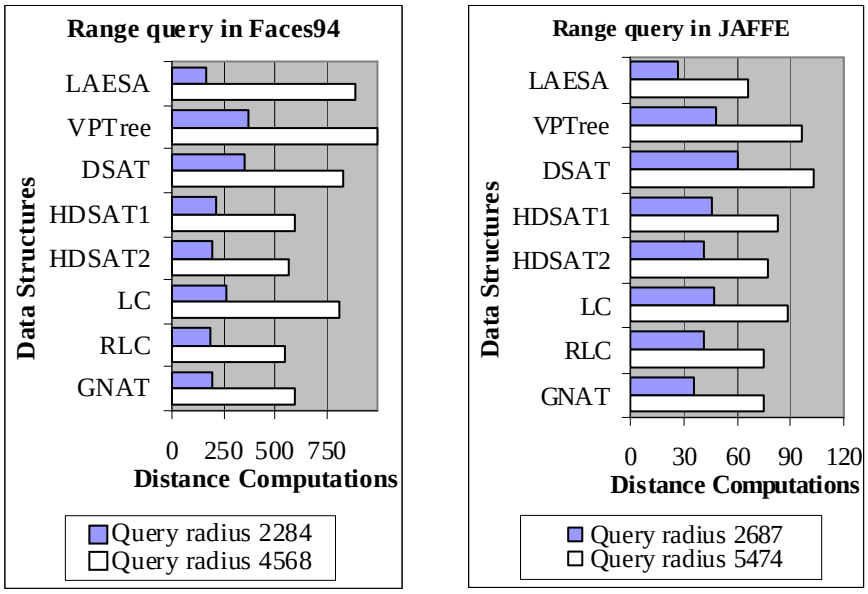

Fig. 1. Average number of distance computations in Faces94 and JAFFE databases

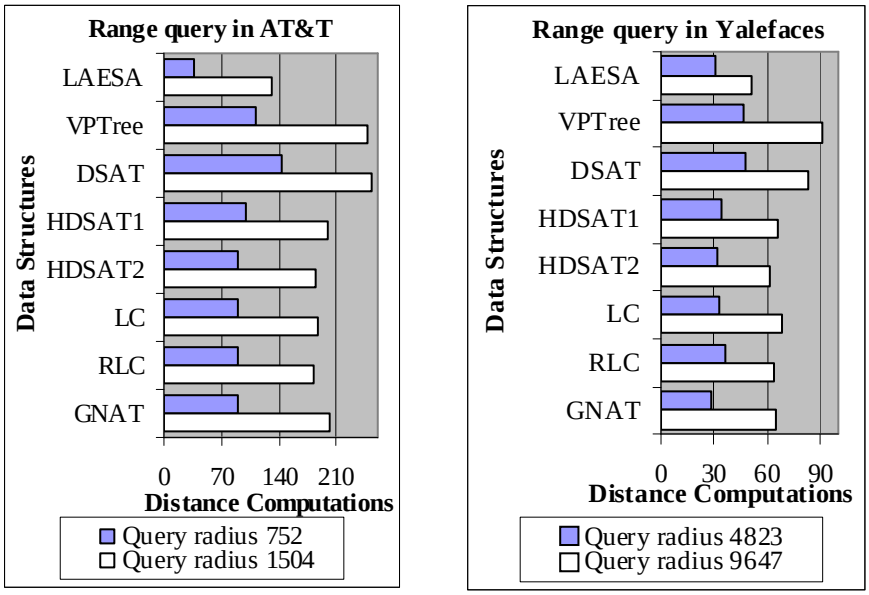

Fig. 2. Average number of distance computations in AT&T and Yalefaces databases

These graphics show that LAESA outperforms all the data structures, except in the Faces94 database with query radius 4569, where RLC achieves the best result, and in Yalefaces database with radius 4823, where the GNAT is the best.

When we compare the four dynamic metric data structures (DSAT, HDSAT1, HDSAT2 and RLC) we conclude that, in the majority of the experimental cases (5 of 8 cases), RLC is the data structure with the best results.

The results from this experiment show that all the metric data structures reduce the ratio between the number of distances computed and the size of the database. Table 5 presents the percentage for each data structure and each database.

Table 5. The Percentage of Distance Computations According to the Database Size

Database	Faces94		JAFFE		AT&T		Yalefaces	
Query radius	2284	4568	2687	5474	752	1504	4823	9646
LAESA	5.38%	29.25%	12.39%	31%	8.86%	32.76%	18.52%	30,77%
VPTree	12.14%	32.71%	22.58%	45.4%	27.68%	62.01%	28.61%	54,67%
DSAT	11.72%	27.36%	28.67%	48.6%	35.96%	63.01%	29,04%	50,06%
HDSAT1	7.25%	19.5%	21.33%	38.91%	24.86%	49.6%	20.78%	39,7%
HDSAT2	6.57%	18.6%	19.6%	36.29%	22.63%	46.26%	19.25%	36,75%
LC	8.57%	26.76%	22.3%	41.71%	22.56%	46,7%	19,98%	41,23%
RLC	6.27%	18.17%	19.55%	35.48%	22.67%	45,83%	22,27%	38.76%
GNAT	6.5%	19.46%	16.64%	35.38%	22.44%	50.72%	17,44%	39.03%

5 Conclusion and Future Work

The need to speed the similar searching of face-images leads us to evaluate the performance of range queries with several metric data structures over face-images data. It is important to remark that there are few works which compare different techniques in the face-image domain.

With respect to the face-images data representation and the Euclidean distance (our metric space), we have good similar search results in these databases. However, we need to compare these results with the results obtained in face-images databases, where the data are described by local features (i.e., noise, eyes) and/or with others metric functions.

With respect to the efficiency of the range queries with metric data structures, the results leave us to conclude that the metric data structures speed the range query in the four databases. This conclusion is based on the observation that a lot of face-images are discarded without computing the associated distance to the face4-image query. The LAESA data structure has the best performance in all the databases, except in two experimental cases. In the majority of the experimental cases, the RLC data structure competes with the other metric data structures, and it is the best dynamic metric data structure in the majority of the experimental cases.

References

1. Samet, H.: Foundations of Multidimensional and Metric Data Structures. Morgan Kaufmann, San Francisco (2006)
2. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.: Searching in metric spaces. ACM Computing Surveys, 33(3), 273—321 (2001)
3. Yianilos, P.: Data structures and algorithms for nearest neighbor search in general metric spaces. In: 4th Annual SIAM Symposium on Discrete Algorithms, pp. 311—321, ACM Press, USA (1993)
4. Bozkaya, T., Ozsoyoglu, M.: Distance-based indexing for high-dimensional metric spaces. In: SIGMOD International Conference on Management of Data (SIGMOD'97) , pp. 357—368, ACM Press, New York (1997)
5. Micó, M., Oncina, J., Vidal, E.: A new version of the nearest-neighbour approximating and eliminating search algorithm. Pattern Recognition Letters, 15(1), 9—17 (1994)
6. Brin, S.: Near neighbor search in large metric spaces. In: 21st International Conference on Very Large Data Bases (VLDB'95), pp. 574—3584, Morgan Kaufmann, Zurich, Switzerland (1995)
7. Arroyuelo, D., Muñoz, F.; Navarro, G., Reyes, N.: Memory-adaptative dynamic spatial approximation trees. In: the 10th International Symposium on String Processing and Information Retrieval (SPIRE) (2003)
8. Chávez, E., and Navarro, G.: A compact space decomposition for effective metric indexing. Pattern Recognition Letters, 26(9), 1363—1376 (2005)
9. Mamede, M.: Recursive Lists of Clusters: a dynamic data structure for range queries in metric spaces. In 20th International Symposium on Computer and Information Sciences (ISCIS 2005) , pp. 843—853, Springer-Verlag, Berlin, Germany (2005)
10. Mamede, M., Barbosa, F.: Range Queries in Natural Language Dictionaries with Recursive Lists of Clusters. In: 22th International Symposium on Computer and Information Sciences, IEEE Xplore, doi:10.1109/ISCIS.2007.4456857 (2007)
11. Barbosa, F.: Similarity-based retrieval in high dimensional data with Recursive Lists of Clusters: a study case with Natural Language Dictionaries. In: International Conference on Information management and engineering (ICIME 2009), IEEE Computer Society, ISBN: 978-1-4244-3774-0 (2009)
12. Barbosa, F., Rodrigues, A.: Range Queries over Trajectory Data with Recursive List of Clusters: a case study with Hurricanes data. In: *Geographic Information Science Research UK (GISRUK 2009)*, UK (2009)
13. Faces94 database : http://cswww.essex.ac.uk/mv/allfaces/
14. Jaffe database: http://www.kasrl.org/jaffe.html
15. Yalefaces database: http://cvc.yale.edu/projects/yalefaces/yalefaces.html
16. AT&T database: http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
17. Allam, H., Rahman, F., Tarnikova, Y., Hartono, R.: A Pair-wise Decision Fusion Framework: Recognition of Human Faces. In: International Conference on Information Fusion (ISIF 2003),pp. 1484-1489, IEEE Xplore, ISBN: 0-9721844-4-9 (2003)
18. Lyons, M., Akamatsu S.: Coding Facial Expressions with Gabor Wavelets. In: 3rd IEEE International Conference on Automatic Face and Gesture Recognition (AFGR 1998), pp. 200—205, IEEE Xplore, ISBN: 0-8186-8344-9 (1998)
19. Aly M.,: Face Recognition using SIFT Features. Available on http://www.vision.caltech.edu/malaa/research/aly06face.pdf
20. Turk M., Pentland A.: Face Recognition using Eigenfaces. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 586—591 (1991)