IEICE Proceeding Series

Similarity-based Image Retrieval considering Artifacts from Plural Key Images by Self-Organizing Map with Refractoriness

Yu MORIMOTO, Yuko OSANA

Vol. 1 pp. 9-12
Publication Date: 2014/03/17
Online ISSN: 2188-5079

2012 International Symposium on Nonlinear Theory and its Applications
NOLTA2012, Palma, Majorca, Spain, October 22-26, 2012

Similarity-based Image Retrieval considering Artifacts from Plural Key Images by Self-Organizing Map with Refractoriness

†Yu MORIMOTO 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: †morimoto@osn.cs.teu.ac.jp, ‡osana@cs.teu.ac.jp

Abstract—In this paper, we propose a similarity-based image retrieval from plural key images by self-organizing map with refractoriness. Most of the conventional image retrieval systems can retrieve only from one key image. In contrast, the proposed system can retrieve similar images which have common features in plural key 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]-[6]. Most of these systems use 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]. However, in these systems, only one image is used as the key. Although the system which can retrieve from plural key images[4] has been proposed, it can only apply to scenery images without artifacts.

In this paper, we propose a similarity-based image retrieval considering artifacts by self-organizing map with refractoriness from plural key images. 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.

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

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

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 Euclidean distance.

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

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

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

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

The learning process of the self-organizing map with refractoriness has 10 steps.

Step 1 : Initialization of Connection Weights

The initial values of weights are chosen randomly.

Step 2 : Calculation of Internal States of Neurons in Map Layer

When the pattern $\boldsymbol{x}$ is given to the Input Layer, the internal state of the neuron i in the Map Layer is calculated by

$$u_i = 1 - \frac{D(\boldsymbol{w}_i, \boldsymbol{x})}{\sqrt{N}} \quad (4)$$

Step 3 : Calculation of Outputs of Neurons in Map Layer

The output of the neuron i in the Map Layer, $\boldsymbol{x}_i^{MAP}$ is calculated by Eq.(2).

Step 4 : Update of Connection Weights

The connection weights $\boldsymbol{w}_i$ are updated by

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + \eta \cdot \exp\left(\frac{\sqrt{(x_i^x - x_c^x)^2 + (x_i^y - x_c^y)^2}}{\delta(t)^2}\right)(\boldsymbol{x} - \boldsymbol{w}_i(t)) \quad (5)$$

where η is the learning rate, x_i^x, x_i^y are the coordinates of the neuron i in the Map Layer and x_c^x, x_c^y are the coordinates of the winner neuron in the Map Layer. $\delta(t)$ is the neighborhood function at the time t, and it is given by

$$\delta(t) = \delta^{ini} \cdot \left(\frac{\delta^{fin}}{\delta^{ini}}\right)^{\frac{t}{T}} \quad (6)$$

where δ^{ini} is the initial size of neighborhood area, δ^{fin} is the final size of neighborhood area and T is the maximum learning time.

Step 5 : Repeat

Steps 2~4 are repeated T times.

Step 6 : Decision of Winner Neuron

The winner neuron c whose internal state given by Eq.(4) is maximum is found.

Step 7 : Update of Connection Weights

If the internal state of the winner neuron c is smaller than the threshold θ_d, the connection weights except those of fixed neurons are changed by

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + \eta^{fin} \cdot (\boldsymbol{x}^{(p)} - \boldsymbol{w}_i(t)) \quad (7)$$

where $\boldsymbol{x}^{(p)}$ is the stored pattern p, and η^{fin} is given by

$$\eta^{fin} = \eta \cdot \exp\left(\frac{-((x_i^x - x_c^x)^2 + (x_i^y - x_c^y)^2)}{\delta^{fin 2}}\right). \quad (8)$$

Step 8 : Repeat

Step 7 is repeated until the internal state of the winner neuron becomes larger than the threshold θ_d.

Step 9 : Fix of Connection Weights

The connection weights of the winner neuron c, $\boldsymbol{w}_c$ are fixed.

Step 10 : Repeat

Steps 6~9 are repeated until all patterns are trained.

3. Image Features

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

3.1. Color

In the proposed system, each image is divided into some regions by the K-means algorithm[8]. Then, normalized average x, y and z coordinates in the HSV color space on each region 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. In the proposed system, the features on color are calculated per nine areas (See Fig.1 and Table 1).

The feature vector on the color of the natural objects at the area s in the image p, $\boldsymbol{x}^{N(p,s)}$ is given by

$$x_i^{N(p,s)} = g\left(\sum_{l \in C_s^N} x_i^{N(p,s,l)}\right) \quad (9)$$

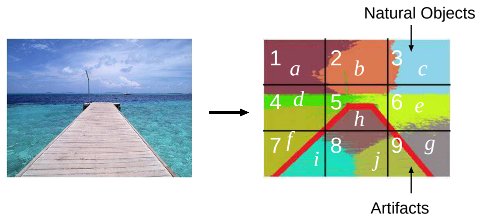


Figure 1: Regions divided by K-means and 9 Areas.

Table 1: Relation between Regions and 9 Areas.

Area No.	Natural Objects	Artifacts
1	a, b	
2	a, b	
3	b, c	
4	a, d, f	
5	a, b, d, e, f	h
6	b, c, e, g	
7	f	i
8		h, i, j
9	g	j

$$g(u) = \begin{cases} 1 & (u > 0) \\ 0 & (u = 0) \end{cases} \quad (10)$$

where C_s^N is the set of the regions which belong to the area s for the natural objects, $x_i^{N(p,s,l)}$ is the output of the neuron i in the self-organizing map which learns the color of the natural objects when the color information at the region l which belongs to the area s in the image p is given.

The feature vector on the color of the artifacts at the area s in the image p, $\boldsymbol{x}^{A(p,s)}$ is given by

$$x_i^{A(p,s)} = g\left(\sum_{l \in C_s^A} x_i^{A(p,s,l)}\right) \quad (11)$$

where C_s^A is the set of the regions which belong to the area s for the artifacts, $x_i^{A(p,s,l)}$ is the output of the neuron i in the self-organizing map which learns the color of the artifacts when the color information at the region l which belongs to the area s in the image p is given.

3.2. Spectrum

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

3.3. Keywords

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

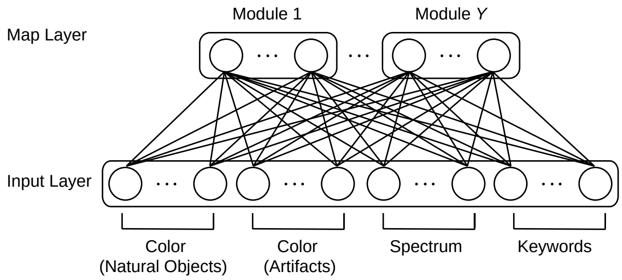


Figure 2: Structure of Proposed Image Retrieval System.

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

4.1. Structure

Figure 2 shows the structure of the proposed system. As shown in this figure, it 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 is judged for each divided area.

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 in the self-organizing map with refractoriness.

4.3. Image Retrieval Process

(1) Generation of Feature Vector

The feature vector is generated from the plural key images. In the proposed system, first, the feature vector for each key image $\boldsymbol{x}^{(p)}(p = 1, \cdots, N^{key})$ is generated. Here, N^{key} is the number of key images. And then, the feature vector for the plural key images $\boldsymbol{x}$ is generated from $\boldsymbol{x}^{(p)}$ as follows:

$$x_i = \begin{cases} x_i^{(p)} & \left(\displaystyle\sum_{p=1}^{N^{key}-1} \sum_{q=p+1}^{N^{key}} |x_i^{(p)} - x_i^{(q)}| = 0 \right) \\ -1 & \text{(otherwise)} \end{cases} \tag{12}$$

As shown in Eq.(12), the proposed system uses only the common features in all key images.

(2) Image Retrieval

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

Step 1 : Input of Feature Vector

Image features of the key images 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 feature of the key images is 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

$$u_i^y(t) = 1 - \frac{D_r(\boldsymbol{w}_i^y, \boldsymbol{x})}{\sqrt{F'}} - \alpha \sum_{d=0}^{t-1} k_r^d x_i^{MAP(y)}(t-d) \tag{13}$$

where F' is the number of image features which is used in the retrieval process, 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

$$D_r(\boldsymbol{w}_i^y, \boldsymbol{x}) = \sqrt{\sum_{f=1}^{F} \mu(f) \sum_{j:x_j \in C_f} (\phi(w_{ij}^y, x_j))^2} \tag{14}$$

where f $(= 1, \cdots, F)$ is an image feature (1 : color (natural objects), 2 : color (artifacts), 3 : spectrum, 4 : keywords). C_f is the set of the inputs corresponds to the feature f, 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

$$\mu(f) = \begin{cases} \dfrac{1}{N'^{(f)}} & (N'^{(f)} > 0) \\ 0 & (N'^{(f)} = 0) \end{cases} \tag{15}$$

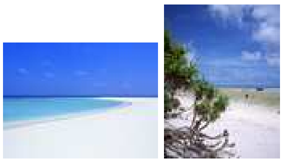
where $N'^{(f)}$ is the number of neurons which correspond to the feature f and is not equal -1.

$\phi(w_{ij}^y, x_j)$ is given by

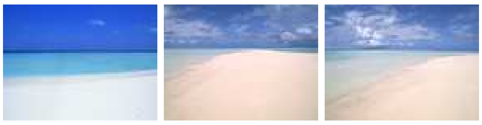
$$\phi(w_{ij}^y, x_j) = \begin{cases} k_w & \Bigg(x_j \in C_4 \text{ and } \bigg(\Big(w_{ij}^y = 0 \text{ and } x_j = 1 \\ & \quad \text{and} \displaystyle\sum_{k:x_k \in C_4^{g_j}} w_{ik}^y > 0 \Big) \text{ or } \Big(w_{ij}^y = 1 \\ & \quad \text{and } x_j = 0 \text{ and} \displaystyle\sum_{k:x_k \in C_4^{g_j}} x_k > 0 \Big) \bigg) \Bigg) \\ 0 & (x_j = -1) \\ w_{ij}^y - x_j & \text{(otherwise)} \end{cases} \tag{16}$$

where C_4 is the neuron set corresponding to keywords, $C_4^{g_j}$ is the neuron set corresponding to the keywords which belong to the same group to j, and k_w $(0 < k_w < 1)$ is the constant.

In each module, the neuron whose internal state calculated by Eq.(13) is maximum is selected as the winner neuron $c^{(y)}$.

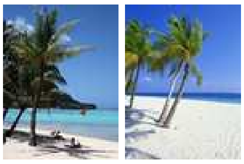


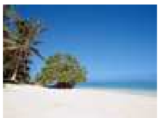
(a) Key Images



(b) Result

Figure 3: Image Retrieval Results (1).





(a) Key Images (b) Result

Figure 4: Image Retrieval Results (2).

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

$$x_i^{MAP(y)}(t) = \begin{cases} 1, & (i = c^{(y)},\ u_i^y(t) > \theta_{s1} \text{ and } D_{max}^{y(i)} < \theta_{s2}) \\ 0, & (\text{otherwise}) \end{cases} \quad (17)$$

where $c^{(y)}$ is the winner neuron in the module y, and 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

$$D_{max}^{y(i)} = \max_f \left(\mu(f) \sum_{j=1}^{N} (\phi(w_{ij}^y, x_j))^2 \right). \quad (18)$$

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.

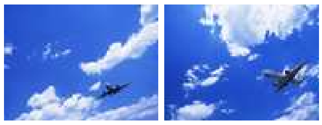
5. Computer Experiment Results

Figures 3 ~ 5 show the image retrieval results of the proposed system. As shown in these figures, the proposed system can retrieve similar images which have common features in the plural key images.

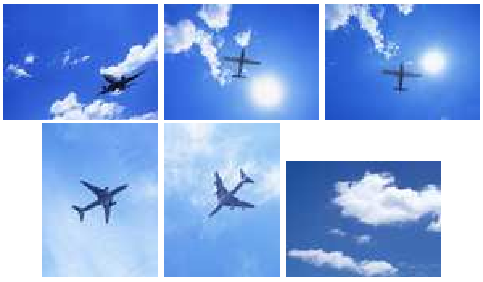
Table 2 shows the F-measure of the retrieval results for 150 key image pairs in the proposed system which stores 540 images.

6. Conclusions

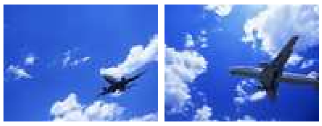
In this research, we have proposed the similarity-based image retrieval considering artifacts from plural key images by self-organizing map with refractoriness. We carried out a series of computer experiments and confirmed that the proposed system can retrieve images appropriately from plural key images.



(a) Key Images



(b) Result from Each Key Image



(c) Result from Two Key Images

Figure 5: Image Retrieval Results (3).

Table 2: F-measure.

Precision	Recall	F-measure
0.993	0.988	0.991

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] S. Hayashida and Y. Osana : "Similarity-based image retrieval considering artifacts by self-organizing map with refractoriness – Image segmentation by K-means algorithm –," Proceedings of IEEE International Conference on System, Man and Cybernetics, Anchorage, 2011.

[6] M. Kobayashi and Y. Osana : "Similarity-based image retrieval considering color of artifacts by self-organizing map with refractoriness," Proceedings of International Symposium on Nonlinear Theory and its Applications, Kobe, 2011.

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

[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.