Image Retrieval Exploiting the Similarity of Vectorized Fractal Code Sets

Takanori Yokoyama, Toshinori Watanabe and Hisashi Koga
Graduate School of Information Systems
University of Electro-Communications
1-5-1, Chofugaoka, Chofu-shi, Tokyo, Japan
email: {yokotaka, watanabe, koga}@sd.is.uec.ac.jp

***Abstract*— We present a new similarity measure based on the spatial relations of vectorized fractal codes for direct retrieval from a fractal code database. A fractal code consists of a set of mapping information between similar regions in an image. This mapping information can be represented as a set of vectors carrying the spatial features of the image. We call this vector set as vectorized fractal code, and introduce a new similarity measure which reflects the rate of one-to-one corresponding vectors between two vectorized fractal codes. This similarity is robust to image distortions unless the correspondence between vectors collapses largely, and our retrieval method with this similarity differs from existing fractal retrieval methods in using the spatial information of fractal codes. In this paper, we explain this similarity and demonstrate its effectiveness by experiments. Especially, we show that the retrieval performance is better than the wavelet and histogram methods.**

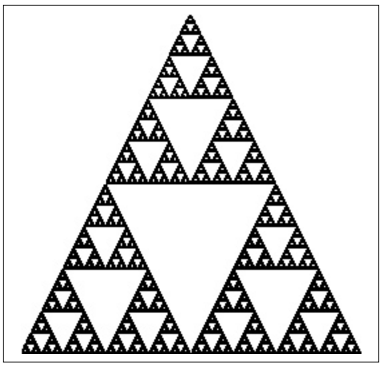


Fig. 1. Sierpinski triangle.

I. Introduction

Many researches of image retrieval prosper in recent years. Image retrieval methods use the various features such as histogram, color, texture and so on. Most of the current image search engines treat uncompressed image data for retrieval. However, because compressed data is generally used in communications on a network and accumulations to databases, it is indispensable to extract features from compression image data by decoding and to create the additional database of features for retrieval. If compression codes are available for image retrieval, practical advantages can be obtained. The retrieval techniques that handle original images are called *Pixel domain* techniques, while the techniques that handle compression codes are called *Compressed domain* techniques [1].

Compressed domain techniques need the feature extraction and/or the similarity definition from compression codes according to each compression technique. There are many compressed domain techniques based on the coefficients of compression codes, such as DCT, JPEG, VQ etc. Some retrieval techniques have been reported using fractal image compression [2]. Zhang *et al*. proposed Joint Fractal Coding [3]. In this method, the similarity between two images is obtained by performing the fractal image compression using both images. Tan and Han proposed Fractal Neighbor Distance [4]. This method applies the fractal code of a reference image to all images in an image database. The above techniques need both of fractal codes and image data, and cannot retrieve fractal codes directly without encoding or decoding processes.

In this paper, we propose a new similarity measure which can retrieve directly from the database that consists of fractal codes. This similarity exploits the spatial relations which fractal codes inherently have. Fractal image compression has a high compression rate, and can restore the image in arbitrary resolution. These properties are useful for image databases with retrieval system. Our retrieval method with the proposed similarity is robust to various image distortions unless the correspondence between vectors collapses largely.

In the next section, we present the overview about fractal image compression. In Sect. III, we explain the proposed retrieval method in detail. We treat a fractal code as a vector set, and define the similarity between the vector sets as images. In Sect. IV, we confirm the effectiveness of the similarity through several experimental results. Section V concludes this paper.

II. Fractal Image Compression

Fractals are generally self-similar and independent of scale. The Figure 1 shows an example image which has the self-similarity. Fractal image compression is based on the self-similarity in an image. Its compression principles were proposed by Barnsley [2]. He used a system of mappings called an Iterated Function System (IFS). However, it is quite difficult for his original method to encode the image automatically. Jacquin [5] improved Barnsley's method to encode automatically. As an image is divided, Jacquin's method is generally called a Partitioned IFS (PIFS). PIFS method has become the foundation of the present fractal image compression techniques [6]. We assume that a fractal code is obtained by PIFS encoding.

The characteristics of fractal image compression are enumerated below.

Image restoration by arbitrary resolution:
Images can be generated in arbitrary resolution from a compression code. Because of this property, a retrieval system has not to prepare images of various sizes. The system can generate images of required size. In addition, the restored image can exceed the resolution of original images, which is a good expression characteristic (called *fractal zoom* or *fractal blowup* [7]).

Secondary uses of a compression code:
The techniques based on fractal image compression, such as segmentation [8], highly precise outline extraction [9], and digital watermarking [10] are proposed. Fractal image compression is one of the promising compression techniques.

Image expression in spatial domain:
Fractal image compression expresses an image by similar relations between regions in spatial domain. These relations present an invariant feature of an image under affine type distortions [4].

From the above advantages, fractal image compression suits for construction the image database which consists of compression codes. In this paper, we will not describe the compression principles and encoding algorithms of the fractal image compression in detail. For details, refer [11] for example.

III. Similarity between Vectorized Fractal Codes

We explain the similarity measure between fractal codes for directly retrieval from a fractal code database. Fractal image compression records the self-similar regions in an image on a fractal code. Self-similarity relations between regions are expressed by contractive mappings. We define the similarity between two contractive mapping sets. The relations of the similar regions like Fig. 2 expressed by contractive mappings do not change so long as the image does not change largely [4]. Our similarity preserves this useful property for image retrieval. Here, we explain a mapping vector and a vectorized fractal code that are essential for our retrieval method, and describe how we treat them as the features of a fractal code which obtained from an image.

A. Fractal Code

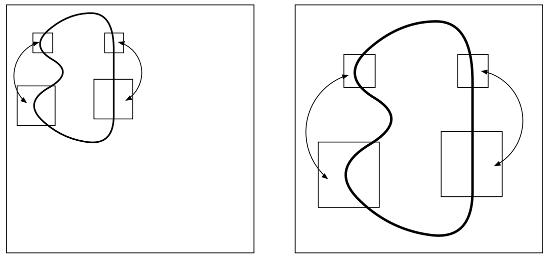
A fractal code consists of a set of contractive mappings. In encoding, an image is divided into large regions (which are called *Domain*) and smaller regions (which are called as *Range*). Domain regions may overlap, while the range regions tile the unit square. Each range region should be a similar contracted copy of the relevant domain region. A contractive mapping represents its relation, and the mapping set W as a compression code is represented by

$$W(\cdot) = \bigcup_{i=1}^{N} w_i(\cdot). \tag{1}$$

Here, N is the number of mappings in $W(\cdot)$ and $w_i(\cdot)$ is a contractive mapping which describes the similarity relation between the range region R_i and the domain region D_i, like the following formula.

$$\begin{bmatrix} x_{R_i} \\ y_{R_i} \\ z_{R_i} \end{bmatrix} = w_i \begin{bmatrix} x_{D_i} \\ y_{D_i} \\ z_{D_i} \end{bmatrix} \tag{2}$$

Usually, the *affine* transformation is used for w_i. Here, x_{R_i}, y_{R_i} represents the position of the R_i, and z_{R_i} is the brightness value at its position. x_{D_i}, y_{D_i} and z_{D_i} are those of domain region D_i. A fractal code has sufficient information which constructs the set of mappings.



(a) original image (b) distorted image

Fig. 2. Invariance of the relation of similar regions under distortions. A pair of squares which are linked by an arrow represents similar regions.

B. Vectorized Fractal Code

We treat a fractal code as a vector set. In fractal coding, the coefficients of luminance are usually determined by least-mean square method. These values of luminance coefficients dynamic vary with each region which have the same luminance after decoding. Moreover, the values of luminance are sensitive to illumination change. Hence, we use the spatial coefficients of a mapping. Coordinates of the top left corner position of a range region R_i and that of a domain region D_i are combined as a 4-dimensional vector $(x_{\dot{R}_i}, y_{\dot{R}_i}, x_{\dot{D}_i}, y_{\dot{D}_i})$. This vector is drawn as the line segment which has the initial point $(x_{\dot{R}_i}, y_{\dot{R}_i})$ and the terminal point $(x_{\dot{D}_i}, y_{\dot{D}_i})$ as shown in Fig. 3. We name this the *Mapping Vector*, and regard as a feature of an image. Fig. 4 shows the mapping vector set extracted from the fractal code of the image. We can extract a mapping vector set from a fractal code without special processes. Hence, we name this set of mapping vectors the *Vectorized Fractal Code*.

In PIFS encoding, image partitioning uses variable size of range region for the compression, and the size of a range region changes with local feature of an image. It means the character of the contractive mapping changes with range sizes. Mapping vectors of small range region size tend to express regions with rapid change such as edges or textures regions.

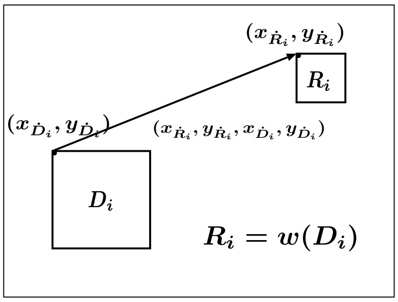


Fig. 3. A mapping vector represents the relation of the similar regions by a 4-dimensional vector $(x_{\dot{R}_i}, y_{\dot{R}_i}, x_{\dot{D}_i}, y_{\dot{D}_i})$.

Mapping vectors of large range size tend to express regions with moderate change such as sky, walls, and clouds. Hence, we classify the mapping vector sets according to the size of a range region.

C. Similarity Definition

We consider two images I_A, I_B, and A and B denote vectorized fractal codes from these images. A is divided into subsets of mapping vectors according to the region size r as $A = \{A_{r_{\min}}, \ldots, A_{r_{\max}}\}$. Here, $r_{\min}$ and $r_{\max}$ expresses the minimum and maximum for the range region size contained in I_A respectively. A subset of A associated with the region of size r is expressed by $A_r = \{a_1^r, ..., a_{|A_r|}^r\}$. Each element (i.e. a vector) in A_r is described as $a_i^r = (x_{\dot{R}_i^r}, y_{\dot{R}_i^r}, x_{\dot{D}_i^r}, y_{\dot{D}_i^r})$. Here, i denotes the sequence number of a vector, and $|X|$ denotes the cardinal number of X. These definitions are also applied to the image I_B.

Next, we introduce the similarity measure between the two vectorized fractal codes A and B. Especially, we intend to define the similarity that preserves the spatial information of vectors. For each mapping vector a_i^r in A_r, the corresponding vector $f_{B_r}(a_i^r)$ in B_r is determined by the formula

$$f_{B_r}(a_i^r) = \arg\min_{b_j^r \in B_r} \|a_i^r - b_j^r\|. \quad (3)$$

Here, $||\cdot||$ is norm. A vector in B_r that is the closest to a_i^r is selected as its corresponding vector. Analogously, the corresponding vectors for the set A_r, denoted by $f(A_r)$, are determined like (3).

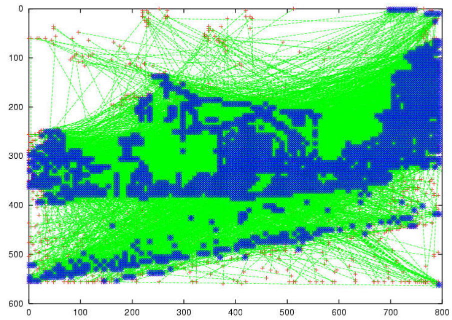
At this time, the following formula expresses the degree of the similarity between A_r and B_r. In this representation, if A_r and B_r are similar, one unique similar vector in B_r is associated with each vector in A_r.

$$s(A_r, B_r) = \frac{|f_{B_r}(A_r)|}{|A_r|} \quad (4)$$

However, this definition of the similarity is asymmetric, and in case $|A_r| \ll |B_r|$, the similarity tends to become high incorrectly. To overcome this problem, we adapt the following formula as the definition of the similarity which performs well both for the cases $|A_r| \ll |B_r|$ and $|A_r| \gg |B_r|$.

$$s^*(A_r, B_r) = \frac{|f_{B_r}(A_r)| + |f_{A_r}(B_r)|}{|A_r| + |B_r|} \quad (5)$$

(a) Original image.



(b) Mapping vectors.

Fig. 4. Mapping vectors. Each line represents the vector which represents the relation of similar regions.

Finally, the similarity between two vectorized fractal codes A and B are calculated as the weighted sum of the similarities of every range size.

$$S(A, B) = \sum_{r=r_{\min}}^{r_{\max}} \alpha_r s^*(A_r, B_r) \quad (6)$$

Here, α_r is the weighted value of the similarity s^* with range size r, and satisfies the conditions, $\sum_{r=r_{\min}}^{r_{\max}} \alpha_r = 1, \alpha_r \geq 0$.

IV. EXPERIMENTS

In this section, we show several experimental results using our retrieval method with the proposed similarity, and make clear its natures and performance. We prepare an image database and several fractal code databases. The image database consists of 1,264 gray scale images which contains natural sceneries, buildings, dolls, and so on. The fractal code database is made from this image database. We use the Mars fractal codec software [12] which was written by Mario Polvere. In this software, the value of the reduction coefficient of a contractive mapping about the space is 0.5, which means that the size of a domain region is twice a range size.

We implement the proposed similarity and retrieval method by C language with gcc 2.9.53 on linux 2.4.2. The computational cost of (3) occupies large weight in the retrieval process.

Eq. (3) corresponds to the nearest neighbor search from a vectorized fractal code. Therefore, it is expected that index structures reduce this calculation cost. We use the k-d tree [13] which provides effective nearest neighbor search for low-dimensional data sets. The average retrieval time without the k-d tree is 221.1 seconds per one query image, whereas the average retrieval time introducing the k-d tree is 77.1 seconds.

We evaluate the performance of proposed similarities from the retrieved images for the query image. We identify 7 similar image groups (91 images) from the image database, and define that the set of relevant images for the query image as the same similar image group. We use two performance measures, $Precision$ and $Recall$ formalized as (7) and (8).

$$Precision = \frac{|Retrieved \cap Relevant|}{|Retrieved|} \quad (7)$$

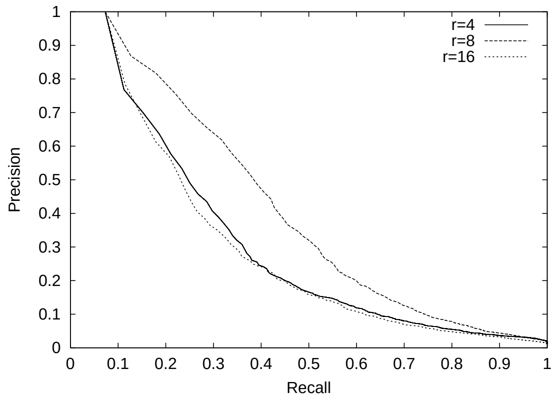
$$Recall = \frac{|Retrieved \cap Relevant|}{|Relevant|} \quad (8)$$

Here, $Retrieved$ is a set of retrieved images for a query image, and $Relevant$ is a set of relevant images for a query image. $Precision$ shows the rate of the retrieved relevant images against all the retrieved images. $Recall$ shows the rate of the retrieved relevant images against all the relevant images in the whole database.

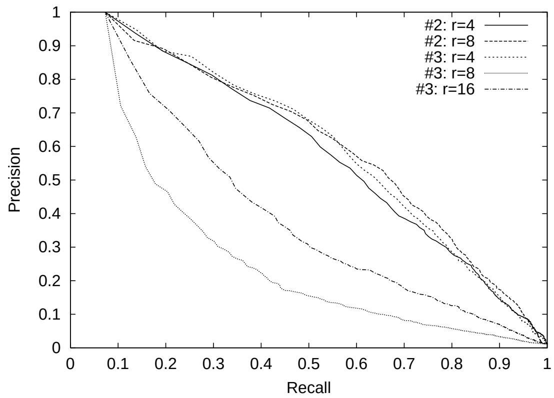
First, we present the performance of the similarity s^* for a specific range size in order to decide the proper value of α_r in S. The similarity S is combined with different range sizes' s^*. We confirm the performance of the similarity s^*, and decide the α_r from this result. For this purpose, we prepare the five compressed code databases. In the three of five databases, fractal codes have only a single range size. In the two remaining databases, fractal codes have multiple range sizes.

Fig. 5(a) shows the average precision-recall curves as the performances of s^* in the three databases. This result shows that s^* obtained with a single range size is not adequate for retrieval. Fig. 5(b) shows the performance of s^* in the two remaining databases. In this figure, "#2" denotes the database which consists of two range sizes (for $r_{\min} = 4$ and for $r_{\max} = 8$). "#3" denotes the database which consists of three range sizes (for $r_{\min} = 4$, $r = 8$ and $r_{\max} = 16$). In the #2 database, the similarity s^* for $r = 4$ and for $r = 8$ have stable higher performance. In the #3 database, the similarity s^* for $r = 4$ have higher performance than for $r = 8$ and $r = 16$. s^* for $r = 4$ in the both databases represents high retrieval performance than others.

The linear combination of s^* forms S. Because the performance of both s^* ($r = 4, r = 8$) in #2 are almost equivalently stable, we decide the weighted values of the similarity S are as follows: $\alpha_4 = 0.5$, $\alpha_8 = 0.5$. Fig. 6 shows each similarity s^* and S which consists of the similarities s^*. From this result, the combination of s^* is proved to improve the retrieval performance. Fig. 7 shows the examples of retrieval results with the similarity S. In this figure, the top left images of each group are query images and all the images are the retrieved images including the query image. We confirmed the usefulness of the similarity S and we can say the retrieval based on the similarity S works properly.



(a) Performance under the similarity s^* in the database which has a single range size.



(b) Performance under the similarity s^* in the database which has multiple range sizes. "#2" represents that fractal codes in this database has two range sizes. "#3" represents that fractal codes in this database has three range sizes

Fig. 5. Performance under the similarity s^* in the five fractal code databases.

Next, we demonstrate the robustness of the similarity S between distorted images and original image. We choose an image from the image database, and make some distorted images (scaling, rotations, translations) from this image. Fig. 8 shows such distorted images.

Fig. 9(a) shows the similarity between the original and scaling images. In this figure, "other image" shows the highest similarity between the original image and other images in the database excluding the distorted images. From this result, we confirm the distorted images have the highest similarity when the rates of scaling are not greater than $\pm 50\%$ from the original image. We also confirmed that the similarity is robust to translations and rotations. Fig. 9(b) and 9(c) show the distorted images have the highest similarity to the original image until the images are translated up to 10% of the image width or rotated up to 20 degree of rotation angles from the

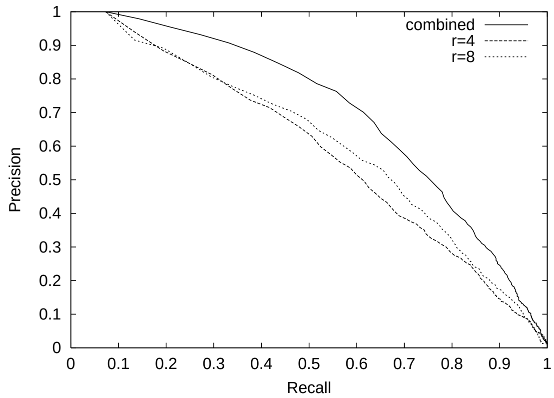


Fig. 6. Performances of the similarities in the #2 database. "$r = 4$" and "$r = 8$" represent the performance of s^*. "combined" represents a combined similarity S ($\alpha_4 = 0.5, \alpha_8 = 0.5$).

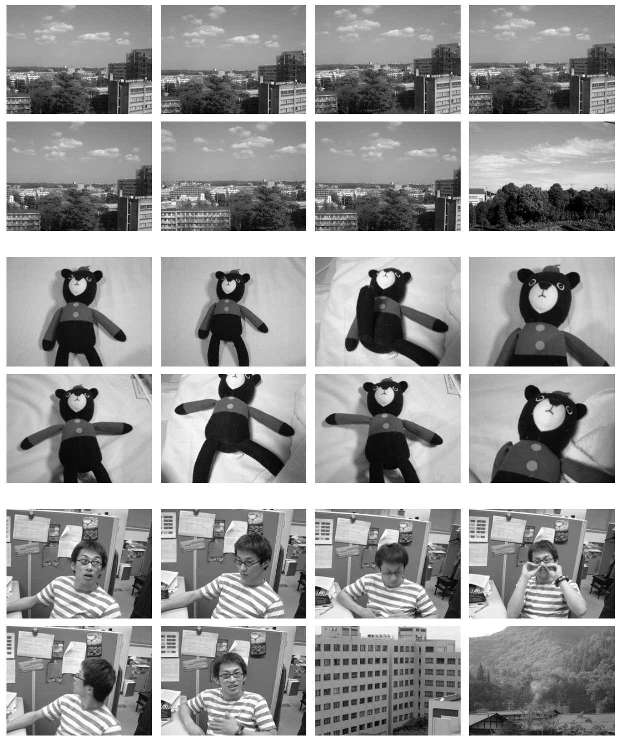


Fig. 7. Examples for retrieval results with the similarity S. The top left image in each group is the query, and the eight images including the query are retrieved.

original image.

Finally, we compare the retrieval performance of the proposed method using the similarity S with two other methods, i.e. Fast Multiresolution Image Querying [14] and Histogram Distance. Fast multiresolution image querying is one of the basic retrieval methods using wavelet transform. Histogram distance which is the most popular retrieval method for uncompressed image data. In this experiment, we use the discrete 8-bin histogram of gray-scale and L_1-norm.

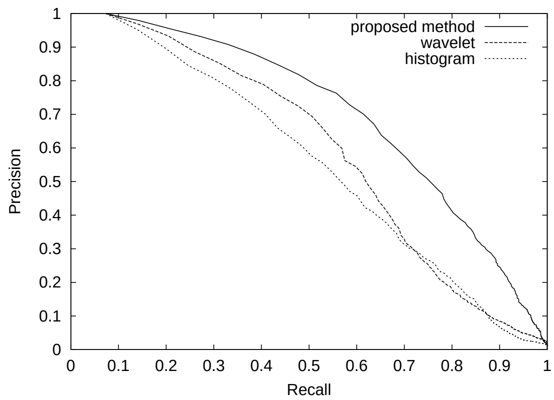


Fig. 10. Precision ratio vs. recall ratio: Proposed similarity S, Wavelet method and Histogram distance.

Fig. 10 shows the average precision-recall curves of our method and others. This result presents the proposed method with the similarity S outperforms two methods.

V. Conclusion

In this paper, we proposed the retrieval method with the similarity measure based on fractal codes. A fractal code expresses the similar relations between the regions in an image by contractive mappings. We treat the fractal code as the set of mapping vectors, and define the similarity based on the rate of corresponding vectors between two vectorized fractal codes. This similarity exploits the spatial relations which contractive mappings of fractal codes inherent have. The effectiveness of the proposed similarity was confirmed through some experiments.

References

[1] M. K. Mandal, F. Idris, and S. Panchanathan, "A critical evaluation of image and video indexing techniques in the compressed domain," *Image and Vision Computing*, vol. 17, no. 7, pp. 513–529, May 1999.

[2] M. F. Barnsley, *Fractals everywhere*. San Diego: Academic Press, 1993, 1988.

[3] A. Zhang, B. Cheng, R. S. Acharya, and R. P. Menon, "Comparison of wavelet transforms and fractal coding in texture-based image retrieval," in *Proc. Visual Data Exploration and Analysis III 2565*. SPIE, 1996, pp. 116–125.

[4] T. Tan and H. Yan, "The fractal neighbor distance measure," *Pattern Recognition*, vol. 33, no. 6, pp. 1371–1387, 2002.

[5] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. on Image Processing*, vol. 1, pp. 18–30, Jan 1992.

[6] B. Wohlberg and G. de Jager, "A review of the fractal image coding literature," *IEEE Trans. on Image Processing*, vol. 8, no. 12, pp. 1716–1729, December 1999.

[7] L. Ning, *Fractal Imaging*. Academin Press, 1997.

[8] T. Ida and Y. Sanbonsugi, "Image segmentaion using fractal coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 567–570, 1995.

[9] T. Ida and Y. S., "Self-affine mapping system and its application to object contour extraction," *IEEE Trans. on Image Processing*, vol. 9, no. 11, pp. 1926–1936, Nov 2000.

[10] M. Haseyama and I. Kondo, "Image authentication based on fractal image coding without contamination of original image," *Journal of IEICE*, vol. J85-D-II, no. 10, 2002.

[11] Y. Fisher, Ed., *Fractal image compression: theory and application*. Springer-Verlag New York, Inc., 1995.

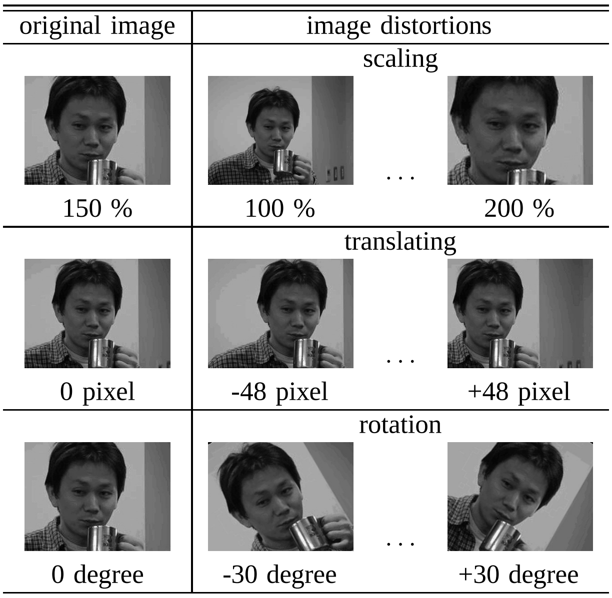
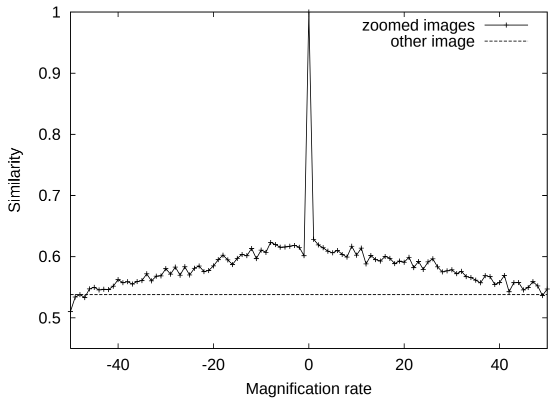
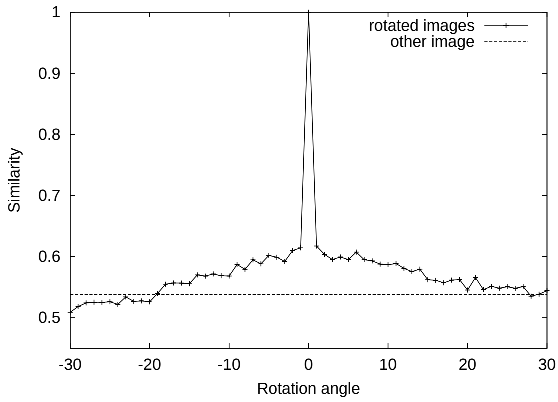


Fig. 8. Image distortions. The left columns in the table show original images before distortion. The right columns show various distorted images.

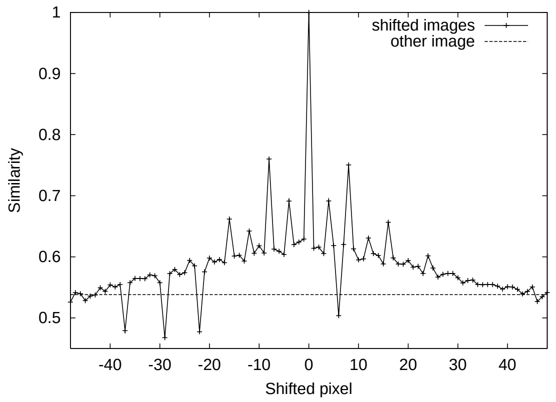
[12] `http://inls.ucsd.edu/~fisher/Fractals/`.
[13] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications ACM*, vol. 18, no. 9, pp. 509–517, September 1975.
[14] C. E. Jacobs, A. Finkelstein, and D. H. Salesin, "Fast multiresolution image querying," in *Proc. SIGGRAPH 95*, aug 1995, pp. 277–286.



(a) Similarity vs. scaled images.



(b) Similarity vs. rotated images.



(c) Similarity vs. shifted images.

Fig. 9. The proposed similarity S ($\alpha_4 = 0.5$, $\alpha_8 = 0.5$) between distorted images and the original image.