2011 International Symposium on Nonlinear Theory and its Applications
NOLTA2011, Kobe, Japan, September 4-7, 2011

Similarity-based Image Retrieval Considering Color of Artifacts by Self-Organizing Map with Refractoriness

†Minato KOBAYASHI and ‡Yuko OSANA

†Graduate School of Bionics, Computer and Media Sciences, Tokyo University of Technology
‡School of Computer Science, Tokyo University of Technology
1404-1 Katakura, Hachioji, Tokyo, 192-0982, Japan Email: †minato@osn.cs.teu.ac.jp, ‡osana@cs.teu.ac.jp

Abstract—In this paper, we propose a similarity-based image retrieval considering color of artifacts by self-organizing map with refractoriness. In the self-organizing map with refractoriness, the plural neurons in the Map Layer corresponding to the input can fire sequentially because of the refractoriness. The proposed system makes use of this property in order to retrieve plural similar images. We carried out a series of computer experiments and confirmed the effectiveness of the proposed system.

1. Introduction

Recently, some similarity-based image retrieval systems which make use of the flexible information processing ability of neural networks have been proposed[1]-[5]. Most of these systems uses color information as feature, the image retrieval for scenery images can be realized. They can retrieve scenery images. However, it is difficult for them to retrieve images including artifacts. In such case, if areas including artifacts can be recognized, we can expect that the retrieval accuracy can be improved[5][6]. In these systems, the original image is divided into some areas based on spatial distribution of colors and the color information in each area is used as image feature. However, color information of artifacts are not used for retrieval.

In this paper, we propose a similarity-based image retrieval considering color of artifacts by self-organizing map with refractoriness. In the self-organizing map with refractoriness, the plural neurons in the Map Layer corresponding to the input can fire sequentially because of the refractoriness. The proposed system makes use of this property in order to retrieve plural similar images. In this image retrieval system, as the image feature, not only color information but also spectrum and keywords are employed.

2. Self-Organizing Map with Refractoriness

Here, we explain the self-organizing map with refractoriness[2] which is used in the proposed system.

2.1. Structure

The self-organizing map with refractoriness is composed of (1) Input Layer and (2) Map Layer as similar as the conventional self-organizing map[7].

2.2. Dynamics

In the self-organizing map with refractoriness, when the pattern $\boldsymbol{x}$ is given to the Input Layer, the internal state of the neuron i in the Map Layer, $u_i(t+1)$ is given by

$$u_i(t+1) = 1 - \frac{D(\boldsymbol{w}_i, \boldsymbol{x})}{\sqrt{N}} - \alpha \sum_{d=0}^{t} k_r^d x_i^{MAP}(t-d) \quad (1)$$

where $\boldsymbol{w}_i$ is the connection weights to the neuron i in the Map Layer, $\boldsymbol{x}$ is the input vector, N is the number of neurons in the Input Layer, α is the scaling factor of refractoriness, k_r is the damping factor of refractoriness, $x_i^{MAP}(t)$ is the output of the neuron i in the Map Layer at the time t, and $D(\cdot)$ is the Euclid distance.

The output of the neuron i in the Map Layer at the time t, $\boldsymbol{x}_i^{MAP}(t)$ is given by

$$x_i^{MAP}(t) = \begin{cases} 1 & (i = c) \\ 0 & (i \neq c) \end{cases} \quad (2)$$

$$c = \underset{i}{\operatorname{argmax}}\{u_i(t); i = 1, 2, \ldots, M\} \quad (3)$$

where M is the number of neurons in the Map Layer.

2.3. Learning

The learning process of the self-organizing map with refractoriness is based on the conventional self-organizing map[7]. After the learning based on the conventional self-organizing map, the connection weights are updated and fixed again so that each neuron in the Map Layer corresponds to one data (image)[2].

3. Image Features

In the proposed system, the color information, spectrum and keywords which are used as image features.

3.1. Color Information

Each image are divided into $N_{H1} \times N_{W1}$ rectangle areas, and normalized average x, y and z coordinates in the HSV color space on each area are used as image feature. If the area includes artifacts, the feature value minus one is used.

3.2. Color Information in each Area

An image is divided into some areas by the K-means algorithm. Then, normalized average x, y and z coordinates in the HSV color space on each area are trained in the self-organizing map, and are used as image feature. Here, two self-organizing maps for natural objects and artifacts are used.

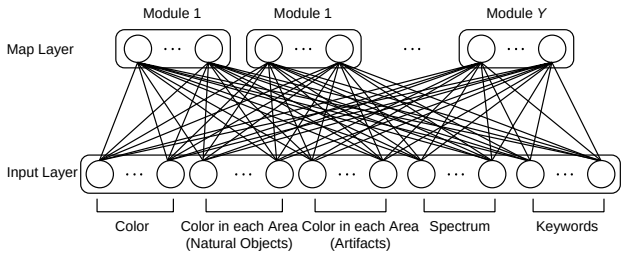


Figure 1: Structure of Proposed Image Retrieval System.

3.3. Spectrum

In the proposed system, an image spectrum is used as image features. In the proposed system, the calculated spectrum is divided into $N^k \times N^l$ areas, and average in each area is used as image features.

3.4. Keywords

Keywords such as sky, cloud, tree are used as query in addition to color information and spectrum.

4. Similarity-based Image Retrieval Considering Colors of Artifacts using Self-Organizing Map with Refractoriness

4.1. Structure

The structure of the proposed system is shown in Fig.1. As shown in this figure, the proposed system has the Input Layer which has some parts corresponding to image features such as color, spectrum and keywords and the Map Layer composed of some modules. In the proposed system, each neuron in the Map Layer corresponds to one of the stored images.

4.2. Learning Process

The learning process of the proposed system has 3 steps.

Step 1 : Extraction of Artifacts

The original image is divided into some regions by the K-means algorithm, and whether or not artifacts are included for each divided area is judged.

Step 2 : Extraction of Image Features

The color information and spectrum are extracted from the images to be stored. And keywords are set to each image manually.

Step 3 : Learning of Self-Organizing Map with Refractoriness

Image features which are extracted in **Step 2** are trained. The proposed system is trained based on the learning algorithm described in **2.3**.

4.3. Image Retrieval Process

(1) Generation of Feature Vector

The feature vector is generated for the key image based on the search request which is selected by a user.

Search Request 1 : Retrieval considering Artifacts Position

Search Request 2 : Retrieval considering Areas without Artifacts

Search Request 3 : Retrieval considering Artifacts

(2) Image Retrieval

The image retrieval process in the proposed system has 4 steps.

Step 1 : Input of Feature Vector

Image features of the key image such as color information, spectrum, and keywords are given to the Input Layer.

Step 2 : Calculation of Internal States of Neurons in Map Layer When the image features of the key image are given to the Input Layer, the internal state of the neuron i of the module y in the Map Layer at the time $t+1$, $u_i^y(t+1)$ is calculated by

$$u_i^y(t+1) = 1 - \frac{D_r(\boldsymbol{w}_i^y, \boldsymbol{x})}{\sqrt{F'}} - \alpha \sum_{d=0}^{t} k_r^d x_i^{MAP(y)}(t-d) \qquad (4)$$

where F' is the number of image features, α is the scaling factor, k_r is the damping factor, and $x_i^{MAP(y)}(t)$ is the output of the neuron i of the module y in the Map Layer at the time t. $D_r(\boldsymbol{w}_i^y, \boldsymbol{x})$ is the distance between the weight vector of the neuron i of the module y in the Map Layer $\boldsymbol{w}_i^y$ and the input $\boldsymbol{x}$, and is given by

$$D_r(\boldsymbol{w}_i^y, \boldsymbol{x}) = \sqrt{\sum_{f=1}^{F} \mu(f) \sum_{j=1}^{N} (\phi(w_{ij}^y, x_j))^2} \qquad (5)$$

where $f(=1, \cdots, F)$ is an image feature (1: color, 2: color in each region (natural objects), 3: color in each region (artifacts) 4: spectrum, 5: keywords). N is the number of neurons in the Input Layer, w_{ij}^y is the connection weight from the neuron j in the Input Layer to the neuron i of the module y in the Map Layer, and x_j is the jth element in the input feature vector. $\mu(f)$ is the weighting coefficient, and it is given by

$$\mu(f) = \begin{cases} \dfrac{1}{N'^{(f)}} & (N'^{(f)} > 0) \\ 0 & (N'^{(f)} = 0) \end{cases} \qquad (6)$$

where $N'^{(f)}$ is the number of neurons which correspond to the feature f and are used in the calculation of the distance.

In the Search Request 1, $\phi(w_{ij}^y, x_j)$ is given by

$$\phi(w_{ij}^y, x_j) = \begin{cases} h_1(w_{ij}^y) - h_2(x_j) & \\ \qquad (x_j \in C_{WC} \text{ and } h_1(w_{ij}^y) \cdot h_2(x_j) \geq 0) & \\ 1 \qquad (x_j \in C_{WC} \text{ and } h_1(w_{ij}^y) \cdot h_2(x_j) < 0) & \\ k_w \quad \Bigg(x_j \in C_K \text{ and } \Bigg((w_{ij}^y = 0 \text{ and } x_j = 1 \text{ and} & \\ \qquad \sum_{k:x_k \in C_K^{g_j}} w_{ik}^y > 0) \text{ or } (w_{ij}^y = 1 \text{ and } x_j = 0 & \\ \qquad \text{and } \sum_{k:x_k \in C_K^{g_j}} x_k > 0) \Bigg) \Bigg) & \\ 0 & (x_j \in C_{AC}) \\ w_{ij}^y - x_j & (\text{otherwise}) \end{cases} \qquad (7)$$

where C_{WC} is the neuron set corresponding to color information, C_{AC} is the neuron set corresponding to the color information in each region (artifacts), C_K is the neuron set corresponding to the keyword, $C_K^{g,j}$ is the neuron set corresponding to the keyword which belongs to the same group to j, and k_w $(0 < k_w < 1)$ is the constant. In Eq.(7), $h_1(w_{ij}^y)$ is given by

$$h_1(w_{ij}^y) = \begin{cases} w_{ij}^y & (0 \le w_{ij}^y) \\ -1 & \text{(otherwise)}. \end{cases} \tag{8}$$

And $h_2(x_j)$ is given by

$$h_2(x_j) = \begin{cases} x_j & (f_j = 0) \\ -1 & (f_j = 1) \end{cases} \tag{9}$$

where f_j is the flag for the neuron j. If the region corresponding to the neuron j includes artifacts, f_j is set to 1.

In the Search Request 2, $\phi(w_{ij}^y, x_j)$ is given by

$$\phi(w_{ij}^y, x_j) = \begin{cases} k_w & \left(x_j \in C_K \text{ and } \left(w_{ij}^y = 0 \text{ and } x_j = 1 \text{ and } \sum_{k:x_k \in C_K^{g_j}} w_{ik}^y > 0\right) \text{ or } \left(w_{ij}^y = 1 \text{ and } x_j = 0 \text{ and } \sum_{k:x_k \in C_K^{g_j}} x_k > 0\right)\right) \\ 0 & \left(\left(x_j \in C_{WC} \text{ and } (f_j = 1 \text{ or } w_{ij}^y < 0)\right) \text{ or } x_j \in C_{AC}\right) \\ w_{ij}^y - x_j & \text{(otherwise)}. \end{cases} \tag{10}$$

In the Search Request 3, $\phi(w_{ij}^y, x_j)$ is given by

$$\phi(w_{ij}^y, x_j) = \begin{cases} w_{ij}^y - x_j & (x_j \in C_{WC} \text{ and } w_{ij}^y \cdot x_j \ge 0) \\ 1 & (x_j \in C_{WC} \text{ and } w_{ij}^y \cdot x_j < 0) \\ k_w & \left(x_j \in C_K \text{ and } \left(\left(w_{ij}^y = 0 \text{ and } x_j = 1 \text{ and } \sum_{k:x_k \in C_K^{g_j}} w_{ik}^y > 0\right) \text{ or } \left(w_{ij}^y = 1 \text{ and } x_j = 0 \text{ and } \sum_{k:x_k \in C_K^{g_j}} x_k > 0\right)\right)\right) \\ w_{ij}^y - x_j & \text{(otherwise)}. \end{cases} \tag{11}$$

In each module, the neuron whose internal state calculated by Eq.(4) is selected as the winner neuron $c^{(y)}$.

Step 3 : Calculation of Outputs of Neurons in Map Layer

The output of the neuron i of the module y in the Map Layer at the time t, $x_i^{MAP(y)}(t)$ is calculated by

$$x_i^{MAP(y)}(t) = \begin{cases} 1, & (i = c^{(y)}, u_i^y(t) > \theta_{s1} \text{ and } D_{max}^{y(i)} < \theta_{s2}) \\ 0, & \text{(otherwise)} \end{cases} \tag{12}$$

where $c^{(y)}$ is the winner neuron in the module y, θ_{s1} and θ_{s2} are the thresholds. $D_{max}^{y(i)}$ is the maximum distance for the image feature in the neuron i of the module y, and is given by

$$D_{max}^{y(i)} = \max_f \left(\mu(f) \sum_{j=1}^{N} (\phi(w_{ij}^y, x_j))^2\right). \tag{13}$$

In the proposed system, each stored image corresponds to a neuron in the Map Layer. So, the images corresponding to the fired neurons in the Map Layer are output.

Step 4 : Repeat

Steps 2 and **3** are repeated.

5. Computer Experiment Results

Figures 2 and 3 show the image retrieval results of the proposed system. As shown in these figures, the proposed system can retrieve similar images even when the key image includes artifacts.

Table 1 shows the F-measure of the retrieval results for 500 key images.

6. Conclusions

In this research, we have proposed the similarity-based image retrieval considering artifacts by self-organizing map with refractoriness. We carried out a series of computer experiments and confirmed that the proposed system can retrieve images more appropriately than the conventional system.

References

[1] J. T. Laaksonen, J. M. Koskela, S. P. Laakso and E. Oja: "PicSOM—content-based image retrieval with self-organizing maps," Pattern Recognition Letters, Vol.21, No.13, 14, pp.1199–1207, 2000.

[2] H. Mogami, M. Otake, N. Kouno and Y. Osana : "Self-organizing map with refractoriness and its application to image retrieval," Proceedings of IEEE International Joint Conference on Neural Networks, Vancouver, 2006.

[3] K. Nagashima, M. Nakada and Y. Osana: "Similarity-based image retrieval by self-organizing map with refractoriness," Proceedings of IEEE International Joint Conference on Neural Networks, Orlando, 2007.

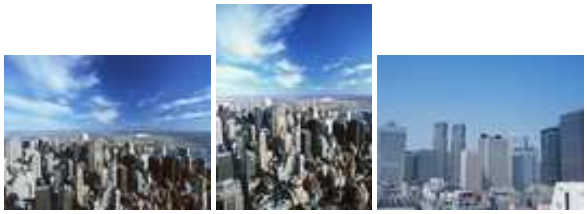
[4] H. Tanimizu and Y. Osana: "Similarity-based image retrieval from plural key images by self-organizing map with refractoriness," Proceedings of IEEE and INNS International Joint Conference on Neural Networks, Hong Kong, 2008.

[5] D. Hanagaki and Y. Osana : "Similarity-based image retrieval considering artifacts by self-organizing map with refractoriness," Proceedings of IEEE and INNS International Joint Conference on Neural Networks, Atlanta, 2009.

[6] S. Hayashida and Y. Osana : "Similarity-based image retrieval considering artifacts by self-organizing map with refractoriness," IEICE Technichal Report, NC, 2010 (in Japanesse).

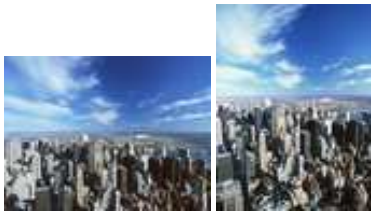
[7] T. Kohonen: Self-Organizing Maps, Springer, 1994.

(a) Key Image



(b) Search Request 1

(c) Search Request 2



(d) Search Request 3

Figure 2: Image Retrieval Results (1).

[8] J. B. MacQueen : "Some methods for classification and analysis of multivariate observations," Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Vol.1, No.297, University of California Press, 1967.

(a) Key Image

(b) Search Request 1

(c) Search Request 2

(d) Search Request 3

Figure 3: Image Retrieval Results (1).

Table 1: F-measure.

	Precision	Recall	F-measure
Search Request 1	0.978	0.771	0.862
Search Request 2	0.801	0.673	0.731
Search Request 3	0.968	0.783	0.866