Dimension Reduction Methods for Iris Recognition

Pavel Moravec and Václav Snášel

Department of Computer Science, FEECS, VŠB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
{pavel.moravec, vaclav.snasel}@vsb.cz

Abstract. In this paper, we compare performance of several dimension reduction techniques, namely LSI, FastMap, and SDD in Iris recognition. We compare the quality of these methods from both the visual impact, and quality of generated "eigenirises".

Keywords: SVD, FastMap, information retrieval, SDD, iris recognition

1 Introduction

Methods of human identification using biometric features like fingerprint, hand geometry, face, voice and iris are widely studied.

A human eye iris has its unique structure given by pigmentation spots, furrows and other tiny features which are stable throughout life. It is possible to scan an iris without physical contact in spite of wearing contact lenses or eyeglasses. The iris is hard to forge which makes the iris a suitable object for the identification of people. Iris recognition seems to be more reliable than other biometric techniques like face recognition[3]. Iris biometrics systems for both private and public use have been designed and deployed commercially by NCR, Oki, IriScan, BT, US Sandia Labs, and others.

In this paper, we use the Petland's approach to image retrieval: image vectors of complete images of the size width $\times$ height of the image [13] build the feature vectors.

To fight the high dimension of image vector, we can extract several *features* which represent the image and concatenate them into a *feature vector*. The feature extraction methods can use different aspects of images as the features, typically the color features (histograms), shape features (moments, contours, templates), texture features and others (e.g. eigenvectors). Such methods are either using a heuristics based on the known properties of the image collection, or are fully automatic and may use the original image vectors as an input.

In this paper we will concentrate on the last category – other feature extraction methods which use known dimension reduction techniques and clustering for automatic feature extraction.

Singular value decomposition (*SVD*) was already successfully used for automatic feature extraction. In case of face collection (such as our test data), the *base vectors* can be interpreted as images, describing some common characteristics of several faces. These base vectors are often called eigenfaces. For a detailed description of eigenfaces, see [13].

K. Richta, J. Pokorný, V. Snášel (Eds.): Dateso 2009, pp. 80–89, ISBN 978-80-01-04323-3.

However SVD is not suitable for huge collections and is computationally expensive, so other methods of dimension reduction were proposed. We test two of them – *Semi-Discrete decomposition*.

Recently, Toeplitz matrix minimal Eigenvalues are also playing a role towards image description and feature extraction [10]. This approach presents a method for the reduction of image feature points as it deals with the geometric relation between the points rather than their geometric position [11]. This can reduce the characteristic or feature points number from n to at least $\frac{n}{10}$ decreasing the computation level and hence the task time.

The rest of this paper is organized as follows. The second section explains used dimension reduction methods. In the third section, we briefly describe qualitative measures used for evaluation of our tests. In the next section, we supply results of tests for several methods on ORL face collection. In conclusions we give ideas for future research.

2 Dimension Reduction Methods

We used three methods of dimension reduction for our comparison – Singular Value Decomposition, Semi-Discrete Decomposition and FastMap, which are briefly described bellow.

2.1 Singular Value Decomposition

SVD [2] is an algebraic extension of classical vector model. It is similar to the *Principal components analysis* (*PCA*) method, which was originally used for the generation of eigenfaces. Informally, SVD discovers significant properties and represents the images as linear combinations of the base vectors. Moreover, the base vectors are ordered according to their significance for the reconstructed image, which allows us to consider only the first k base vectors as important (the remaining ones are interpreted as “noise” and discarded). Furthermore, SVD is often referred to as more successful in recall when compared to querying whole image vectors [2].

Formally, we decompose the matrix of images A by *singular value decomposition* (*SVD*), calculating singular values and singular vectors of A.

We have matrix A, which is an $n \times m$ rank-r matrix and values $\sigma_1, \ldots, \sigma_r$ are calculated from eigenvalues (λ_i) of matrix AA^T as $\sigma_i = \sqrt{\lambda_i}$. Based on them, we can calculate column-orthonormal matrices $U = (u_1, \ldots, u_r)$ and $V = (v_1, \ldots, v_r)$, where $U^T U = I_n$ a $V^T V = I_m$, and a diagonal matrix $\Sigma = diag(\sigma_1, \ldots, \sigma_r)$, where $\sigma_i > 0, \sigma_i \geq \sigma_{i+1}$.

The decomposition

$$A = U \Sigma V^T$$

is called *singular decomposition* of matrix A and the numbers $\sigma_1, \ldots, \sigma_r$ are *singular values* of the matrix A. Columns of U (or V) are called *left* (or *right*) singular vectors of matrix A.

Now we have a decomposition of the original matrix of images A. We get r nonzero singular numbers, where r is the rank of the original matrix A. Because the singular values usually fall quickly, we can take only k greatest singular values with the corresponding singular vector coordinates and create a *k-reduced singular decomposition* of A.

Let us have k $(0 < k < r)$ and singular value decomposition of A

$$A = U\Sigma V^T \approx A_k = (U_k U_0) \begin{pmatrix} \Sigma_k & 0 \\ 0 & \Sigma_0 \end{pmatrix} \begin{pmatrix} V_k^T \\ V_0^T \end{pmatrix}$$

We call $A_k = U_k \Sigma_k V_k^T$ a k-reduced singular value decomposition (rank-k SVD).

Instead of the A_k matrix, a matrix of image vectors in reduced space $D_k = \Sigma_k V_k^T$ is used in SVD as the representation of image collection. The image vectors (columns in D_k) are now represented as points in k-dimensional space (the *feature-space*). For an illustration of rank-k SVD see Figure 1.

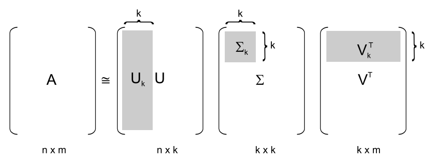

Fig. 1. rank-k SVD

Rank-k SVD is the best rank-k approximation of the original matrix A. This means that any other decomposition will increase the approximation error, calculated as a sum of squares (*Frobenius norm*) of error matrix $B = A - A_k$. However, it does not implicate that we could not obtain better precision and recall values with a different approximation.

To execute a query Q in the reduced space, we create a reduced query vector $q_k = U_k^T q$ (another approach is to use a matrix $D'_k = V_k^T$ instead of D_k, and $q'_k = \Sigma_k^{-1} U_k^T q$). Instead of A against q, the matrix D_k against q_k (or q'_k) is evaluated.

Once computed, SVD reflects only the decomposition of original matrix of images. If several hundreds of images have to be added to existing decomposition (*folding-in*), the decomposition may become inaccurate. Because the recalculation of SVD is expensive, so it is impossible to recalculate SVD every time images are inserted. The *SVD-Updating* [2] is a partial solution, but since the error slightly increases with inserted images. If the updates happen frequently, the recalculation of SVD may be needed soon or later.

2.2 SDD Method

Semidiscrete decomposition (*SDD*) is one of other LSI methods, proposed recently for text retrieval in [8]. As mentioned earlier, the rank-k SVD method (called *truncated SVD* by authors of semidiscrete decomposition) produces dense matrices U and V, so

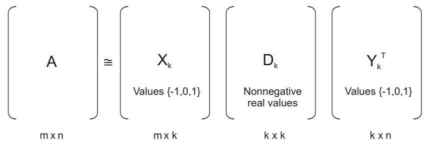

Fig. 2. "rank-k" SDD

the resulting required storage may be even larger than the one needed by the original term-by-document matrix A.

To improve the required storage size and query time, the semidiscrete decomposition was defined as

$$A \approx A_k = X_k D_k Y_k^T,$$

where each coordinate of X_k and Y_k is constrained to have entries from the set $\varphi = \{-1, 0, 1\}$, and the matrix D_k is a diagonal matrix with positive coordinates.

The SDD does not reproduce A exactly, even if $k = n$, but it uses very little storage with respect to the observed accuracy of the approximation. A rank-k SDD (although from mathematical standpoint it is a sum on rank-1 matrices) requires the storage of $k(m + n)$ values from the set $\{-1, 0, 1\}$ and k scalars. The scalars need to be only single precision because the algorithm is self-correcting. The SDD approximation is formed iteratively.

The optimal choice of the triplets (x_i, d_i, y_i) for given k can be determined using greedy algorithm, based on the residual $R_k = A - A_{k-1}$ (where A_0 is a zero matrix).

2.3 FastMap

FastMap [6] is a pivot-based technique of dimension reduction, suitable for Euclidean spaces.

In first step, it chooses two points (feature vectors) from the matrix A, which should be most distant for calculated reduced dimension. Because it would be expensive to calculate distances between all points, it uses following heuristics (all chosen points are image vectors from matrix A):

1. A random point c_0 is chosen.
2. The point b_i having maximal distance $\delta(c_i, b_i)$ from c_i is chosen, and based on it we select the point a_i with maximal distance $\delta(b_i, a_i)$
3. We iteratively repeat step 2 with $c_{i+1} = a_i$ (authors suggest 5 iterations).
4. Points $a = a_i$ and $b = b_i$ in the last iteration are pivots for the next reduction step.

In second step (having the two pivots a, b), we use the cosine law to calculate position of each point on line joining a and b. The coordinate x_i of point p_i is calculated as

$$x_i = \frac{\delta^2(a_i, p_i) + \delta^2(a_i, b_i) - \delta^2(b_i, p_i)}{2\delta(a_i, b_i)}$$

and the distance function for next reduction step is modified to

$$\delta'^2(p'_i, p'_j) = \delta^2(p_i, p_j) - (x_i - x_j)^2$$

The pivots in original and reduced space are recorded and when we need to process a query, it is projected using the second step of projection algorithm only. Once projected, we can again use the original distance function in reduced space.

3 Qualitative Measures of Retrieval Methods

Since we need an universal evaluation of any retrieval method, we use some measures to determine quality of such method. In case of Information Retrieval we usually use two such measures - *precision* and *recall*. Both are calculated from the number of objects relevant to the query Rel – determined by some other method, e.g. by manual annotation of given collection and the number of retrieved objects Ret. Based on these numbers we define precision (P) as a fraction of retrieved relevant objects in all retrieved objects and recall (R) as a fraction of retrieved relevant objects in all relevant objects. Formally:

$$P = \frac{|Rel \cap Ret|}{|Ret|} \text{ and } R = \frac{|Rel \cap Ret|}{|Rel|}$$

So we can say that recall and precision denote, respectively, completeness of retrieval and purity of retrieval. Unfortunately, it was observed that with the increase of recall, the precision usually decreases [12]. This means that when it is necessary to retrieve more relevant objects, a higher percentage of irrelevant objects will be probably obtained, too.

For the overall comparison of precision and recall across different methods on a given collection, we usually use the technique of rank lists [1], where we first sort the distances from smallest to greatest and then go down through the list and calculate maximal precision for recall closest to each of the 11 standard recall levels (0.0, 0.1, 0.2, . . . , 0.9, 1.0). If we are unable to calculate precision on i-th recall level, we take the maximal precision for the recalls between $i - 1$-th and $i + 1$-th level. From all levels, we calculate mean average, which is a single-value characteristics of overall precision-recall ratio.

4 Experimental Results

For testing of the different methods, we used iris collection consisting of 384 irises. The iris were scanned by TOPCON optical device connected to the CCD Sony camera. The acquired digitized image is RGB of size 576 $\times$ 768 pixels. Only the red (R) component of the RGB image has been used in our experiments because it appears to be more reliable than recognition based on green or blue components or converting the irises to grayscale first. It is in accord with [4], where near-infrared wavelengths are used anyway.

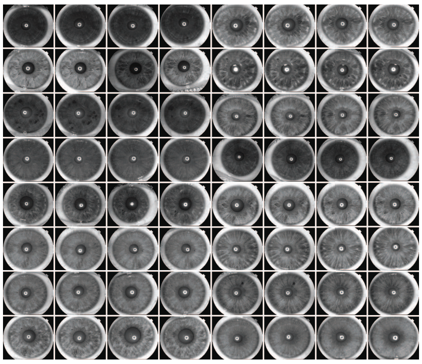

Fig. 3. Several irises from the collection

We have excluded one of the three irises for each eye for further querying (so that the query iris would not be included in the collection and skew the query results), which led to a collection of 256 irises of 64 people.

An example of several irises from the collection is shown in Figure 3, the first 12 query vectors are shown in Figure 8. We did not isolate the central part and eyelids to provide comparable results with[9].

4.1 Generated "Eigenirises" and Reconstructed Images

Many of tested methods were able to generate a set of base images, which could be considered to be "eigenirises" as is the case of PCA, SVD and several other methods. We are going to provide examples of both factors (base vectors) – "eigenirises" and reconstructed images which can be obtained from regenerated A_k. We calculated results for all methods in several dimensions, for the demonstration images we will use $k = 64$. We do not provide these images for FastMap, where it is not possible (we could have provided the images used as pivots in each step of FastMap process).

With SVD, we obtain factors with different generality, the most general being among the first. The first few are shown in figure 4. The eigenirises with higher index bring more details to reconstructed images.

The reconstructed images for rank-64 SVD method are somewhat blurred, but generally still recognizable, which can bee observed in figure 5.

The SDD method differs slightly from previous methods, since each factor contains only values $\{-1, 0, 1\}$. Gray in the factors shown in figure 6 represents 0; -1 and 1 are represented with black and white respectively.

The images in figure 7 are reconstructed least exactly from all methods (although consistently), but this is to be expected due to the three-valued encoding of base vectors. One may note a general loss of fine details, which is unfortunate, since it means that the query process would be highly affected and the retrieval results poor.

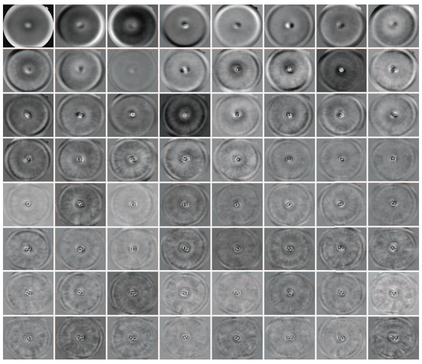

Fig. 4. First 64 eigenirises (out of possible 256) for SVD method

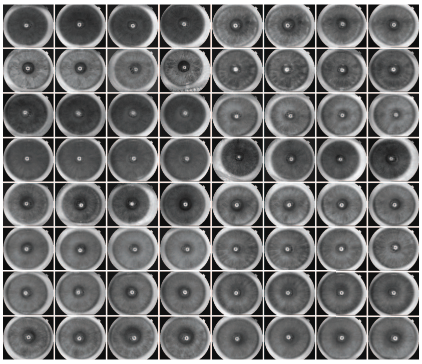

Fig. 5. Reconstructed images for SVD method

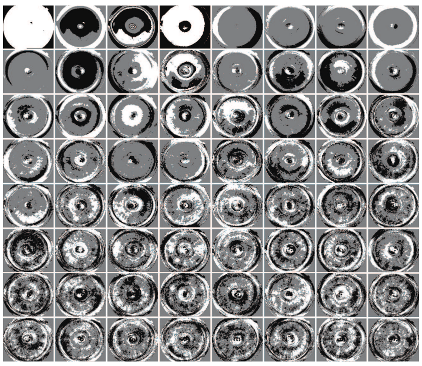

Fig. 6. First 18 base vectors (out of 100) for SDD method

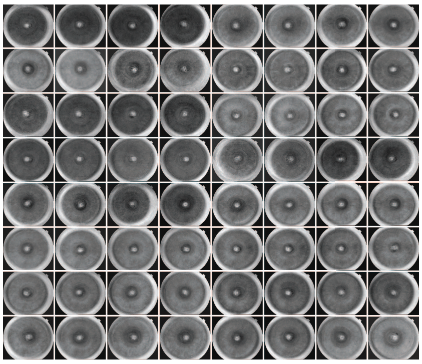

Fig. 7. Reconstructed images for SDD method

4.2 Query Evaluation

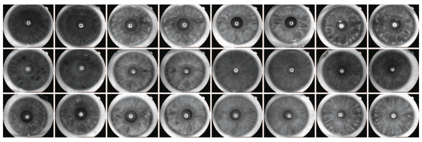

Fig. 8. Several irises used for querying

First, we calculated the mean average precision (MAP) for all relevant images in rank lists. The relative MAPs (against original matrix A – 100%) are shown in Table 1.

One would suspect, that querying in original dimension would provide better results that any of the dimension reduction methods. In Table 2 we show the number of queries, where the first returned iris was of the same person (out of 128).

5 Conclusion

In this paper, we have compared several dimension reduction methods on real-live image data (using L_2 metrics). Whilst the SVD is known to provide quality results, it is computationally expensive and in case we only need to beat the "curse of dimensionality" by reducing the dimension, FastMap may suffice.

There are some other newly-proposed methods, which may be interesting for future testing, e.g. the SparseMap [7]. Additionally, faster pivot selection technique for FastMap may be considered. We may also benefit from the use of Toeplitz matrices and their minimal eigenvalues relation.

Table 1. Mean average precision of iris comparison (VSM: 49%)

	Reduction method		
k	*FastMap*	**SVD**	SDD
4	25%	14%	3%
8	33%	31%	3%
16	39%	44%	4%
32	42%	46%	4%
64	45%	48%	5%
128	47%	49%	5%

Table 2. Number of queries, where the person was successfully identified (VSM: 83)

	Reduction method		
k	*FastMap*	**SVD**	SDD
4	32	37	2
8	47	56	3
16	54	75	3
32	60	81	5
64	64	82	4
128	96	84	5

What we currently need is a better iris segmentation, i.e. removing the central piece (in our case in light reflection), eyelids, and identifying the exact iris position, such as methods described in [5].

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, New York, 1999.
2. M. Berry, S. Dumais, and T. Letsche. Computational Methods for Intelligent Information Access. In *Proceedings of the 1995 ACM/IEEE Supercomputing Conference*, San Diego, California, USA, 1995.
3. J. Daugman. Statistical richness of visual phase information: Update on recognizing persons by iris patterns. *International Journal of Computer Vision*, 45(1):25–38, 2001.
4. J. Daugman. The importance of being random: statistical principles of iris recognition. *Pattern Recognition*, 36(2):279–291, 2003.
5. J. Daugman. New methods in iris recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 37(5):1167–1175, 2007.
6. C. Faloutsos and K. Lin. FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets. *ACM SIGMOD Record*, 24(2):163–174, 1995.
7. G. R. Hjaltason and H. Samet. Properties of embedding methods for similarity searching in metric spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):530–549, 2003.
8. T. G. Kolda and D. P. O'Leary. Computation and uses of the semidiscrete matrix decomposition. In *ACM Transactions on Information Processing*, 2000.

9. P. Praks, L. Machala, and V. Snášel. Iris Recognition Using the SVD-Free Latent Semantic Indexing. In *MDM/KDD2004 – Fifth International Workshop on Multimedia Data Mining "Mining Integrated Media and Complex Data" in conjunction with KDD'2004 - The 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; Section 2. Multimedia Data Mining: Techniques and Applications*, pages 67–71, Seattle, WA, USA, 2004.
10. K. Saeed. *Image Analysis for Object Recognition*. Bialystok Technical University Press, Bialystok, Poland, 2004.
11. K. Saeed and M. K. Nammous. A speech-and-speaker identification system: Feature extraction, description, and classification of speech-signal image. *IEEE Trans. on Industrial Electronics*, 54(2):887–897, 2007.
12. G. Salton and G. McGill. *Introduction to Modern Information Retrieval.* McGraw-ill, New York, USA, 1983.
13. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.