# Retrieval-Aware Image Compression, Its Format and Viewer Based Upon Learned Bases

Naoto Katsumata[1], Yasuo Matsuyama[2], Takeshi Chikagawa[3], Fuminori Ohashi[2], Fumiaki Horiike[2], Shun'ichi Honma[2], and Tomohiro Nakamura[2]

[1] Yahoo Japan,
[2] Waseda University, Tokyo 169-8555, Japan
[3] Nomura Research Institute, Japan
{katsu, yasuo, take-c-chika, fumi, fmi_h, shunichi1029, nt_naka}@wiz.cs.waseda.ac.jp
http://www.wiz.cs.waseda.ac.jp/index-e.html

**Abstract.** A retrieval-aware image format (rim format) is developed for the usage in the similar-image retrieval. The format is based on PCA and ICA which can compress source images with an equivalent or often better rate-distortion than JPEG. Besides the data compression, the learned PCA/ICA bases are utilized in the similar-image retrieval since they reflect each source image's local patterns. Following the format presentation, an image search viewer for network environments (Wisvi; Waseda image search viewer) is presented. Therein, each query is an image per se. The Wisvi system based on the "rim" method successfully finds similar-images from non-uniform network environments. Experiments support that the PCA/ICA methods are viable to the joint compression and retrieval of digital images. Interested test users can download a $\beta$-version of the tool for the joint image compression and retrieval from a web site specified in this paper.

## 1 Joint Data Compression and Retrieval of Images

Growing popularity of the Internet increases the necessity of image retrieval systems more and more. For instance, the service of flickr [1] helps image sharing among blog groups. Thus, retrieved images migrate frequently among the network environments which contain PC's and mobile phones. In such cases, the direct retrieval from a query image is desirable. Then, computational intelligence methods with learning are expected to contribute to this class of problems Therefore, this paper addresses the following problems:

(a) To utilize learned image bases from the principal component analysis (PCA) and the independent component analysis (ICA) so that the joint data compression and retrieval is effectively achieved. The data compression presented in the text can outperform JPEG. On the similar-image retrieval, the authors had made extensive experiments to compare the color bin method and the learned bases method [2]. Due to this, the main purpose of this paper is

I. King et al. (Eds.): ICONIP 2006, Part II, LNCS 4233, pp. 420–429, 2006.

set to find the method to achieve the joint performance of the compression and retrieval of digital images.

(b) To define the retrieval-aware image format, say rim (Retrieval-aware IMage format).

(c) To give a useful viewer. As will be observed in the main text, items (a) and (b) are successfully fulfilled. Therefore, systems which can handle images directly as queries become worthy to build. The Wisvi (Waseda Image Searchable VIewer) is presented for this purpose so that uninitiated users can find desirable images via human-friendly method. This system is helpful for the opinion test to measure the system performance.

(d) To design the whole methods and systems to be applicable to scattered network environments as well as databases.

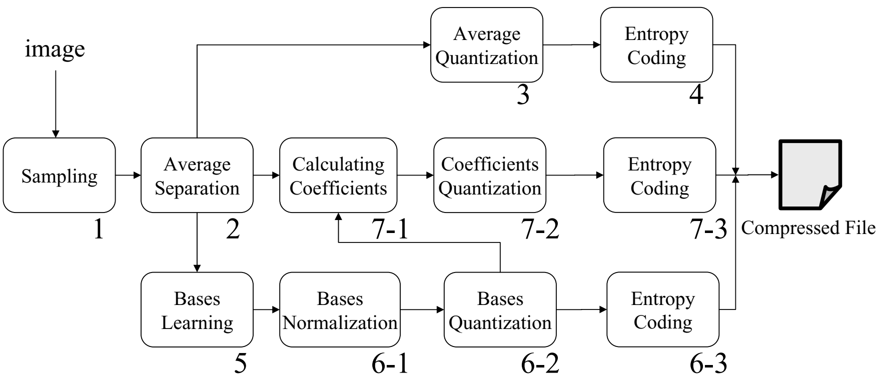


**Fig. 1.** Retrieval-aware image compression

## 2 Utilization of Learned Image Bases

As was stated in Section 1, the purpose of this paper is

(a) to show efficient and retrieval-aware image compression methods,
(b) to give an effective format for this purpose,
(c) and to design a user-friendly viewer system.

Figure 1 illustrates the total system for the image compression with the purpose of the similar image retrieval. This system will use PCA and ICA bases after the mean value separation. The blocks with numbers in this figure have the functions explained below. Each item number corresponds to the block number.

(1) Sampling:
At this stage, patches $\{I(x, y)\}$ with the size of $m \times m$ are obtained. Each patch is considered as a vector $\boldsymbol{x}$.

$$\begin{aligned}\boldsymbol{x} &= [\ R(x_1,y_1), R(x_2,y_1),\cdots, R(x_m,y_m),\\ &\quad G(x_1,y_1), G(x_2,y_1),\cdots, G(x_m,y_m),\\ &\quad B(x_1,y_1), B(x_2,y_1),\cdots, B(x_m,y_m)\ ]^T\\ &\stackrel{\text{def}}{=} [\boldsymbol{x}_R, \boldsymbol{x}_G, \boldsymbol{x}_B]^T \end{aligned} \tag{1}$$

(2) Average separation:
Color component's sample mean values are computed and subtracted from each component of {R, G, B}.

$$\boldsymbol{x} \leftarrow [\boldsymbol{x}_R - \mu_R,\ \boldsymbol{x}_G - \mu_G,\ \boldsymbol{x}_B - \mu_B]^T \tag{2}$$

In later experiments, the image compression using

$$\boldsymbol{\mu} = [\mu_R,\ \mu_G,\ \mu_B] \tag{3}$$

will be found better than using $\boldsymbol{\mu}_{all-color}$, which is a single vector mean of vector patches, contrary to our naive intuition.

(3) Average quantization:
The average $\mu_{color}$ (the index "*color*" stands for $R$, $G$, or $B$) is quantized as follows.

$$\hat{\mu}_{color} \leftarrow \lfloor \mu_{color}/q_{avg} \rfloor \tag{4}$$

The quantization step size is as follows.

$$q_{avg} \leftarrow \lfloor q_{cff}/(1.5m) \rfloor \tag{5}$$

Here, $q_{cff}$ is the quantization size for basis coefficients explained in later sections.

(4) Average entropy coding:
This step computes the difference between contiguous frames as is adopted in JPEG.

$$\Delta\hat{\mu}_{color}(k) \leftarrow \hat{\mu}_{color}(k) - \hat{\mu}_{color}(k-1) \tag{6}$$

After this step, the run-length Huffman coding is executed for the average compression.

(5) PCA bases learning:
Computing the bases for PCA starts from the normalization for the zero mean. This is already completed in the average separation. Then, the covariance is computed by

$$\boldsymbol{C} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T]. \tag{7}$$

Then, the data reduction matrix is computed.

$$\boldsymbol{V} = \boldsymbol{D}^{-1/2}\boldsymbol{E}^T \tag{8}$$

Here, $\boldsymbol{D}$ is a diagonal matrix of the first $L$ large eigenvalues of $\boldsymbol{C}$. $\boldsymbol{E}$ is a matrix whose columns are eigenvectors corresponding to $\boldsymbol{D}$. Then, the reduced or low-pass filtered vector is expressed by

$$\boldsymbol{z} = \boldsymbol{V}\boldsymbol{x}. \tag{9}$$

Then,

$$\bar{\boldsymbol{x}} \stackrel{\text{def}}{=} \boldsymbol{V}^{-1}\boldsymbol{z} \stackrel{\text{def}}{=} \hat{\boldsymbol{U}}_{PCA}\boldsymbol{z} \tag{10}$$

is the image restoration. This $\hat{\boldsymbol{U}}_{PCA}$ is the set of the PCA bases.

(5') ICA bases learning:
After obtaining the PCA bases, another set of powerful bases can be obtained. These are ICA bases (independent component analysis) [3], [4].

$$\hat{\boldsymbol{s}} = \hat{\boldsymbol{W}}\boldsymbol{z} = \hat{\boldsymbol{W}}\boldsymbol{V}\boldsymbol{x} \tag{11}$$

Here, $\hat{\boldsymbol{s}}$ is the estimated coefficients whose components are independent each other. The image restoration is performed by

$$\bar{\boldsymbol{x}} \stackrel{\text{def}}{=} (\hat{\boldsymbol{W}}\boldsymbol{V})^{-1}\hat{\boldsymbol{s}} \stackrel{\text{def}}{=} \hat{\boldsymbol{U}}_{ICA}\hat{\boldsymbol{s}}. \tag{12}$$

This $\hat{\boldsymbol{U}}_{ICA}$ is the set of the ICA bases.

(6) Bases normalization (6-1):
Let $\boldsymbol{A}$ be the PCA or ICA basis matrix. First, each basis which is a column vector of $\boldsymbol{A}$ is normalized so that the basis norm is unity: $\|\boldsymbol{a}_i\| = 1$.
Basis quantization (6-2):
Then, the quantization step size is computed.

$$q_{bases}(i) \leftarrow \left\lfloor \left(2^{b_{prec}-1} - 1\right) / a_{max}(i) \right\rfloor \tag{13}$$

Here, $a_{max}(i)$ is the maximum value of the normalized basis $\boldsymbol{a}(i)$. The number $b_{prec}$ sets the granularity of the quantization. Experimentally decided value is 6 bits, i.e., $b_{prec} = 6$. Then, the quantization for the bases is performed by

$$\hat{\boldsymbol{a}}(i) \leftarrow \lfloor \boldsymbol{a}(i) q_{bases}(i) \rfloor. \tag{14}$$

Entropy coding (6-3):
The loss-less data compression is performed by the run-length Huffman coding after computing the difference as is illustrated in Figure 2.

(7) Coefficients calculation (7-1):
Using the quantized bases, the superposition coefficients for the bases are computed.

$$\boldsymbol{s} \leftarrow \boldsymbol{A}^{-1}\boldsymbol{x} \tag{15}$$

Coefficients quantization (7-2):
The $i$-th component is quantized as follows.

$$\hat{s}(i) \leftarrow \lfloor (s(i) q_{bases}(i)) / q_{cff} \rfloor \tag{16}$$

Here, $q_{cff}$ is a design parameter which can be set by users.
Entropy coding (7-3):
Finally, the run-length Huffman coding is applied column-wise to the coefficient matrix $\boldsymbol{S}$ for the handled image.

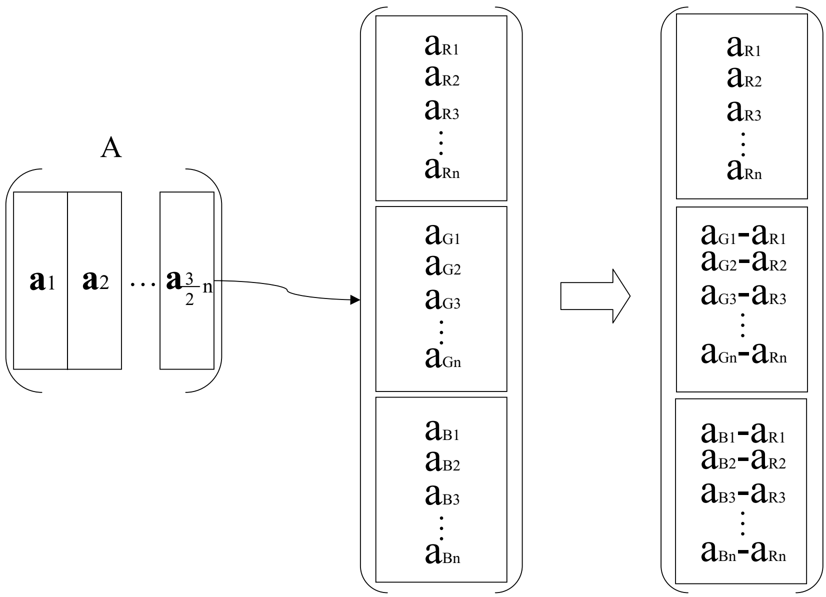


**Fig. 2.** Computation of bases components' differences

## 3 File Format with Learned Bases

Figure 3 illustrates the file format which contains headers and compressed information. As can be observed in this figure, the organization of this format is blockwise.

(1) File header:
The file header contains the image size and the offset for each block. But, the file header is free from each block's format so that each block information's independence is maintained.

(2) Information header:
This part is prepared for extra important information which may or may not be related to the image compression. Such information includes the author name and the copyright. Tags like MPEG7 can also be such information.

(3) Average:
The average is used for the image retrieval using color information. This information can be utilized to make thumbnails and progressive expressions.

(4) Bases:
This part contains compressed information of image bases. PCA bases and ICA bases are major targets in this study.

(5) Coefficients:
This part contains compressed information for the superposition of bases.

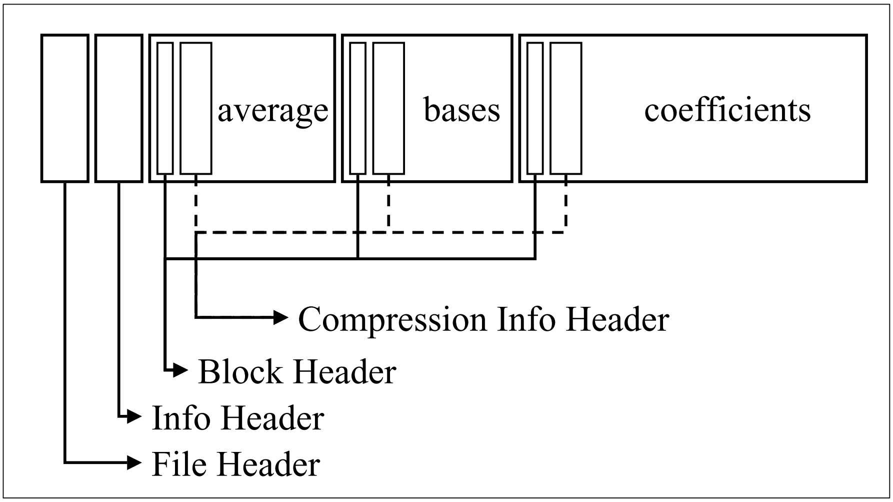


**Fig. 3.** File format

## 4 Viewer for Similar Image Retrieval

Figure 4 is a screen shot of the designed viewer Wisvi for the similar-image retrieval. Given a directory name, Wisvi looks for images including all subdirectories. Wisvi is used in the compression performance evaluation and opinion tests for the retrieval.

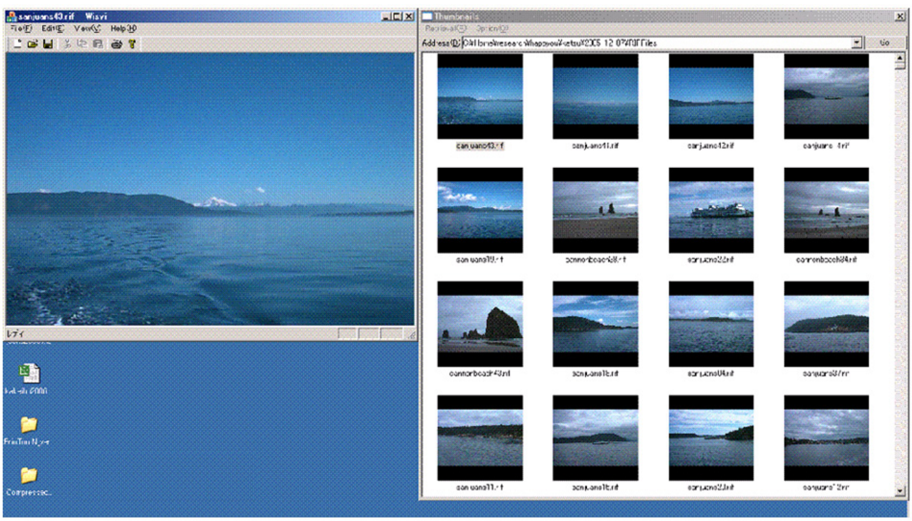

**Fig. 4.** Wisvi: A similar-image retrieval viewer

The large picture in the upper-left of Figure 4 is the query image. The task is to find similar images from specified directories. Small images on the right side are sorted from the query image to others which are judged similar. It is worthy to note here that the PCA/ICA-based search described in the next section can find similar images with different $x$-$y$ ratios [2].

## 5 Preliminary Experiments to Evaluate the Design Principle

Here, we explain why the aforementioned compression method fits to the similar-image retrieval.

(a) Average of colors:
In this system, three averages of {R, G, B} are encoded for the image compression. From an uninitiated intuition, this might look inferior to using a single granular average of the whole color. But, experiments showed that the three component method beats the one average method. This illustration is omitted because of the space limitation.

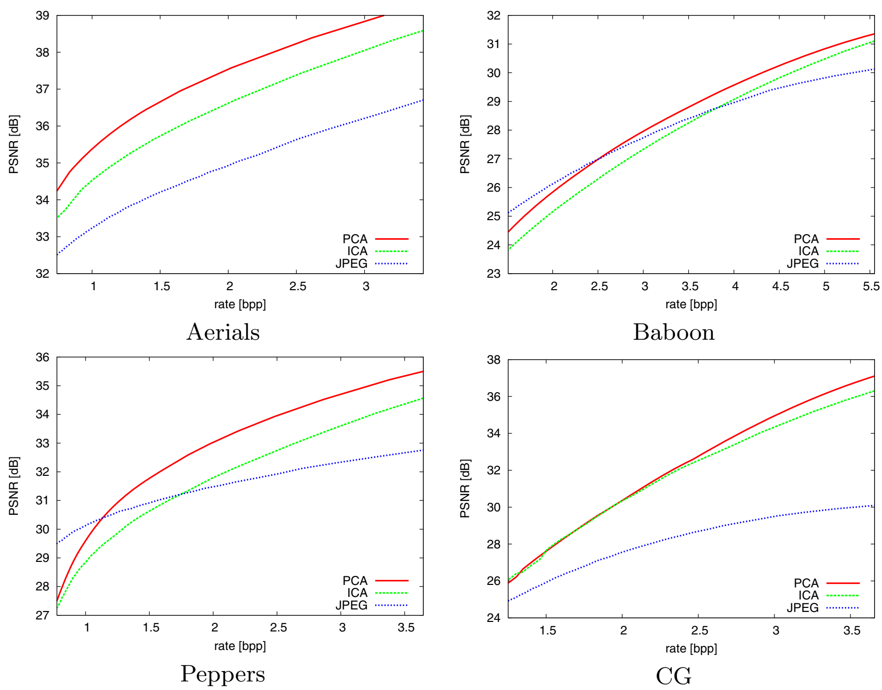


**Fig. 5.** Rate-distortion curve

(b) Tight fitting versus universal bases:
Here, "tight fitting bases" stands for the bases from handled images (e.g., the query image). On the other hand, "universal bases" means the bases obtained from a good amount of mixtures of images. Experiments show that the universal bases wins for images with smaller sizes. On the other hand, tightly fitting bases are better for images with larger sizes (illustration is omitted because of the space limitation). Since pixel sizes adopted by recent digital cameras and cellular mobile phones are increasing, this experiment recommends the method of tightly fitting bases.

(c) Compression performance:
Figure 5 illustrates the rate-distortion curves of three compression methods {PCA bases, ICA bases, JPEG} (actually, they are rate-quality curves). These are examples from extensive studies. As can be observed from this figure, the PCA bases outperform the two others. JPEG wins only within low quality ranges. Due to the margin of this win, the joint compression and retrieval of images is made possible.

(d) Similarity measures:
The similarity measure which compare two images is a combination of the color-sensitive part $S_{color}$ and the texture/edge sensitive part $S_{bases}$.

$$S = \alpha S_{bases} + (1 - \alpha) S_{color}, \quad 0 \leqq \alpha \leqq 1. \tag{17}$$

Here, $\alpha$ is a design parameter set by users (e.g., $\alpha = 0.3$).

The color similarity $S_{color}$ is computed as the average of patch similarity defined by the inner product. The basis similarity $S_{bases}$ is also computed by using the inner product. But, this part needs to consider how to find bases pair to compare for the computational efficiency. Readers are requested refer to [2] for details.

## 6 Performance of the Similar-Image Retrieval

Figure 6 summarizes the result of the opinion tests by 10 uninitiated users on the Ground Truth Database [5]. This figure compares the retrieval performances by the PCA basis method and the ICA basis method.

The retrieval is judged to be in success if the target to the query image was contained within top $x\,\%$ of all images. This $x$ is called the success line which is the horizontal axis of Figure 6. The vertical axis, the success rate, is measured by showing images one by one to the opinion test subjects. The following summarizes this result on the similar-image retrieval.

(a) Both PCA and ICA methods are judged to be viable.
(b) The ICA basis method outperforms the PCA basis method.
(c) The PCA basis method is faster. This is because the ICA basis computation requires the result of PCA (cf. Section 2).

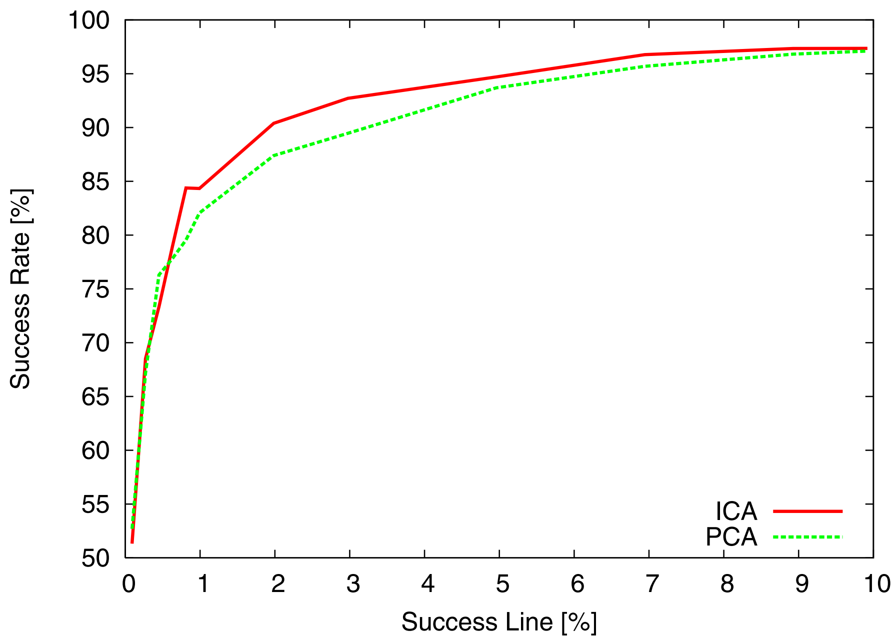


**Fig. 6.** Retrieval success rates for PCA and ICA methods

## 7 Conclusion

The retrieval-aware image compression using learned bases was presented. This paper presented the following.

(1) The presented methods for the joint data compression and similar-image retrieval were successful. These methods are based on the learned PCA and ICA image bases.
(2) A basic image format was presented (the rim format; Retrieval-aware IMage format).
(3) The PCA basis method outperforms JPEG for data compression.
(4) Both PCA and ICA bases are successfully retrieval-aware.

This paper leads to the following studies for improvements, which are in progress. Some already show promising results.

(a) The compression experiments in the text used the uniform quantization and the run-length Huffman coding. This part can be improved at the cost of slight increase of computational complexity. The arithmetic coding is one possibility. Therefore, we applied EBCOT (Embedded Block Coding with Optimal Truncation) which is used in JPEG2000. EBCOT comprises the arithmetic coding as the main step. The compression performance was improved. Quantitative results will be given in a separate repots.

(b) Computation speedup for bases using software and/or hardware is desirable.
(c) Comparison with JPEG2000 needs to be studied. JPEG2000 is compression effective. On the compression performance per se, item (a) already gives an answer. On the performance for the retrieval-awareness, additional sophistications are necessary. This is in progress.
(d) Upon this paper is published, a $\beta$-version of the tool set for the joint image compression and retrieval will be made downloadable at the URL given in the first page of this paper.

## References

1. Flikr: (2005) www.flickr.com
2. Katsumata, N., Matsuyama, Y.: Database retrieval for similar images using ICA and PCA bases. Engineering Applications of Artificial Intelligence **18** (2005) 705–717
3. Hyvärinen, A.: Fast and robust fixed-point algorithm for independent component analysis. IEEE Trans. NN **10** (1999) 626–639
4. Matsuyama, Y., Katsumata, N., Imahara, S.: The alpha-ICA algorithm. Proc. 2000 Int. workshop on ICA and BSS, Espoo, Finland (2000) 297–302
5. Ground Truth Database:
(1999) www.cs.washington.edu/research/imagedatabase/groundtruth