IMAGE RETRIEVAL USING EFFICIENT FEATURE VECTORS GENERATED FROM COMPRESSED DOMAIN

Daidi Zhong, Irek Defée

Department of Information Technology, Tampere University of Technology.
P.O. Box 553, FIN-33101 Tampere, Finland
{daidi.zhong, irek.defee}@tut.fi

ABSTRACT

We consider the use of quantized block transform coefficients to the image database retrieval problem. Based on the transform coefficients feature vectors are constructed. These feature vectors are used in histograms and combination of histograms with similarity measure for database retrieval. Experiments on public face image database show good performance of the approach in comparison with other methods.

1. INTRODUCTION

Block transforms are widely used in signal compression due to their ability to preserve of perceptual information even under strong quantization. This property of block transform should be also very useful for pattern recognition and retrieval tasks. However, application of block transforms in these areas has to be done within a suitable framework. This is because patterns are formed by global distribution of features which is described both by detailed geometry (structure) as well as statistics. In this paper formation of features and description of their global distribution based on statistics are investigated. Methods which are developed are compared with other approaches within the standardized framework of image database retrieval evaluation.

Our approach to feature representation based on quantized block transforms is related to the Local Binary Pattern (LBP) [1, 11] and texture spectrum unit methods [2]. We do not apply those methods directly in the image pixel domain but to the quantized block transform coefficients and we use different thresholding schemes and we do not apply them directly in the image pixel domain but to the quantized block transform coefficients. We form features as binary and ternary vectors based on the DC and AC quantized transform coefficients.

Global description of patterns used in this paper is based on histograms of feature vectors. The histograms describe the statistical content of images using defined feature vectors. We use both histograms and combination of histograms for different feature vectors. Such description does not use geometry of feature locations but nevertheless we are able to show that it is quite powerful. Using histograms, standard city-block metrics can be used as similarity measure for patterns. We apply this approach to the image database retrieval problem. Evaluation is done and results are presented in the standardized framework developed for FERET face database with its performance evaluation based on Cumulative Match Score (CMS) [3]. We first tune the free coefficients using limited training set and then evaluation of performance is conducted using large database test set. Results are presented for the FERET database and compared with other methods. It is shown that our method which is based only on global statistics of selected features and defined similarity measure has performance in the range of achieved by other methods which are more complex.

2. BLOCK TRANSFORMS AND QUANTIZATION

The specific block transform used in this paper was introduced in the H.264 standard [4] as particularly effective and simple. The transform matrix of the transform is denoted as $\boldsymbol{B_f}$ and the inverse transform matrix is denoted as $\boldsymbol{B_i}$. They are defined as

$$\boldsymbol{B}_{\mathrm{f}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \qquad \boldsymbol{B}_{\mathrm{i}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0.5 & -0.5 & -1 \\ 1 & -1 & -1 & 1 \\ 0.5 & -1 & 1 & -0.5 \end{bmatrix}$$

A 4x4 image pixel block $\boldsymbol{P}$ can be forward transformed to block $\boldsymbol{H}$ using (1), and the quantization process **Q( )** is used to remove the irrelevant information, which will result in quantized version of $\boldsymbol{H}$, $\mathbf{Q}(\boldsymbol{H})$. During the reconstruction process, the inverse quantization process $\mathbf{Q}^{-1}$ **[]** is applied to the quantized block $\mathbf{Q}(\boldsymbol{H})$, and block $\boldsymbol{R}$ is subsequently reconstructed from the inverse-quantized block $\mathbf{Q}^{-1}$ [$\mathbf{Q}(\boldsymbol{H})$], using (2)

$$H = B_f \times P \times B_f^T \tag{1}$$

$$R = B_i^T \times Q^{-1}[Q(H)] \times B_i \tag{2}$$

with superscript T denoting transposition.

The leading element of the matrix $\boldsymbol{H}$ is called the DC coefficient. All other elements are called AC coefficients. There are thus 15 AC coefficients in the matrix $\boldsymbol{H}$ but many of them will have zero value after the scalar quantization $\mathbf{Q}(\boldsymbol{H})$ is applied. Quantization has the effect of limiting the number of different blocks in an image. The quantized blocks with AC coefficients can be used directly in image retrieval as described in [5] where they were called AC Block Patterns (ACBP).

3. FEATURE VECTORS

In this paper we define features as Feature Vectors (FV) based on quantized block transform coefficients. The block transform (1) can be performed for non-overlapping image blocks or for partially overlapping image blocks. In either case we can group separately DC and AC coefficients for all transform blocks of an image into matrices. From these matrices feature vectors are formed.

3.1 DC binary feature vectors

From the matrix of DC coefficients, 3x3 submatrices of DC coefficients are considered. The eight DC coefficients surrounding the center one can be thresholded to form a binary vector with length eight:

If the DC value < threshold then put 0
If the DC value ≥ threshold then put 1

Value of the threshold can defined in many ways, e.g. as the value of the central coefficient in the 3x3 submatrix or as the mean value of all its nine coefficients. We choose the latter way and the binary feature vector formed in this way is denoted as DC-BFV.

3.2 AC binary feature vectors

Following the procedure described for the formation of the DC-BFV above, the binary feature vectors can be also defined for the AC coefficients. We shall denote such vectors as AC-BFV. There are 15 AC coefficients possible within each block but many coefficients will take zero value when quantization is applied. We select in the 4x4 block matrix coordinates AC coefficients which are (0,1), (1,0) and (3,0) and their positions are shown marked in Figure 1. Such decision is made based on the retrieval tests run over training data. In addition, using smaller number of AC coefficients can reduce the complexity.

	X		
X			
X			

Figure 1 – The AC coefficients used

3.3 Ternary feature vectors

Extension of binary feature vector described above to a ternary feature vector which can be obtained by the following thresholding:

If the DC value < threshold put 0
If the DC value = threshold put 1
If the DC value > threshold put 2

with the same rules for the AC coefficients.

The Ternary Feature Vector (TFV) is based on flexible threshold range instead of a single value of the threshold above. For a 3x3 matrix of DC or AC coefficients, the threshold T is defined as

$$T_{-}^{+} = M \pm (X - N) \times f \qquad (3)$$

where $\boldsymbol{f}$ is real number from the interval (0,1), $\boldsymbol{X}$ and $\boldsymbol{N}$ are maximum and minimum coefficient values in the coefficient matrix, and $\boldsymbol{M}$ is the mean value of the coefficients. The thresholded values are to be either 0, 1 or 2

If the coefficient value ≤ T+ put 0
If the coefficient value ≥ T- put 2
otherwise put 1

The resulting thresholded vectors of length eight are subsequently converted to decimal numbers in the range of [0, 6560].

4. HISTOGRAMS OF FEATURE VECTORS AND SIMILARITY MEASURE

4.1 Histograms of Feature Vectors

Statistical distribution of specific feature vectors in images can be represented in normalized way by histograms. Example of such a histogram is shown in Figure 2. The histograms can be also seen as 1-D vectors and city-block metrics can be used as similarity measure.

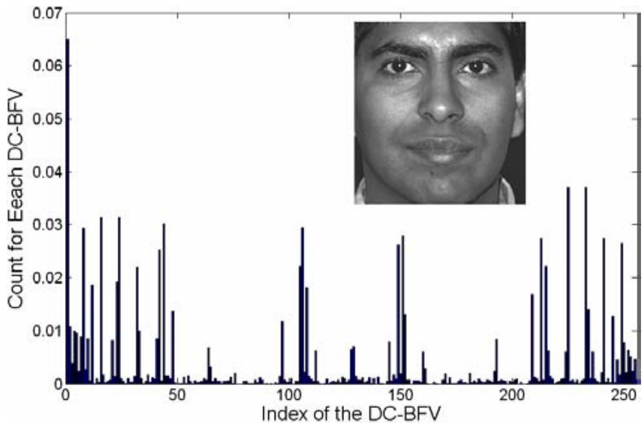

Figure 2. Example histogram of DC Binary Feature Vectors

Several types of feature vectors which were described above will lead to different feature vector histograms. We shall define now combinations of these histograms and similarity measure for them.

4.2 Combination of histograms

We can form three types of feature histograms based on ACBP, DC Feature Vectors (DC-FV) and AC Feature Vectors (AC-FV) respectively; These histograms can be combined in several ways starting with any two of the histograms, as follows

[Combined_2Histogram] =
[Histo_A α x Histo_B] (4)

where Histo_A and Histo_B can be any one of the histograms with ACBP, DC-FV or AC-FV.

The combined histogram is thus a vector formed by concatenating the histograms A and B with coefficient $\boldsymbol{\alpha}$ introduced for controlling the relative weight of A and B in the combination. In the same way we can combine the three histograms, with two weight parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$:

[Combined_3Histogram] =
[Histo_ACBP α x Histo_AC-FV β x Histo_DC-FV] (5)

For AC-FV histograms, we use three AC coefficients as shown in Figure 1. They are combined in the following way:

[Histo_AC] =
[Histo_coef1 Histo_coef2 Histo_coef3] (6)

The similarity measure based on the combined histogram is the same as for the single histogram but obviously the values of coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ have to be established first. This is done within the framework of database retrieval system training described next.

4.3 Image Database Retrieval

Our image database retrieval problem is formulated as follows. Given an image database set **S**={ I_1, …., I_n} we would like to establish for a certain key image **I** if there are images similar to it in the database. For this we use the feature vector histograms of images and similarity measure defined above to find its minimum values for the image **I** and images from the database **S**.

However, before this can be done the parameters used for the calculation of histograms and similarity measure need to be found using training database set. This set can be selected as small subset of the database ***S*** or some key images selected from outside of it. Knowing the correct responses for the training database allows us to tune the parameters to achieve best retrieval results. The optimal parameter set which is going to be found out during the training process includes: the Quantization Scalar for (QS) for the quantization of transform blocks $\mathbf{Q}(\boldsymbol{H})$ in (2), coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in (5,6) and the parameters $\boldsymbol{f}$ in (3). The optimal parameter set is identified as the one which is maximizing the retrieval performance over training database. The resulting optimal parameter set is applied to the test database ***S*** to evaluate the actual system performance.

5. RESULTS OF RETRIEVAL PERFORMANCE

5.1 Image database

To train and test the retrieval performance of the proposed system we used a famous public database of face images -- FERET [6], which contains about two thousand images from about one thousand subjects. The advantage of using FERET database, apart of its size, is standardized evaluation method based on performance statistics reported as Cumulative Match Scores (CMS), which are plotted on a graph [3]. The horizontal axis of the graph is the retrieval rank and vertical axis is the probability of identification (PI) (or percentage of correct matches). This lets one know how many images have to be examined to get a desired level of performance. For simplicity, many researchers use the CMS at the first rank to represent the recognition rate.

The FERET database provides some tools for preprocessing of the face images. The images can be cropped by the provided geometrical information about eye, nose and mouth. They can be subsequently aligned, and adjusted by illumination normalization. In our evaluation, the images

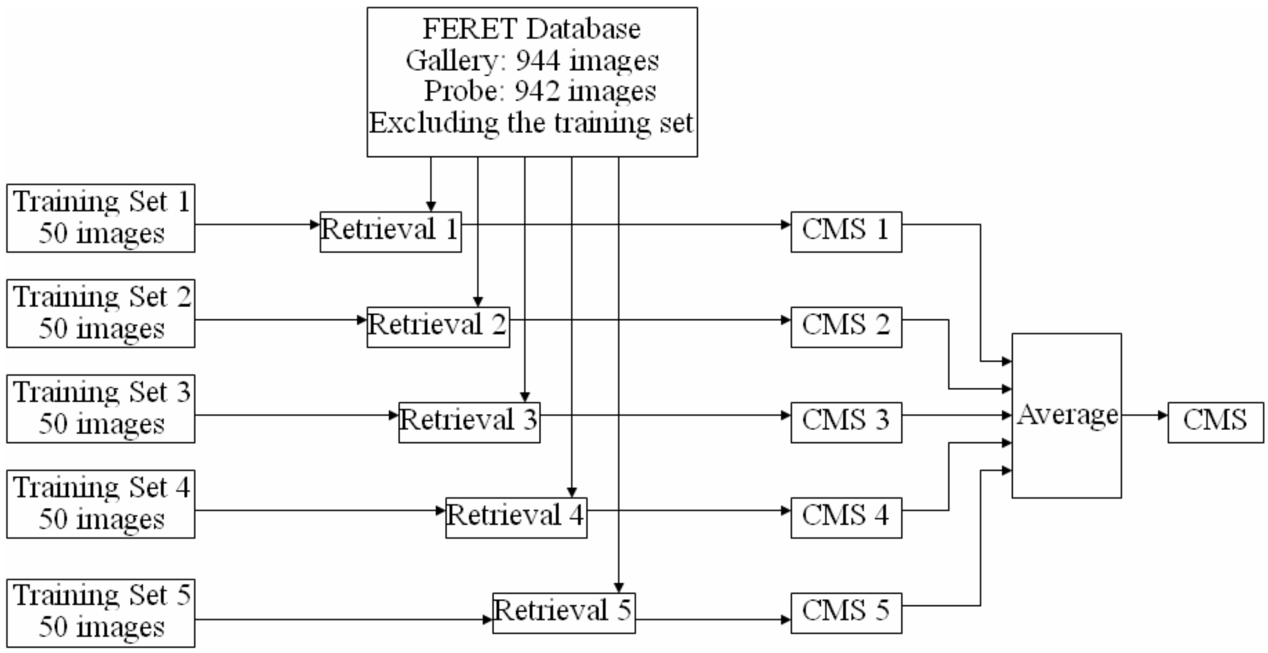

Figure 4. The training and retrieval system. Five training sets are used for testing the robustness and generality of our system.

were simply cropped to sizes, which roughly contain the face area (e.g. Figure 3).

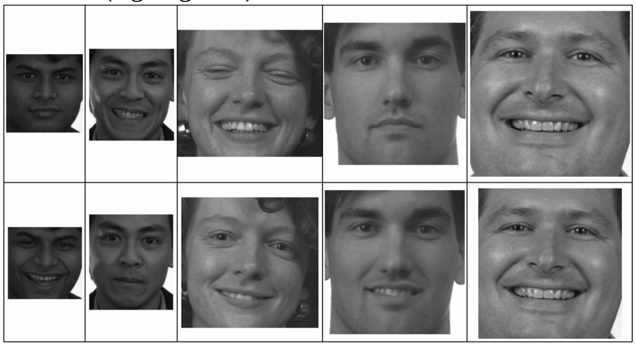

Figure 3. Some exemplar images from FERET database. The upper row is from gallery set, and the lower row is from probe set. No pre-processing is performed over them. They have different sizes.

5.2 Training process

Before the real retrieval task is performed, we first select the training sets from FERET database. Here the problem is selection of the training set which on one hand should be small but on the other hand the results should not depend on particular selection of the set. In our experiment, five different test image sets are used for training, each composed of fifty images selected randomly. The training and retrieval system is illustrated in Figure 4.

The size of one training set is 50 images. Some example pairs of images are shown in Figure 3. For each of the test sets optimal coefficients described in Section 4.3 were found. In result we established that coefficients found for each set are very similar and the system performance is not sensitive to the selection of the test set.

5.3 Performance of BFV histograms using FERET

We first studied the performance of BFV histograms using the FERET database. Figure 5 shows the Cumulative Match Scores (CMS) results for AC-BFV and DC-BFV histograms. From the result, one can find out that DC-BFV performs better than AC-BFV and the Rank-1 CMS is 0.733 and 0.603, meaning that probability for correct first hit retrieval is 73.3% and 60.3% respectively.

5.4 Performance of TFV histograms using FERET

Here we compare the performance of DC-TFV and AC-TFV. The TFV histogram may contain maximum 6561 bins. To reduce the complexity, the bins are sorted and only the first several hundred high rank bins are used for the histogram. The results are shown in Figure 6. Comparing with Figure 5 one can see that the DC-TFV and AC-TFV histograms provide much better results, with 80.4% and 73.3% correct first hit retrieval.

5.5 Performance of combined ACBP and TFV histograms

Since histograms of ternary TFV vectors provide better results than histograms of binary BFV vectors we evaluated the performance of combined histogram of ACBP, AC-TFV and DC-TFV histograms as defined in (6). In Figure 7 the Cumulative Match Scores (CMS) results of combined histograms are shown overlaid with results for single histograms. One can see that combination of ACBP and TFV histograms provides significantly better retrieval performance than using them individually, with 91% Rank-1 CMS correct retrieval.

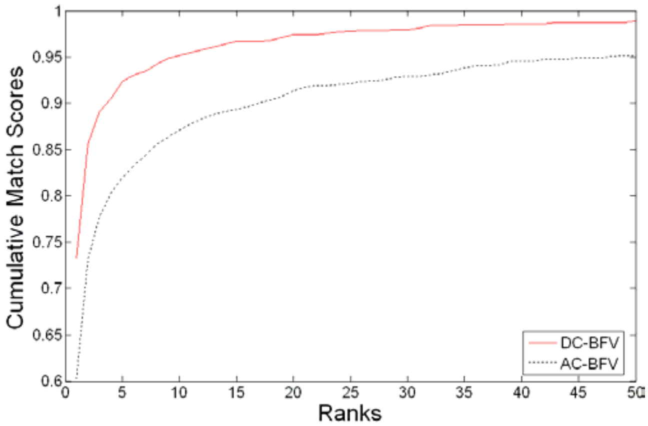

Figure 5. CMS over FERET database using DC-BFV and AC-BFV. The CMS at certain rank represents the ratio of correct retrievals among 992 probe images.

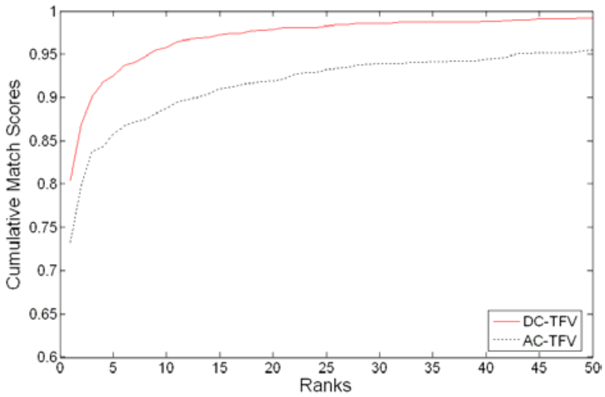

Figure 6. CMS over FERET database using DC-TFV and AC-TFV. The CMS at certain rank represents the ratio of correct retrievals among 992 probe images.

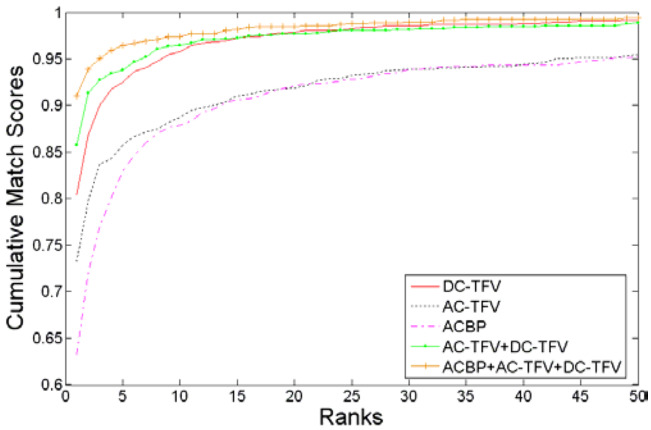

Figure 7. Cumulative match scores results of combined histograms.

And the variation between the performances of different training sets is 1%.

For comparison, results obtained by other methods are shown in Table 1 which lists the results based on the FERET database of the same version (released in 2003); The results are shown for the Rank-1 CMS. Such comparisons are only informative since there are differences in the preprocessing of images used for testing. Nevertheless, our approach having over 90% correct retrieval is much better than many other methods and it is lower only to some which are much more complicated.

Reference	[7]	[8]	[9]	[10]	Proposed
Rank-1 CMS (%)	79.4	60.2	73.08	97.9	91

Table 1. List of the referenced results based on FERET

6. CONCLUSIONS

In this paper we defined several types of feature vectors based on quantized block transform coefficients. Next histograms of feature vectors and combination histograms are introduced together with city-block similarity measure. The system performance evaluation is done using face databases FERET with cumulative match score performance measure. Our final results show over 90% correct recognition rate for FERET. Performance is very good since our approach is based only on the statistics of features, while other more complicated methods may also incorporate some structural information.

REFERENCES

[1] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions", *Pattern Recognition*, Vol. 29, No. 1, pp. 51-59, 1996.
[2] He, D.-C. and Wang, L., "Texture unit, texture spectrum, and texture analysis", *IEEE Trans. Geo Sci. Remote Sens*, 28 (1), 509-513, 1990
[3] P. J. Phillips, H. Moon, P. J. Rauss, and S. Rizvi, "The FERET evaluation methodology for face recognition algorithms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 10, October 2000.
[4] Joint Video Team of ITU-T and ISO/IEC JTC 1, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)", *Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050*, 2003.
[5] D. Zhong and I. Defée, "DCT histogram optimization for image database retrieval", *Pattern Recognition Letters*, 26(14): 2272-2281, 2005.
[6] FERET Face Database, Available at: http://www.itl.nist.gov/iad/humanid/feret/, 2003.
[7] J. Shi, A. Samal, and D. Marx, "Face Recognition Using Landmark-Based Bidimensional Regression", in *Proceeding of ICDM 2005*, 765-768, 2005.
[8] J. Shi, A. Samal, and D. Marx, "How Effective are Landmarks and Their Geometry for Face Recognition", *Computer Vision and Image Understanding*, 102(2):117-133, 2006.
[9] J. Roure, and M. Faundez-Zanuy, "Face recognition with small and large size databases", in Proc. of ICCS 2005.
[10] C. Kim, J. Oh, and C.H. Choi, "Combined Subspace Method Using Global and Local Features for Face Recognition", in *Proceedings of IJCNN 2005*.
[11] Timo Ahonen, Matti Pietikainen, Abdenour Hadid, Topi Maenpaa, "Face Recognition Based on the Appearance of Local Regions", in Proceedings of ICPR 2004.