

## CONTENT-BASED IMAGE RETRIEVAL USING A SIGNATURE GRAPH AND A SELF-ORGANIZING MAP

THANH THE VAN <sup>a,b</sup>, THANH MANH LE <sup>a</sup>

<sup>a</sup>Faculty of Information Technology  
University of Sciences/Hue University, 77 Nguyen Hue Street, Hue City, Vietnam  
e-mail: vanthethanh@gmail.com

<sup>b</sup>Center for Information Technology  
HCMC University of Food Industry, 140 Le Trong Tan Street, Tan Phu District, HoChiMinh City, Vietnam

In order to effectively retrieve a large database of images, a method of creating an image retrieval system CBIR (content-based image retrieval) is applied based on a binary index which aims to describe features of an image object of interest. This index is called the binary signature and builds input data for the problem of matching similar images. To extract the object of interest, we propose an image segmentation method on the basis of low-level visual features including the color and texture of the image. These features are extracted at each block of the image by the discrete wavelet frame transform and the appropriate color space. On the basis of a segmented image, we create a binary signature to describe the location, color and shape of the objects of interest. In order to match similar images, we provide a similarity measure between the images based on binary signatures. Then, we present a CBIR model which combines a signature graph and a self-organizing map to cluster and store similar images. To illustrate the proposed method, experiments on image databases are reported, including COREL, Wang and MSRDI.

**Keywords:** binary signature, similarity measure, signature graph, image retrieval.

### 1. Introduction

Multimedia data, especially images, are integrated more and more in various information systems such as the World Wide Web, digital libraries, geographic information systems, satellite observation systems or even police crime recording and investigation systems (Muneesawang *et al.*, 2014; Marques and Furht, 2002). The exploration of a number of camera devices and mobile devices with a camera leads to a huge amount of images created and shared everyday. The growth of image datasets raises interest not only in developing image-based applications, but also in challenging problems of retrieving images from such large image collections. They have to be automatically classified, indexed and stored in a compressed way to efficiently perform similar images queries.

*Image retrieval systems* return a set of images which are similar to an image in a query by matching their features (Acharya and Ray, 2005; Liu *et al.*, 2007; Liu and Yang, 2013; Muneesawang *et al.*, 2014; Alzu'bi

*et al.*, 2015). There are two well-known techniques used in many systems: text-based and content-based (Liu *et al.*, 2007; Acharya and Ray, 2005; Muneesawang *et al.*, 2014; Marques and Furht, 2002). *Text-based image retrieval* was introduced in the 1970s, and is used by traditional text retrieval techniques for image annotations. It is very limited in terms of scalability due to the lack of automation when image annotations are manually created and labeled by a human. *Content-based image retrieval*, which was announced in the 1980s, applies image processing techniques to automatically extract image features and then retrieve relevant images based on the match of their features.

A content-based image retrieval system usually has two main modules: image feature extracting and indexing and an image feature similarity comparator (Liu *et al.*, 2007; Acharya and Ray, 2005; Muneesawang *et al.*, 2014; Alzu'bi *et al.*, 2015). The main objectives of research on content-based image retrieval are to improve the efficiency of an image feature extractor, to reduce the size of the

indexes, to increase the accuracy of the image feature comparator and to decrease the query response time. Approaching these objectives, our work proposes a model for content-based image retrieval built upon a binary signature and a self-organizing map.

In this model, the features of images are computed by their colors and textures, which are segmented with the techniques of the  $CIEL^*a^*b^*$  color space and *discrete wavelet frames* (DWFs). They are later encoded into bit-strings called binary signatures. The index of image databases is the collection of these binary signatures, which are stored in a graph-based model database called the signature graph. On the signature graph, binary signatures are classified into separated clusters by their similarity. The clustering process is conducted automatically by a self-organizing map model, a type of artificial network which runs unsupervised learning.

By compacting image features that are extracted from images into a binary signature, the size of metadata can be significantly reduced. Applying a self-organizing map model to indexing can enable image querying with a high accuracy rate. Furthermore, this indexing technique also simplifies image similarity comparison and reduces the query response time.

This paper is inspired by our earlier works (Van and Le, 2014c; 2014b). The main contribution of this paper is an alternative model for content-base image retrieval. After describing the details of the model and proving its efficiency, we design a set of different experiments. The empirical evaluations are to measure the metadata size, accuracy and time delay of this model's implementation on three well known image databases: COREL, Wang and MSRDI (Wang, 2016; COREL, 2016; Microsoft, 2016).

The contribution of the paper includes creating a binary signature to describe an image's content, giving a similarity measure between images based on binary signatures, combining the signature graph and SOM based on binary signature, clustering binary signatures and storing them in the signature graph. Then we present algorithms based on a combination between the SOM and signature graph to quickly query similar images.

The remainder of this paper is structured as follows. We discuss several related works in the next section. After that, we describe the technical details of three main components of the model: the creation of a binary signature, the structure of the signature graph and image retrieval with the SOM. In the following section, we present an empirical experiment and discuss the evaluation result. Finally, we finish the paper with our conclusions and discuss the directions for possible future work.

## 2. Related work

A number of works related to queries about image contents have been published recently, such as extracting image objects based on changes in histograms (Liu and Yang, 2013), similar image retrieval based on a comparison of characteristic regions and a similar relationship of feature regions on images (Bartolini *et al.*, 2010), color image retrieval based on the detection of local feature regions by Harris–Laplace (Wang *et al.*, 2010), color image retrieval based on the bit plane and the  $L^*a^*b^*$  color space (Wang *et al.*, 2013), converting the color space and building a hash table in order to query the content of color images (Tang *et al.*, 2013), the similarity of images based on a combination of image colors and texture (Singha and Hemachandran, 2012), using the EMD distance in image retrieval (Bahri and Hamid, 2011), the image indexing and retrieval technique VBA (*variable-bin allocation*) based on signature bit strings and the S-tree (Nascimento *et al.*, 2002), etc.

There are many feature detection methods (Wang *et al.*, 2013), including the angle and edge detector method introduced in 1998 by Harris and Stephens, the SIFT (*scale-invariant feature transform*) introduced in 2003 by Lowe based on the filter of a convolution mask between the image and the DoG (*difference of Gaussians*) to approximate the Laplacian operator of a Gaussian function, SURF (*speeded up robust features*) introduced in 2006 by Bay *et al.*, the point detector method based on the Laplacian operator of a Gaussian function introduced in 2001 by Mikolajczyk and Schmid, etc.

Achary and Ray (2005) proposed a method of approximate texture based on the human visual system. The wavelet transform is applied in analyzing texture and classifying images based on a multi-resolution decomposition of images.

Kumar *et al.* (2009) proposed an automatic segmentation method based on the Wavelet transform in order to create segmentation quickly and easily. The content of this paper shows a method which effectively segments large images and implements easily more than other methods.

Nevertheless, it is time-consuming as well as storage-wasting if we find similar images by matching directly the content of images. In consequence, we need to have methods of description of image contents by meta-data to retrieve similar images.

A number of works on image retrieval based on meta-data have been published recently, including image retrieval using a binary index and a signature tree (Manolopoulos *et al.*, 2003), a similar image retrieval method based on the binary signature (Chappell and Geva, 2013), color image retrieval based on the binary signature (Nascimento and Chitkara, 2002), a color-based image retrieval technique that uses a spiral

bit-string representation for the color content of an image (Abdesselam *et al.*, 2010), image retrieval based on the similarity measure EMD and the S-tree (Le and Van, 2013), color image retrieval using the Hamming distance and the S-tree (Van and Le, 2014a), image retrieval based on the signature graph (Van and Le, 2014c; 2014b), multimedia content-based visual retrieval based on the binary signature of the SIFT (Zhou *et al.*, 2014), image matching using a variable-length signature of the image (Liu *et al.*, 2015), video matching using the binary signature (Li *et al.*, 2005), content-based video using a geometric relation based on the binary signature (Ozkan *et al.*, 2014), etc.

Related works show that the method of image retrieval based on the the binary signature is very effective. Accordingly, we propose a method which creates the binary signature based on the interesting object of the segmented image. The binary signature describes the shape, color and location of the object of interest. Then we collect binary signatures to quickly query for similar images.

### 3. Creation of an image binary signature

**3.1. Binary signature.** The image binary signature is a compact representation of an image-based data object. The retrieval procedure assumes that the image signatures are kept in a storage. To process a query, the storage is scanned and all image signatures are compared with the signature of the query image using a well-defined similarity metric. The images are then returned and ordered by their similarity with the query image.

An  $m$ -bit long binary signature is an  $m$ -bit string which contains  $k$  bits 1 and  $(m - k)$  bits 0. There are three outcomes when comparing two signatures. Firstly, the candidate signature  $s_i$  that matches the signature's query  $s_q$  when for every bit set in  $s_i$  the corresponding bit in  $s_q$  is also set (i.e.,  $s_q \wedge s_i = s_q$ ). Secondly, the candidate signature does not match the query signature (i.e.,  $s_q \wedge s_i \neq s_q$ ). Finally, the signature comparison returns a *false drop* if this indicates a match but the object in fact does not match the search criteria.

Using the binary signature to represent image features could significantly reduce the size of the metadata generated for future querying. For example, for each image of size  $N \times N = 500 \times 500$ , this gives 250,000 pixels. If we use the RGB color space, then each pixel needs three color values: red, green, blue, so we need 750,000 storage values. But if we describe an image by the binary signature quantized on the MPEG7 standard including 25 colors, with each dominant color described by a bit string of length  $m = 10$  for describing a color rate in the image, we need to store a binary string of the length of 250 bits (equivalently, 0.033% in comparison with the storage by the color vector).

In the paper, we set forth a content-based image retrieval method based on the binary signature to create an image index. The binary signature of the image will be represented as an  $n$ -dimensional vector in the binary space  $B^n$ , such that  $B^n = \{(v_1, \dots, v_n) : v_1, \dots, v_n \in \{0, 1\}\}$ .

In this section, we describe the techniques of creation of image binary signatures in our model. The interest regions of the image as image features are segmented and encoded, compacted into binary signatures. The dissimilarity between images is later computed on their corresponding signatures.

**3.2. Image segmentation.** In a content-based image retrieval system, the interest regions of images are extracted and compared to find similar images. There are many different methods to define and segment the image regions of interest. They can be extracted from a specific position on the images, for example, the center of the image (Kim *et al.*, 2003). Regions of interest are also defined by the extracted color and texture of images (Yoo *et al.*, 2002), etc.

In the paper, for any image, there is a need to extract interest regions to create a binary signature to describe the content of an image. This extraction is done on the basis of the image segmentation method by the KMCC (*K-means-with-connectivity-constraint*) algorithm (Kompatsiaris and Strintzis, 2000). The algorithm classifies pixels based on the color, texture and location of the center of the interest region. The result of the method is a segmentation mask, i.e., a gray scale image with different gray levels corresponding to interest regions.

For any pixel  $p(x, y)$ , the color vector  $I(p)$  and the texture vector  $T(p)$  are calculated. The former consists of three values of coordinates in the color space  $CIEL^*a^*b^*$  because it is perceptually uniform, i.e., the numerical distance in this color space is approximately proportional to the perceived color difference (Mezaris *et al.*, 2004). Thus, the color vector of pixel  $p$  is  $I(p) = (I_L(p), I_a(p), I_b(p))$ .

The texture vector  $T(p)$  is calculated by the DWF (*discrete wavelet frames*) transform (Mezaris *et al.*, 2004). In Fig. 1, we describe the model of the DWF transform. This method is similar to the DWT (*discrete wavelet transform*) in decomposing the intensity of the image into sub-samples. The low-pass filter is used with Haar wavelet methods with  $H(z) = \frac{1}{2}(1 + z^{-1})$ , where  $H(z)|_{z=1} = 1$ . The high-pass filter  $G(z)$  is defined by the low-pass filter as  $G(z) = zH(-z^{-1})$ . The bank filter  $H_V(z)$ ,  $G_i(z)$ ,  $i = 1, \dots, V$ , is created by  $H(z)$ ,  $G(z)$ , with  $H_{k+1}(z) = H(z^{2^k})H_k(z)$ ,  $G_{k+1}(z) = G(z^{2^k})H_k(z)$ , with  $k = 0, \dots, V - 1$ ,  $H_0(z) = 1$ .

The standard deviation reflects entropy around the

expectation and this entropy describes the texture of discrete signals. In consequence, the standard deviation of all detail components of the DWF transform is used as the feature texture. For this reason, the texture vector corresponding to pixel  $p$  is  $T(p) = [\sigma_1(p), \sigma_2(p), \dots, \sigma_{9 \times V}(p)]$ , which is calculated on the neighboring square. For the sake of clarity, we present a specific example in Fig. 2.

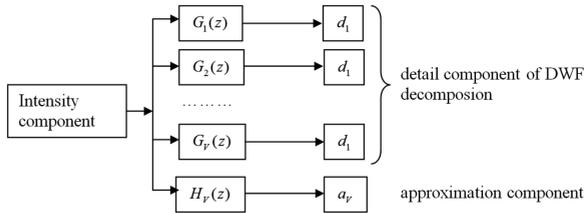


Fig. 1. Model of the DWF transform.

5	4	2
2	6	3
3	4	2

Fig. 2. Neighbor rectangle of pixel  $p$ . The texture vector  $T(p)$  with  $V = 2$  is  $T(p) = (2.50, 1.25, 2.00, 1.00, 1.00, 0.50, 1.00, 0.50, 3.00, 1.50, 1.50, 0.75, 1.50, 0.75, 2.00, 1.00, 1.00, 0.50)$ .

After extracting the texture and color of the image, we implement the process of clustering all pixels on the image by the  $k$ -means method. In the first step, we choose the center cluster relying on the contrast  $C$  of the image. To quickly execute, the image  $I$  is divided into non-overlapping blocks called superpixels (Fig. 3). Therefore, the texture vector  $T^b(b_l)$  and the color vector  $I^b(b_l)$  of the block  $b_l$  are the average values of texture vectors and color vectors of all pixels in the block.

**Definition 1.** Given the two arbitrary blocks  $b_l, b_n$ , the contrast of an image is defined as

$$C = \max\{d = \alpha ||I^b(b_l) - I^b(b_n)|| + \beta ||T^b(b_l) - T^b(b_n)||\}. \quad (1)$$

In the experiment, we use  $\alpha = \beta = 0.5$ . The background and the foreground of the image correspond to the blocks which have low and high energies, respectively.

In the next step, we find the set of complement centers  $O$  (i.e., we search the nearest blocks with the foreground relying on the measure  $d$ ). In the experiment, we find the centers which have  $d > \mu C$  (with  $\mu = 0.4$ ) to cluster all pixels in the image. The image segmentation algorithm is given as Algorithm 1.



Fig. 3. Sample image is divided into a set of blocks.

**Algorithm 1.** Image segmentation.

**Require:** Color image  $I$

- 1: *Step 1:* Extract the texture vector  $T(p)$  and the intensity vector  $I(p)$  for each pixel on the image.
- 2: *Step 2:* Compute the center of blocks by calculating the mean of texture vectors  $T(p)$  and color vectors  $I(p)$  of all pixels in each block.
- 3: *Step 3:* Calculate the contrast  $C$  to form a background and foreground of the image.
- 4: *Step 4:* Find the set of complement centers  $O$  based on the contrast  $C$ .
- 5: *Step 5:* Cluster all the pixels in the image  $I$  based on the set of centers  $O$ .
- 6: *Step 6:* Create a mask  $M$  with clustered pixels.
- 7: *Step 7:* Remove the regions which have a small area based on the mask  $M$ .
- 8: *Step 8:* Return the mask  $M$ .

The result of the segmentation process is a mask (i.e., gray image) to describe the image objects of interest. On the basis of the mask, we compute the connected regions and remove those which have a small area (the experiment removes those which have the area smaller than  $\theta = 5\%$  of the image). In Fig. 4, we show some results of image segmentation. The computing of the region areas is done by a 4-neighboring algorithm presented as Algorithm 2.

**3.3. Generating the binary signature.** After partitioning the image regions of interest, we generate bit-strings describing these extracted images. The binary signature of a region of interest  $O$  which is segmented from  $n \times k$  cell image is introduced in Definition 2.

**Definition 2.** The binary signature of the region of interest  $O$  of image  $I$  is a bit-string  $Sig_I(O) = b_1 b_2 \dots b_N$ , where  $b_i = 1$  if the  $i$ -th cell of image  $I$  has an overlap with the region of interest  $O$ ; otherwise,  $b_i = 0$  with  $N = n \times k$  being the number of image cells.

One image may have more than one regions of interest. Hence, the binary signature of an image should be a combination of the binary signatures of its regions of interest. We define this combination in Definition 3.

**Definition 3.** Let  $O_1, O_2, \dots, O_m$  be different regions of interest of image  $I$  and assume that they have

**Algorithm 2.** Compute the area of a connected region.

**Require:** Mask  $M$  and a location  $(r, c)$

```

1: Step 1: Initialize  $S = 0$ ;  $Stack = \emptyset$ ;
2: Step 2:  $Push(Stack, r, c)$ ;
3: Step 3:
4: while  $Stack \neq \emptyset$  do
5:    $(r, c) = Pop(Stack)$ ;
6:    $S = S + 1$ ;
7:   if  $(r > 1)$  and  $(M(r, c) = M(r - 1, c))$  then
8:      $Push(Stack, r - 1, c)$ ;
9:   end if
10:  if  $(r < rows)$  and  $(M(r, c) = M(r + 1, c))$  then
11:     $Push(Stack, r + 1, c)$ ;
12:  end if
13:  if  $(c > 1)$  and  $M(r, c) == M(r, c - 1)$  then
14:     $Push(Stack, r, c - 1)$ ;
15:  end if
16:  if  $(c < columns)$  and  $M(r, c) = M(r, c + 1)$  then
17:     $Push(Stack, r, c + 1)$ ;
18:  end if
19: end while
20: return  $S$ ;

```

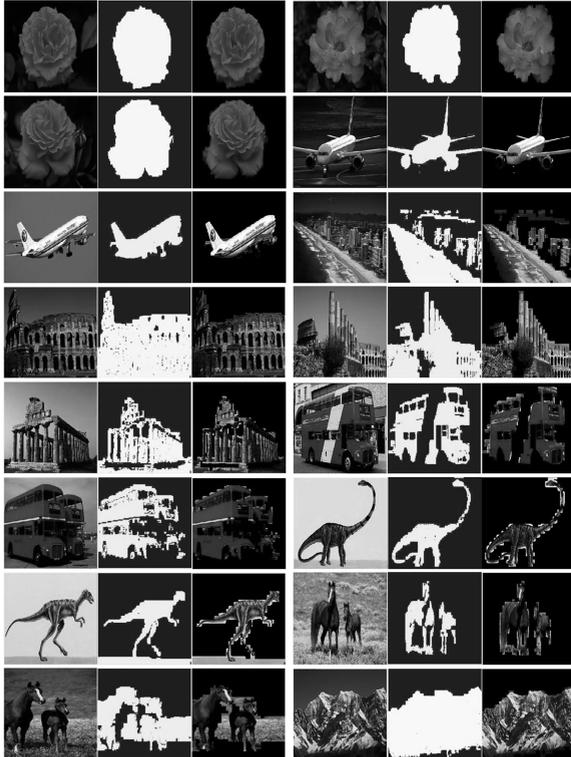


Fig. 4. Some of segmented images.

binary signatures  $Sig_I(O_1), Sig_I(O_2), \dots, Sig_I(O_m)$ , respectively. Then the binary signature of all regions of

interest of image  $I$  is

$$Sig_O^I = \bigcup_{i=1}^m Sig_I(O_i).$$

For any image quantized in the form of  $M$  colors (in the experiment reported in the paper, we use the MPEG7 standard), the binary signature describes the colors of the image defined as follows.

**Definition 4.** Let  $c_1, c_2, \dots, c_m$  be the indexes of dominance colors of regions of interest  $O_1, O_2, \dots, O_m$  with  $c_i \in \{1, 2, \dots, M\}$ . Then, the *binary signature* describes colors of image  $I$  based on regions of interest  $O_1, O_2, \dots, O_m$  as a bit-string  $Sig_C^I = b_1 b_2 \dots b_M$ , where  $b_i = 1$  if  $i \in \{c_1, c_2, \dots, c_m\}$ , otherwise  $b_i = 0$ .

On the basis of the binary signature of regions of interest and the ones of dominance colors, the binary signature of the image is introduced in Definition 5. In Fig. 5, we show an example of an image binary signature.

**Definition 5.** Given an image  $I$ , let  $Sig_O^I = b_1^O \dots b_N^O$  and  $Sig_C^I = b_1^C \dots b_M^C$  be the binary signature of regions of interest and the ones of dominance colors of image  $I$ . Then, the binary signature of image  $I$  is calculated by the following formula:

$$Sig(I) = Sig_O^I \oplus Sig_C^I = b_1^O \dots b_N^O b_1^C \dots b_M^C. \quad (2)$$

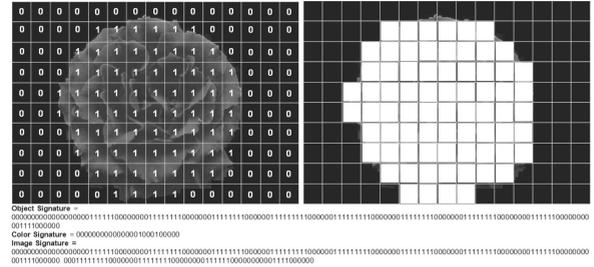


Fig. 5. Example of a binary signature of an image.

**3.4. Dissimilarity measure.** The dissimilarity measure between two images is a necessary component when performing the process of similar image retrieval. However, each image is described in the form of a binary signature, so the problem needs to build a dissimilarity measure between binary signatures so as to assess the similarity of images. Then we define the dissimilarity measure as follows.

**Definition 6.** Let  $Sig(I) = Sig_O^I \oplus Sig_C^I$  and  $Sig(J) = Sig_O^J \oplus Sig_C^J$  be binary signatures of image  $I$  and  $J$ . Then, the dissimilarity measure based on regions of interest and

colors  $\mu_O(sig_O^I, sig_O^J)$  and  $\mu_C(sig_C^I, sig_C^J)$  is defined as follows:

$$\mu_O(sig_O^I, sig_O^J) = \frac{1}{N} \sum_{i=1}^N (sig_O^I[i] XOR sig_O^J[i]), \quad (3)$$

$$\mu_C(sig_C^I, sig_C^J) = \frac{1}{M} \sum_{i=1}^N (sig_C^I[i] XOR sig_C^J[i]). \quad (4)$$

**Theorem 1.** *The function of dissimilarity measure  $\mu_\alpha$  is a metric because of the following properties:*

(i) *non-negativity:*

$$\mu_\alpha(sig_\alpha^I, sig_\alpha^J) \geq 0$$

and

$$\mu_\alpha = 0 \Leftrightarrow sig_\alpha^I = sig_\alpha^J,$$

(ii) *symmetry:*

$$\mu_\alpha(sig_\alpha^I, sig_\alpha^J) = \mu_\alpha(sig_\alpha^J, sig_\alpha^I),$$

(iii) *triangle inequality:*

$$\mu_\alpha(sig_\alpha^I, sig_\alpha^J) + \mu_\alpha(sig_\alpha^J, sig_\alpha^K) \geq \mu_\alpha(sig_\alpha^I, sig_\alpha^K).$$

*Proof.*

(i) Let  $sig_\alpha^I$  and  $sig_\alpha^J$  be two binary signatures of images  $I$  and  $J$ , with  $sig_\alpha^I[i], sig_\alpha^J[i] \in \{0, 1\}, i = 1, \dots, N$ . Then  $sig_\alpha^I[i] XOR sig_\alpha^J[i] \geq 0$ . We see that,  $\mu(sig_\alpha^I, sig_\alpha^J) \geq 0$ .

Hence, the function of dissimilarity measure  $\mu_\alpha(sig_\alpha^I, sig_\alpha^J)$  is non-negative.

Observe that

$$\begin{aligned} \mu_\alpha(sig_\alpha^I, sig_\alpha^J) &= 0 \\ \Leftrightarrow \frac{1}{N} \sum_{i=1}^N (sig_\alpha^I[i] XOR sig_\alpha^J[i]) &= 0 \\ \Leftrightarrow sig_\alpha^I[i] = sig_\alpha^J[i], i = 1, \dots, N & \\ \Leftrightarrow \frac{1}{N} \sum_{i=1}^N (sig_\alpha^I[i] XOR sig_\alpha^J[i]) &= 0. \end{aligned}$$

Consequently, the function of dissimilarity measure  $\mu_\alpha(sig_\alpha^I, sig_\alpha^J)$  is unique.

(ii) Since the XOR operator is commutative, we get

$$\begin{aligned} \mu_\alpha(sig_\alpha^I, sig_\alpha^J) &= \frac{1}{N} \sum_{i=1}^N (sig_\alpha^I[i] XOR sig_\alpha^J[i]) \\ &= \frac{1}{N} \sum_{i=1}^N (sig_\alpha^J[i] XOR sig_\alpha^I[i]) \\ &= \mu_\alpha(sig_\alpha^J, sig_\alpha^I). \end{aligned}$$

As a result,  $\mu(sig_\alpha^I, sig_\alpha^J) = \mu(sig_\alpha^J, sig_\alpha^I)$ .

Hence, the function of dissimilarity measure  $\mu_\alpha(sig_\alpha^I, sig_\alpha^J)$  is symmetric.

(iii) Let  $sig_\alpha^I, sig_\alpha^J, sig_\alpha^K$  be three binary signatures of images  $I, J$  and  $K$ . We have

$$\begin{aligned} \mu_\alpha(sig_\alpha^I, sig_\alpha^J) + \mu_\alpha(sig_\alpha^J, sig_\alpha^K) \\ = \frac{1}{N} \sum_{i=1}^N (sig_\alpha^I[i] XOR sig_\alpha^J[i]) \\ + (sig_\alpha^J[i] XOR sig_\alpha^K[i]). \end{aligned}$$

We can use a truth table to infer

$$\begin{aligned} (sig^I[i] XOR sig^J[i]) + (sig^J[i] XOR sig^K[i]) \\ \geq (sig^I[i] XOR sig^K[i]). \end{aligned}$$

Then

$$\mu_\alpha(sig_\alpha^I, sig_\alpha^J) + \mu_\alpha(sig_\alpha^J, sig_\alpha^K) \geq \mu_\alpha(sig_\alpha^I, sig_\alpha^K).$$

Therefore, the function of dissimilarity measure  $\mu_\alpha(sig_\alpha^I, sig_\alpha^J)$  satisfies the condition of the triangle inequality. ■

**Definition 7.** (*Dissimilarity measure*) Let  $Sig(I) = Sig_O^I \oplus Sig_C^I$  and  $Sig(J) = Sig_O^J \oplus Sig_C^J$  be binary signatures of image  $I$  and  $J$ . Then the dissimilarity measure between two images  $I$  and  $J$  is defined as follows:

$$\phi(I, J) = \alpha \mu_O(sig_O^I, sig_O^J) + \beta \mu_C(sig_C^I, sig_C^J), \quad (5)$$

where  $\alpha, \beta \in [0, 1]$  are adjustment coefficients,  $\alpha + \beta = 1$ ,  $N$  is the number of image cells and  $M$  is the number of colors to quantize the image. (In the experiment reported the paper, we use  $N = 40$  and  $M = 25$ .)

**Theorem 2.** *The dissimilarity measure  $\phi(I, J)$  in Definition 7 is a metric.*

*Proof.* According to Theorem 1,  $\mu_O(sig_O^I, sig_O^J)$  and  $\mu_C(sig_C^I, sig_C^J)$  are metrics. Thus we deduce that  $\phi(I, J)$  is a metric. ■

Throughout the paper, we use the dissimilarity measure  $\phi$  to assess a similarity level between two images. Therefore, the dissimilarity measure  $\phi$  is a similarity measure. On the basis of the similarity measure between two images, we perform the query process of similar images. For any query image, we find out relevant images based on the similarity measure. Retrieval of similar images is defined as follows.

**Definition 8.** Given an image database  $\mathfrak{S}$ , let  $I$  be a query image. Let  $\mathfrak{R}_I = \{J_i^I | (J_i^I \in \mathfrak{S}) \wedge (\phi(I, J_i^I) \leq \phi(I, J_j^I) \Leftrightarrow J_i^I \succ J_j^I) \wedge (i \neq j) \wedge (i, j = 1, \dots, n)\}$  be

an ordered set based on the measure  $\phi$ . A set of similar images  $Q \subset \mathfrak{S}$  including  $k$  similar images is defined as follows:

$$Q = \{J_i \in \mathfrak{S} | \phi(I, J_i) \leq \theta, \forall J \in \mathfrak{S}, i = 1, \dots, |Q|\}, \quad (6)$$

where  $\theta$  is the threshold of  $\phi(I, J_i)$

**Theorem 3.** *If  $I$  is the query image, then the set of similar images  $Q \subset \mathfrak{S}$  is an ordered set on the relation  $\succ$ .*

*Proof.*

(i) *Symmetry.* If  $I$  is the query image and  $J \in Q$  is an arbitrary image, then  $\phi(I, J) = \phi(I, J)$ , i.e., it satisfies the condition  $\phi(I, J) \leq \phi(I, J)$ . Hence  $J \succ J$ , i.e.,  $Q$  has symmetry on  $\succ$ .

(ii) *Antisymmetry.* Given  $J_i, J_j \in Q$  with  $i \neq j$ , suppose that  $J_i \succ J_j$  (i.e.,  $\phi(I, J_i) \leq \phi(I, J_j)$ ) and  $J_i \neq J_j$  as a result of  $\phi(I, J_i) < \phi(I, J_j)$ . In addition, Theorem 2 shows that  $\phi$  is a metric. Therefore, we have  $\phi(I, J_j) \not\leq \phi(I, J_i)$ . In consequence, if  $J_i \succ J_j$ , then  $J_j \not\succeq J_i$ . Hence,  $Q$  has antisymmetry on  $\succ$ .

(iii) *Transitivity.* Let  $J_1, J_2, J_3 \in Q$  be three images corresponding to the image query  $I$ . Suppose that  $J_1 \succ J_2$  and  $J_2 \succ J_3$ , i.e.,  $\phi(I, J_1) \leq \phi(I, J_2)$  and  $\phi(I, J_2) \leq \phi(I, J_3)$ . According to Theorem 2,  $\phi$  is a metric, so  $\phi(I, J_1) \leq \phi(I, J_3)$ . Deduce that, if  $J_1 \succ J_2$  and  $J_2 \succ J_3$ , then  $J_1 \succ J_3$ , i.e.,  $Q$  has transitivity on  $\succ$ .

From (i)–(iii) we conclude that the set of similar images  $Q \subset \mathfrak{S}$  is an ordered set on the relation  $\succ$ . ■

#### 4. Signature graph

Having defined the binary signature and the similarity measure between images, the problem to be solved is to perform queries quickly and reduce the storage requirement of the query system. In consequence, we should build a data structure to store the binary signature as well as to describe the relationship between the images. In the paper, we build a graph structure in order to describe the similarity relationship based on the binary signature (Definition 5) and the similarity measure (Definition 7). This graph structure is called an SG (*signature graph*), for any vertex in the graph including the pair of identification  $oid_I$  and the signature  $sig_I$  corresponding to the image  $I$ . The weight between two vertexes is the similarity measure  $\phi$ . This signature graph is developed from our own paper (Van and Le, 2014b).

The signature graph describes the relationship among the binary signatures, which indicates the relationship among image contents. The SG is a graph which has the weight, including a set of vertexes  $V_{SG}$  and a set of edges  $E_{SG}$  which are defined as follows.

**Definition 9.** The *signature graph*  $SG = (V, E)$  is a graph which describes the relationship among the images, has a set of vertexes  $V = \{\langle oid_I, Sig(R^I) \rangle | I \in \mathfrak{S}\}$  and a set of edges,  $E = \{\langle I, J \rangle | \phi(I, J) = \phi(R^I, R^J) \leq \theta, \forall I, J \in \mathfrak{S}\}$ . With the weight of each edge  $\langle I, J \rangle$ ,  $\phi(I, J)$  is the similarity measure,  $\theta$  is a threshold value and  $\mathfrak{S}$  is an image database.

For any vertex  $v \in V$  in  $SG$ , we determine  $k$  nearest elements based on the similarity measure. However, if the number of images is large, it would be difficult to determine the set of similar images corresponding to the query image. Therefore, we introduce the notion of an S- $k$ Graph so that each vertex includes  $k$  nearest images, and call it *k-neighboring*.

For any *k-neighboring*, we build a cluster including similar images. This cluster is defined as follows.

**Definition 10.** A *cluster*  $V_i$  has center  $I_i$  with  $k_i\theta$  being a radius, and it is defined as  $V_i = V_i(I_i) = \{J | \phi(I_i, J) \leq k_i\theta, J \in \mathfrak{S}, i = 1, \dots, n\}$ ,  $k_i \in N^*$ .

On the basis of clusters, we define the data structure S- $k$ Graph including vertexes as clusters and the weight between two vertexes as the similarity measure  $\phi$ . The data structure S- $k$ Graph (Fig. 6) is defined as follows.

**Definition 11.** Let  $\Omega = \{V_i | i = 1, \dots, n\}$  be a set of clusters, with  $V_i \cap V_j = \emptyset, i \neq j$ . The S- $k$ Graph =  $(V_{SG}, E_{SG})$  is a graph which has the weight, including a set of vertexes  $V_{SG}$  and a set of edges  $E_{SG}$  defined as follows:

$$V_{SG} = \Omega \\ = \{V_i | \exists! I_{i_0} \in V_i, \forall I \in V_i, \phi(I_{i_0}, I) \leq k_{i_0}\theta\}, \quad (7)$$

$$E_{SG} = \{\langle V_i, V_j \rangle | V_i, V_j \in V_{SG}, \\ d(V_i, V_j) = \phi(I_{i_0}, J_{j_0})\}, \quad (8)$$

with  $d(V_i, V_j)$  being the weight between two clusters and  $\forall I \in V_i, \phi(I_{i_0}, I) \leq k_{i_0}\theta$ .

With each image required to be classified in clusters through the data structure S- $k$ Graph, we need to have the rule of distribution in clusters of the S- $k$ Graph. These rules are defined as follows.

**Definition 12.** (*Rules of distribution of images*) Let  $\Omega = \{V_i | i = 1, \dots, n\}$  be a set of clusters, with  $V_i \cap V_j = \emptyset, i \neq j$ . Let  $I_0$  be an image to be distributed in a set of clusters  $\Omega$ . Let  $I_m$  be a center of cluster  $V_m$ , so that  $(\phi(I_0, I_m) - k_m\theta) = \min\{\phi(I_0, I_i) - k_i\theta, i = 1, \dots, n\}$ , with  $I_i$  being a center of cluster  $V_i$ . There are three cases as follows:

1. If  $\phi(I_0, I_m) \leq k_m\theta$ , then distribute the image  $I_0$  in cluster  $V_m$ .

2. If  $\phi(I_0, I_m) > k_m\theta$ , then set

$$k_0 = \frac{1}{\theta}(\phi(I_0, I_m) - k_m\theta)$$

at that time.

2.1 If  $k_0 > 0$ , then create cluster  $V_0$  with center  $I_0$  and radius  $k_0\theta$ ; at that time  $\Omega = \Omega \cup \{V_0\}$ .

2.2 Otherwise (i.e.,  $k_0 = 0$ ), distribute the image  $I_0$  in cluster  $V_m$  and set  $\phi(I_0, I_m) = k_m\theta$ .

For any image, there is a need to have a cluster in the S-kGraph, so that images are classified. Moreover, to avoid invalid data in clusters, the images are distributed in a unique cluster. Theorems 4 and 5 show the unique distribution.

**Theorem 4.** Given the S-kGraph =  $(V_{SG}, E_{SG})$ , let  $\langle V_i, V_j \rangle \in E_{SG}$  and  $I_{i_0}, J_{j_0}$  be a center of  $V_i, V_j$ . Then,  $d(V_i, V_j) = \phi(I_{i_0}, J_{j_0}) > (k_{i_0} + k_{j_0})\theta$ , with  $\forall I \in V_i, \phi(I_{i_0}, I) \leq k_{i_0}\theta$  and  $\forall J \in V_j, \phi(J_{j_0}, J) \leq k_{j_0}\theta$ .

*Proof.* We have  $\forall I \in V_i, \phi(I_{i_0}, I) \leq k_{i_0}\theta$  and  $\forall J \in V_j, \phi(J_{j_0}, J) \leq k_{j_0}\theta$ . From  $\forall I' \in \text{Boundary}(V_i), \forall J' \in \text{Boundary}(V_j)$ , it follows that  $\phi(I_{i_0}, I') = k_{i_0}\theta$  and  $\phi(J_{j_0}, J') = k_{j_0}\theta$ . Moreover, since  $V_{SG} = \Omega$  is a set of unconnected cluster, we get  $V_i \cap V_j = \emptyset$  and  $\phi(I', J') > 0$ .

Deduce that if  $\forall I' \in \text{Boundary}(V_i), \forall J' \in \text{Boundary}(V_j)$ , then  $\phi(I_{i_0}, I') + \phi(I', J') + \phi(J_{j_0}, J') > (k_{i_0} + k_{j_0})\theta$ . Otherwise, since  $\phi$  is a metric,  $\phi(I_{i_0}, I') + \phi(I', J') + \phi(J_{j_0}, J') \geq \phi(I_{i_0}, J_{j_0})$  and  $\exists I'_0 \in \text{Boundary}(V_i), \exists J'_0 \in \text{Boundary}(V_j)$ , so that  $\phi(I_{i_0}, I'_0) + \phi(I'_0, J'_0) + \phi(J_{j_0}, J'_0) = \phi(I_{i_0}, J_{j_0})$ .

Therefore,  $\phi(I_{i_0}, J_{j_0}) > (k_{i_0} + k_{j_0})\theta$ . ■

**Theorem 5.** If each image  $I$  is distributed in a set of clusters  $\Omega = \{V_i | i = 1, \dots, n\}$ , then it belongs to a unique cluster.

*Proof.* Let  $I$  be an any image. Suppose  $\exists V_i, V_j$  as two clusters, so  $V_i \neq V_j$  and  $(I \in V_i) \wedge (I \in V_j)$ . Setting  $I_i, I_j$  in turn as two cluster centers  $V_i, V_j$ , we have  $\phi(I_i, I) \leq k_i\theta$  and  $\phi(I_j, I) \leq k_j\theta$ . Thus,  $\phi(I_i, I) + \phi(I_j, I) \leq (k_i + k_j)\theta$ . Furthermore, as  $\phi$  is a metric, we have  $\phi(I_i, I) + \phi(I_j, I) \geq \phi(I_i, I_j)$ . Otherwise,  $I_i, I_j$  are in turn two cluster centers  $V_i, V_j$  so that  $\phi(I_i, I_j) > (k_i + k_j)\theta$ . Hence  $\phi(I_i, I) + \phi(I_j, I) \geq \phi(I_i, I_j) > (k_i + k_j)\theta$  and  $\phi(I_i, I) + \phi(I_j, I) \leq (k_i + k_j)\theta$ . For this reason, the supposition is illogical, i.e., each image  $I$  is only distributed in a unique cluster. ■

In order to avoid invaliding data, the rules of distribution (Definition 12) need to ensure that the image is classified in a unique cluster. Theorems 6–8 show this problem.

**Theorem 6.** If  $\phi(I, I_m) - k_m\theta \leq 0$ , then this happens only at a unique  $I_m$ .

*Proof.* Suppose that  $\exists I_0$  is a center of cluster  $C_0 \in \Omega$  so that  $\phi(I, I_0) - k_0\theta \leq 0 \Leftrightarrow \phi(I, I_0) \leq k_0\theta$ , i.e.,  $I$  belongs to cluster  $C_0$ . Otherwise, according to the supposition,  $\phi(I, I_m) - k_m\theta \leq 0$ , i.e.,  $I$  belongs to cluster  $C_m \neq C_0$ . This means that  $I$  belongs to two different clusters and, by Theorem 5, for any image  $I$  belongs only to a unique cluster. Thus, the supposition is illogical. That means that if the value is  $\phi(I, I_m) - k_m\theta \leq 0$ , it only occurs at one unique  $I_m$ . ■

**Theorem 7.** If  $\Omega = \{V_i | i = 1, \dots, n\}$  is a set of clusters and  $I$  is an image, then there is a cluster  $V_{i_0} \in \Omega$  such that  $I \in V_{i_0}$ .

*Proof.* According to Definition 12, for any image  $I$  there also exists a cluster  $V_{i_0} \in \Omega$  so that  $I \in V_{i_0}$ . ■

**Theorem 8.** For any image,  $I$  is distributed in a unique cluster  $C_{i_0} \in \Omega$ .

*Proof.* According to Definition 12, for any image  $I$  there also exists a cluster  $V_{i_0} \in \Omega$  so that  $I \in V_{i_0}$ . According to Theorem 5, any image  $I$  is only distributed in a unique cluster. This means that any image  $I$  is distributed in a unique cluster  $C_{i_0} \in \Omega$ . ■

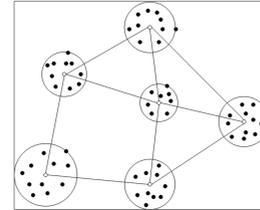


Fig. 6. Model of the S-kGraph.

On the basis of the similarity measure  $\phi$ , the S-kGraph is shown in Definition 11 and the rules of distribution of the image are given in Definition 12, and the paper proposes an algorithm to create the data structure S-kGraph. With the input image database  $\mathfrak{S}$  and the threshold  $k\theta$ , we need to return the S-kGraph. (In the experiment reported in this paper, we initialize  $k = 1, \theta = 0.1$ .)

Firstly, we initialize the set of vertexes  $V_{SG} = \emptyset$  and the set of edges  $E_{SG} = \emptyset$ , and after that we create the first cluster. With each image  $I$  we evaluate the distance  $\phi$  from the cluster center and find out the nearest cluster according to  $(\phi(I, I_m^0) - k_m\theta) = \min\{\phi(I, I_i^0) - k_i\theta, i = 1, \dots, n\}$ . If the condition  $\phi(I, I_m^0) \leq k_m\theta$  is satisfied, the image  $I$  is distributed in cluster  $V_m$ . Otherwise, we consider the rules of distribution as shown in Definition 12 to classify the image  $I$  into the appropriate cluster.

The cost of the construction of the S-kGraph is  $O(m \times n)$ , where  $n$  is the number of signatures and  $m$  is the number of clusters. In the experiment, if we build

the S- $k$ Graph on the COREL database, then  $n = 1,000$ ,  $m = 144$ ; on the Wang database, we have  $n = 10,800$  and  $m = 875$ ; on the MSRDI database, we have  $n = 16,710$ ,  $m = 499$ ; on the imageCLEF database, we have  $n = 20,000$ ,  $m = 115$ . Consequently, the average value of  $m$  is less than 10% in comparison with  $n$ . This algorithm is as given by Algorithm 3.

---

**Algorithm 3.** Create the S- $k$ Graph.

**Require:** Image database  $\mathfrak{S}$  and threshold  $k\theta$

```

1:  $V_{SG} = \emptyset; E_{SG} = \emptyset; k_I = 1; n = 1;$ 
2: for  $\forall I \in \mathfrak{S}$  do
3:   if  $V_{SG} = \emptyset$  then
4:      $I_n^0 = I; r = k_I\theta;$ 
5:     Initialize cluster  $V_n = \langle I_n^0, r, \phi = 0 \rangle;$ 
6:      $V_{SG} = V_{SG} \cup V_n;$ 
7:   else
8:      $(\phi(I, I_m^0) - k_m\theta) = \min\{\phi(I, I_i^0) - k_i\theta, i = 1, \dots, n\};$ 
9:     if  $\phi(I, I_m^0) \leq k_m\theta$  then
10:       $V_m = V_m \cup \langle I, k_m\theta, \phi(I, I_m^0) \rangle;$ 
11:    else
12:       $k_I = [(\phi(I, I_0^0) - k_m\theta)/\theta];$ 
13:      if  $k_I > 0$  then
14:         $I_{n+1}^0 = I; r = k_I\theta;$ 
15:        Initialize cluster  $V_{n+1} = \langle I_{n+1}^0, r, \phi = 0 \rangle;$ 
16:         $V_{SG} = V_{SG} \cup V_{n+1};$ 
17:         $E_{SG} = E_{SG} \cup \{(V_{n+1}, V_i) | \phi(I_{n+1}^0, I_i^0) \leq k\theta, i = 1, \dots, n\};$ 
18:         $n = n + 1;$ 
19:      else
20:         $\phi(I, I_m^0) = k_m\theta;$ 
21:         $V_m = V_m \cup \langle I, k_m\theta, \phi(I, I_m^0) \rangle;$ 
22:      end if
23:    end if
24:  end if
25: end for
26: return S- $k$ Graph =  $(V_{SG}, E_{SG});$ 

```

---

## 5. Image retrieval on a self-organizing map

**5.1. Sig-SOM structure.** The classification is performed with the input data as the image signature  $Sig(I) = b_1b_2 \dots b_N$ . After classifying, this signature belongs to a cluster  $V_j = \langle I_j^0, k_j\theta \rangle$ , with  $I_j^0$  as the center of cluster  $V_j$  and  $k_j\theta$  the radius of the cluster.

In order for the classification to perform quickly and efficiently, we apply the Sig-SOM network model (Fig. 7) to classify binary signatures. The Sig-SOM structure is defined as follows.

**Definition 13.** (*Sig-SOM structure*) The Sig-SOM network has two layers, including the input layer and the output layer. The input layer has  $N$  items corresponding

to the binary signature  $Sig(I) = b_1b_2 \dots b_N$ , with  $b_i \in \{0, 1\}$ . The output layer has  $n$  clusters corresponding to the signature graph S- $k$ Graph =  $(V_{SG}, E_{SG})$ . Each  $j$ -th cluster in the output layer is fully connected to  $N$  items in the input layer. The edge from the  $i$ -th item in the input layer to the  $j$ -th cluster has a weight  $w_{ij} \in \{0, 1\}$ .

**Definition 14.** (*Winner cluster*) Let  $Sig(I) = b_1b_2 \dots b_N$  be a binary signature of image  $I$ . Let  $SIG = \{Sig(I_1^0), Sig(I_2^0), \dots, Sig(I_n^0)\}$  be a set of binary signatures corresponding to the centers of clusters in the output layer. The winner cluster  $V_m$  corresponding to the input data  $Sig(I) = b_1b_2 \dots b_N$  satisfies the condition as  $\delta(I, I_m^0) = \max\{\delta(I, I_i^0) | \delta(I, I_i^0) = \alpha \times d(sig_O^I, sig_O^{I_i}) + \beta \times d(sig_C^I, sig_C^{I_i}), i = 1, \dots, n\}$ , with  $\alpha, \beta \in [0, 1], \alpha + \beta = 1$ :

$$d(sig_O^I, sig_O^{I_i}) = \frac{1}{N} \sum_{i=1}^N NOT(Sig_O^I[i] XOR Sig_O^{I_i}[i]), \quad (9)$$

$$d(sig_C^I, sig_C^{I_i}) = \frac{1}{M} \sum_{i=1}^M NOT(Sig_C^I[i] XOR Sig_C^{I_i}[i]). \quad (10)$$

At that time, the winner cluster  $V_m$  is connected directly to the winner vector  $W_m$ .

**Theorem 9.** If  $V_m$  is a winner cluster corresponding to input data  $Sig(I) = b_1b_2 \dots b_N$ , then the similarity measure  $\phi(I, I_m^0)$  is minimum, i.e.,  $\phi(I, I_m^0) = \min\{\phi(I, I_i^0) | i = 1, \dots, |V_{SG}|\}$ , where  $I_i^0$  is the center of cluster  $V_i \in V_{SG}$ .

*Proof.* According to Eqns. (3) and (9), we have

$$\begin{aligned}
& \mu_O(sig_O^I, sig_O^J) + d(sig_O^I, sig_O^J) \\
&= \frac{1}{N} \sum_{i=1}^N (sig_O^I[i] XOR sig_O^J[i]) \\
&+ \frac{1}{N} \sum_{i=1}^N NOT(sig_O^I[i] XOR sig_O^J[i]) \\
&= \frac{1}{N} \sum_{i=1}^N [(sig_O^I[i] XOR sig_O^J[i]) \\
&+ NOT(sig_O^I[i] XOR sig_O^J[i])] \\
&= 1.
\end{aligned}$$

Deduce that  $\mu_O(sig_O^I, sig_O^J) = 1 - d(sig_O^I, sig_O^J)$ .

With a similar method, we have  $\mu_C(sig_C^I, sig_C^J) = 1 - d(sig_C^I, sig_C^J)$ . Thus,  $\phi(I, I_m^0) = (\alpha + \beta) - (\alpha \times \mu_O(sig_O^I, sig_O^{I_m^0}) + \beta \times \mu_C(sig_C^I, sig_C^{I_m^0})) = 1 - \delta(I, I_m^0)$ .

Accordingly, if  $\delta(I, I_m^0) = \max\{\delta(I, I_i^0), i = 1, \dots, n\}$ , then  $\phi(I, I_m^0) = \min\{\phi(I, I_i^0) | i = 1, \dots, |V_{SG}|\}$ . ■

However, the output data of this network age the cluster  $V_i$  combined with the center cluster  $I_j^0$  and the radius of cluster  $k_j\theta$ . On the other hand, for any input image signature to be classified in a cluster but not to satisfy the condition for the radius of the cluster, this network training process allows increasing the number of output clusters. Accordingly, we propose a Sig-SOM network model in order to classify the input data including image signatures  $Sig(I) = b_1b_2 \dots b_N$ . It gives the output data as clusters  $V_j = \langle I_j^0, k_j\theta \rangle$ ; at the same time the Sig-SOM network allows creating more new output classifications in order to collect the cluster of image signatures which fail to satisfy the condition on the current clusters. There must be a full connection between the input layer and the output layer of the Sig-SOM network via connected weight vectors  $W_1, W_2, \dots, W_n$ , with  $W_j = (w_{1j}, w_{2j}, \dots, w_{Nj})$ ,  $w_{ij} \in \{0, 1\}$ .

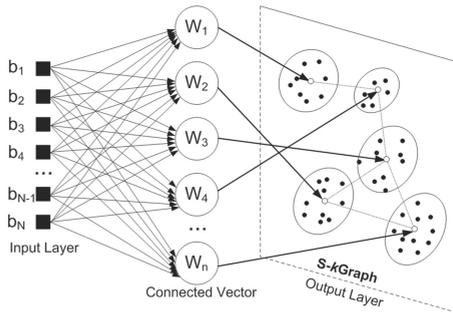


Fig. 7. Model of a signature self-organizing map.

**5.2. Network training algorithm.** In order for the classification to be accurate, the Sig-SOM network must perform a training process. This process adjusts the connection weight vector  $W_j = (w_{1j}, w_{2j}, \dots, w_{Nj})$  in order to accurately classify the input image signatures.

For any image signature  $Sig(I) = b_1b_2 \dots b_N$  in the training data of the Sig-SOM network, we perform the calculation of the measure  $\delta(I_i, W_i)$  (in Definition 14). Here  $\delta(I_i, W_i) \in [0, 1]$  corresponds to connected weights  $W_j = (w_{1j}, w_{2j}, \dots, w_{Nj})$ . Using this measure, we select the winner cluster  $V_J$ , i.e.,  $\delta(I_i, W_J) = \max\{\delta(I_i, W_j) | j = 1, 2, \dots, n\}$ ; after that the algorithm checks the corresponding condition on each cluster or creates new classification on the Sig-SOM network.

**Definition 15. (Rule of training)** Given  $Sig(I) = b_1b_2 \dots b_N$  as a binary signature of image  $I$ , let  $W_m = (w_{1m}, w_{2m}, \dots, w_{Nm})$  be a connection vector at the time  $t$ . The process of training at the time  $t + 1$  is defined as follows:

$$w_{i,m}^{(t+1)} = b_i \vee w_{i,m}^{(t)}, \tag{11}$$

$$w_{i,j}^{(t+1)} = \neg b_i \wedge w_{i,j}^{(t)}, \tag{12}$$

with  $j \neq m$  and  $j = 1, \dots, n$ .

**Theorem 10.** Given the input data  $Sig(I) = b_1b_2 \dots b_N$ , the connection vector  $W_m = (w_{1m}, w_{2m}, \dots, w_{Nm})$  at the time  $t + 1$  is more optimal than at the time  $t$ , i.e.,  $\delta(Sig(I), W_m^{(t)}) \leq \delta(Sig(I), W_m^{(t+1)})$ .

*Proof.* We have

$$\begin{aligned} \delta(I, W_m^{(t+1)}) &= \alpha d(\text{sig}_O^I, \text{sig}_O^{W^{(t+1)}}) + \beta \times d(\text{sig}_C^I, \text{sig}_C^{W^{(t+1)}}) \\ &= \frac{\alpha}{N} \sum_{i=1}^N \text{NOT}(b_i \text{ XOR } w_{i,m}^{(t+1)}) \\ &\quad + \frac{\beta}{M} \sum_{i=1}^M \text{NOT}(b_i \text{ XOR } w_{i,m}^{(t+1)}) \\ &= \frac{\alpha}{N} \sum_{i=1}^N \text{NOT}(b_i \text{ XOR } (b_i \vee w_{i,m}^{(t)})) \\ &\quad + \frac{\beta}{M} \sum_{i=1}^M \text{NOT}(b_i \text{ XOR } (b_i \vee w_{i,m}^{(t)})) \\ &= \frac{\alpha}{N} \sum_{i=1}^N \text{NOT}(b_i \text{ XOR } w_{i,m}^{(t)}) \\ &\quad + \frac{\beta}{M} \sum_{i=1}^M \text{NOT}(b_i \text{ XOR } w_{i,m}^{(t)}). \end{aligned}$$

Conclude that

$$\delta(Sig(I), W_m^{(t)}) \leq \delta(Sig(I), W_m^{(t+1)}).$$

■

After the training process, the Sig-SOM network has new connection weights. This training network is a premise for a similar image query corresponding to each query image. The Sig-SOM network training algorithm is defined as Algorithm 4.

**5.3. Image retrieval algorithm.** The image retrieval algorithm finds out the set of similar images corresponding to the query image  $I_Q$ . On the basis of the Sig-SOM network structure after training, the algorithm determines the cluster  $V_J$  which stores images with highest similarities; afterwards it produces the result including images of cluster  $V_J$ . However, the clusters link together in accordance with the S-kGraph, so that the clusters near cluster  $V_J$  can find out similar images. Therefore, in the similar image retrieval algorithm, we look for images on contiguous clusters of the cluster  $V_J$ , i.e., we look for images of clusters  $V_i$  to satisfy the condition that the distance between two cluster centers do not exceed given  $k\theta$ . After performing the similar image retrieval process, we select the images which have the highest similarity with query image  $I_Q$ . For each query image  $I_Q$  and each distance threshold  $k\theta$ , the similar image retrieval algorithm proceeds as Algorithm 5.

**Algorithm 4.** SOM network training.

---

**Require:** Training dataset  $\Gamma = \{Sig(I_i) = (b_{i1}, b_{i2}, \dots, b_{iN}) | i = 1, \dots, K\}$

**Step 1: Initializing the network**

- 1: Initializing a training set  $\Gamma$  including  $K$  vectors  $Sig(I_i) = (b_{i1}, \dots, b_{iN})$  as binary signatures describing the set of images  $\mathfrak{S} = \{I_i | i = 1, \dots, K\}$ ;
- 2: Initializing an output layer including  $n$  cluster  $\Omega = \{V_j = \langle C_j, k_j \theta \rangle\}$ , with  $C_j = (c_{j1}, \dots, c_{jN})$ ,  $c_{jk} \in \{0, 1\}$ ,  $k = 1, \dots, N$ ,  $j = 1, \dots, n$ ;
- 3: Initializing randomly a weight set  $W = \{W_j | W_j = (w_{1j}, w_{2j}, \dots, w_{nj})$ ,  $w_{ij} \in \{0, 1\}$ ,  $i = 1, 2, \dots, n\}$ ;
- 4: With each input vector  $Sig(I_i)$ ,  $S = \emptyset$  is done and implemented from **Step 2** to **Step 4** as follows:
 

**Step 2: Selecting a winner cluster with  $Sig(I_i)$**

  - 5: **if**  $W = S$  **then**
  - 6: go to *Step 4*;
  - 7: **else**
  - 8: Calculating  $\delta(I_i, W_i)$  with each weight vector  $W_j \in W \setminus S$  of clusters at the output layer;
  - 9: Selecting a winner cluster  $V_J$ , i.e.,  $\delta(I_i, W_J) = \max\{\delta(I_i, W_j) | j = 1, 2, \dots, n\}$ ;
  - 10: **end if**

**Step 3: Checking the condition of a cluster**

  - 11: Calculating the similarity measure  $\phi$  between input vector  $Sig(I_i)$  and center  $C_j = (c_{j1}, v, c_{jN})$ ;
  - 12: **if**  $\phi(Sig(I_i), C_j) \leq k_j \theta$  **then**
  - 13: Input vector  $Sig(I_i)$  belongs to cluster  $V_J$ ;
  - 14: Training  $W_J$  as follows:
    - 15:  $w_{i,J}^{(t+1)} = b_i \vee w_{i,J}^{(t)}$ ;
    - 16:  $w_{i,j}^{(t+1)} = \neg b_i \wedge w_{i,j}^{(t)}$ , with  $j \neq J$  and  $j = 1, \dots, n$ ;
  - 17: **else**
  - 18:  $S = S \cup \{W_J\}$ ; go to *Step 2*;
  - 19: **end if**

**Step 4: Creating a new cluster**

  - 20: Finding cluster  $V_m$  nearest  $V_J$ , i.e.,  $(\phi(C_J, C_m) - k_m \theta) = \min\{(\phi(C_J, C_i) - k_i \theta), i = 1, \dots, n\}$ ;
  - 21: Setting  $k_0 = [(\phi(I_0, I_m) - k_m \theta) / \theta] \geq 0$ ;
  - 22: **if**  $k_0 > 0$  **then**
  - 23: From a cluster  $V_0 = \langle C_0, k_0 \theta \rangle$ :  $C_0 = Sig(I_i)$ ;
  - 24: From a connection vector  $W_0 = (w_{1,0}, \dots, w_{N,0})$ , with  $w_{i,0}$  random in  $\{0, 1\}$ ;
  - 25: **end if**
  - 26: **if**  $k_0 = 0$  **then**
  - 27: Input vector  $Sig(I_i)$  belongs to cluster  $V_m$ ;
  - 28: Training  $W_m$  as follows:
    - 29:  $w_{i,m}^{(t+1)} = b_i \vee w_{i,m}^{(t)}$ ;
    - 30:  $w_{i,j}^{(t+1)} = \neg b_i \wedge w_{i,j}^{(t)}$ , with  $j \neq m$  and  $j = 1, \dots, n$ ;
  - 31: **end if**
  - 32: **return** SOM network;

---

**Algorithm 5.** Image retrieval.

---

**Require:** Query image  $I_Q$  and Sig-SOM network

**Step 1: Initializing**

- 1: Initializing a set of image results  $IMG = \emptyset$ ;
- 2: Initializing a set of clusters  $V = \emptyset$ ;

**Step 2: Clustering**

- 3: Calculating  $\delta(I_Q, W_i)$ ,  $W_j \in W$ ,  $j = 1, \dots, n$
- 4: Choosing winner cluster  $V_J$ , so that  $\delta(I_i, W_J) = \max\{\delta(I_i, W_j) | j = 1, 2, \dots, n\}$ ;
- 5:  $V = V \cup \{V_J\}$ ;

**Step 3: Finding neighboring clusters**

- 6: **for**  $V_i \in V_{SG}$  **do**
- 7: **if**  $\phi(I_m^0, I_i^0) \leq k \theta$  **then**
- 8:  $V = V \cup V_i$ ;
- 9: **end if**
- 10: **end for**

**Step 4: Retrieving similar images**

- 11: **for**  $V_j \in V$  **do**
  - 12:  $IMG = IMG \cup \{I_j^k \in V_j, k = 1, \dots, |V_j|\}$ ;
  - 13: **end for**
  - 14: **return** A set of similar images  $IMG$ ;
- 

## 6. Evaluation

In this section, we present results of experiments on our prototype to prove the efficiency of our model. The experiments are to evaluate accuracy, the size of intermediate data and the query response time.

**6.1. Prototyping system.** The prototype of applying the binary signature and the self-organizing map in a content-based image retrieval system is illustrated in Fig. 8.

The experiment is shown in Fig. 8 and proceeds as follows:

**Phase 1: Creating meta-data**

*Step 1:* For any image in a database of images, extract the regions of interest.

*Step 2:* Based on the image regions of interest, create meta-data in the form of a binary signature.

*Step 3:* Based on Algorithm 4, perform training, classify binary signatures and store them on the S-kGraph.

**Phase 2: Similar image retrieval**

*Step 1:* For each query image, perform conversion into the binary signature based on the image regions of interest.

*Step 2:* Based on the trained SOM network, perform the image retrieval process according to Algorithm 5.

*Step 3:* Having the result of a similar images, arrange images into similar levels and output the result.

**6.2. Experiment.** The application was built based on *Image Processing Toolbox* of Matlab 2015. The

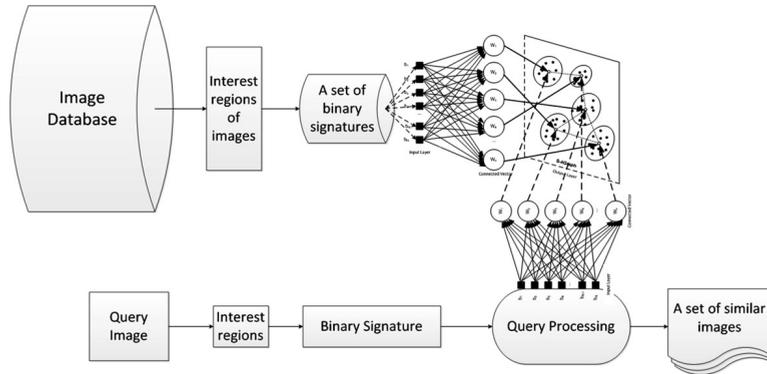


Fig. 8. Model of a CBIR based on Sig-SOM and the S-kGraph.

experiment was performed on a computer with Intel(R) CoreTM i7-2620M, 2.70 GHz CPU, 4 GB RAM running Windows 7 Professional.

Processing was made on image databases including COREL, Wang and MSRDI. For any query image, we retrieve the most similar images in the database. Then we compare the list of subjects of images to evaluate the accurate method. Figure 9 shows the results of the proposed retrieval method.

In order to evaluate the performance of the proposed CBIR system, we calculate the values including *precision*, *recall*, the *F-measure* and the *true positive rate* of the ROC (*receiver operating characteristic*) curve. According to Alzu'bi *et al.* (2015), *precision* is the ratio of the number of relevant images within the first *k* results to the number of total retrieved images. *Recall* is the ratio of the number of relevant images within the first *k* results to the number of total relevant images. The *F-measure* is the harmonic mean of precision and recall. The corresponding formulas are as follows:

$$\text{precision} = \frac{(\text{relevant images} \cap \text{retrieved images})}{\text{retrieved images}}, \quad (13)$$

$$\text{recall} = \frac{(\text{relevant images} \cap \text{retrieved images})}{\text{relevant images}}, \quad (14)$$

$$\text{F-measure} = 2 \times \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}. \quad (15)$$

The COREL database has 1,000 images and is divided into 10 subjects. There are 100 images in each subject. Figure 10 describes the *precision-recall* curve and the ROC curve. For the sake of clarity, we calculate the means of performance indexes and show them in Fig. 11.

In a similar method, the Wang database (10,800 images) is divided into 80 subjects, each containing from 100 to 400 images. The MSRDI database (16,710 images) is divided 31 subjects, each containing from 300 to 500 images. The results of the query process are described in Figs. 12–15. These *precision-recall* curves and ROC curves show that the proposed method is very effective.

Moreover, we compare the performance of the proposed method with the state-of-the-art methods such as the S-tree, the S-kGraph, the points-of-interest method, the color histogram, the fuzzy color histogram, or local feature regions (Wang *et al.*, 2010; 2013; Nascimento *et al.*, 2002; Le and Van, 2013; Van and Le, 2014a; 2014c; 2014b). The results of comparisons are shown in Tables 1–5.

In order to assess the query time of the proposed method, we calculate the average query time of each subject in the images database (COREL, Wang, MSRDI). Additionally, we compare the results with other co-techniques such as linear methods, the S-tree, or the S-kGraph (Le and Van, 2013; Van and Le, 2014a; 2014c; 2014b). The results are described in Figs. 16–18 and Table 6.

Furthermore, we compare the size of image databases with that of the binary signature database in Table 7. The results show that the query storage is reduced significantly.

Table 1. Average accuracy on the COREL database.

Accuracy	S-tree	S-kGraph	Our method
recall	0.542	0.779	0.862
precision	0.549	0.748	0.884
F-measure	0.545	0.763	0.873

Table 2. Average accuracy on the Wang database.

Accuracy	S-tree	S-kGraph	Our method
recall	0.542	0.742	0.806
precision	0.567	0.746	0.877
F-measure	0.565	0.744	0.839

## 7. Conclusions

In the paper, we built a CBIR based on the Sig-SOM network and the signature graph. According to the



Fig. 9. Some results of similar image retrieval.

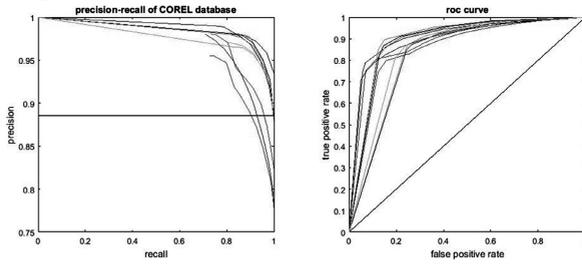


Fig. 10. Accuracy of the system on the COREL database.

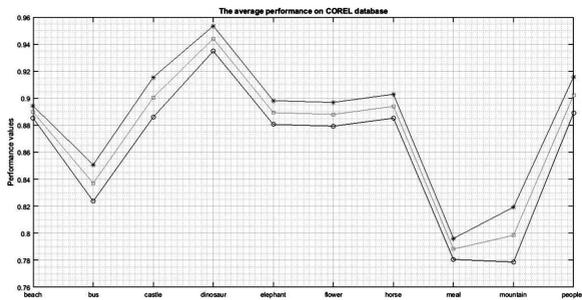


Fig. 11. Average accuracy of the system on the COREL database.

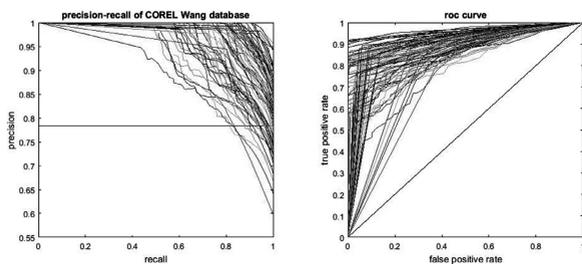


Fig. 12. Accuracy of the system on the Wang database.

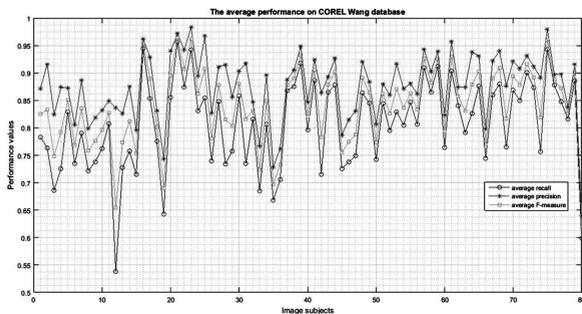


Fig. 13. Average accuracy of the system on the Wang database.

Table 3. Average accuracy on the MSRDI database.

Accuracy	S-tree	S-kGraph	Our method
recall	0.562	0.752	0.716
precision	0.568	0.758	0.797
F-measure	0.565	0.755	0.754

experiment, the image retrieval problem can exploit the binary signature very efficiently. At the same time our approach helps to quickly query and significantly reduce

Table 4. Average accuracy on the COREL database.

Accuracy	Interest points	Histogram	Our method
recall	0.7045	0.607	0.862
precision	0.657	0.488	0.884
F-measure	0.680	0.541	0.873

Table 5. Average accuracy on the COREL database.

Accuracy	F-histogram	Region	Our method
recall	0.612	0.782	0.862
precision	0.509	0.850	0.884
F-measure	0.556	0.814	0.873

Table 6. Average values of query time (s).

DB	linear	S-tree	S-kGraph	Our method
COREL	2.000	0.981	0.463	0.0789
Wang	3.353	1.143	0.889	0.442
MSRDI	6.113	2.655	2.320	1.128

Table 7. Comparison of storage volumes sizes.

DB	image DB	signature	reduce rate
COREL	56.93 Mb	0.55 Mb	99.043%
Wang	39.92 Mb	4.89 Mb	87.752%
MSRDI	343.52 Mb	7.44 Mb	97.835%

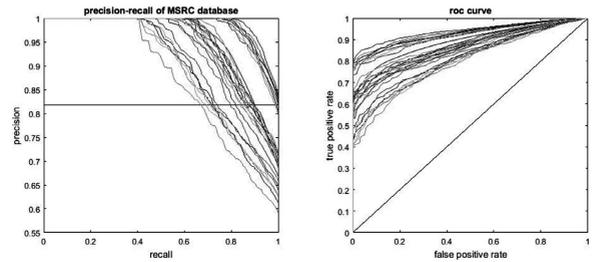


Fig. 14. Accuracy of the system on the MSRDI database.

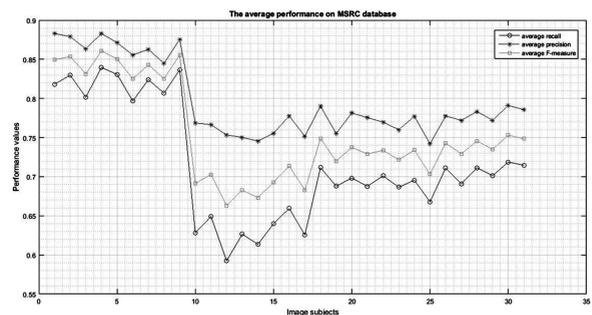


Fig. 15. Average accuracy of the system on the MSRDI database.

the query storage. In order to improve the accuracy, we extract a set of image objects of interest and describe them based on basic objects for creating a binary signature. With that, we build a data structure to store and classify

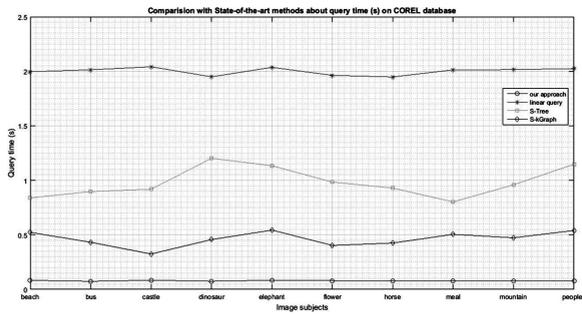


Fig. 16. Query time of a retrieval process on COREL.

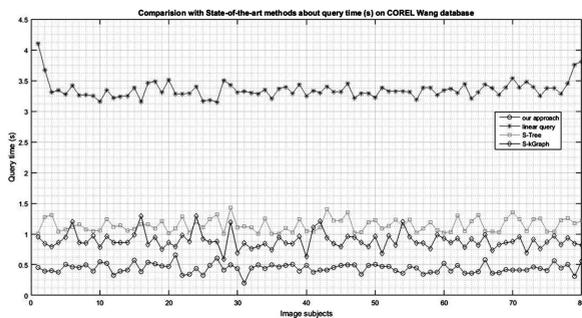


Fig. 17. Query time of a retrieval process on Wang.

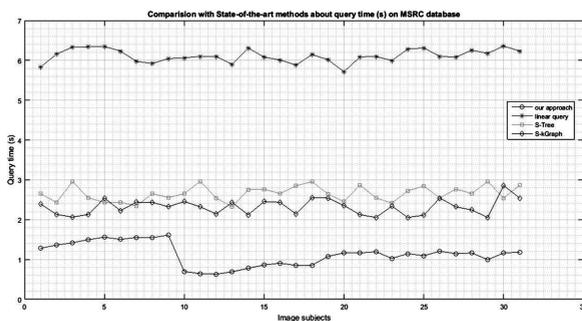


Fig. 18. Query time of a retrieval process on MSRDI.

similar images, and then provide a classification algorithm for a set of images.

### Acknowledgment

The authors wish to thank the anonymous reviewers for their helpful comments and valuable suggestions. We would also like to thank the Faculty of Information Technology, University of Sciences/Hue University, Vietnam, and the Center for Information Technology, HCMC University of Food Industry, Vietnam.

### References

Abdesselam, A., Wang, H.H. and Kulathuramaiyer, N. (2010). Spiral bit-string representation of color for image retrieval,

*International Arab Journal of Information Technology* 7(3): 223–230.

Acharya, T. and Ray, A.K. (2005). *Image Processing: Principles and Applications*, John Wiley and Sons, Hoboken, NJ.

Alzu'bi, A., Amira, A. and Ramzan, N. (2015). Semantic content-based image retrieval: A comprehensive study, *Journal of Visual Communication and Image Representation* 32: 20–54.

Bahri, A. and Hamid, Z. (2011). EMD similarity measure and metric access method using EMD lower bound, *International Journal of Computer Science & Emerging Technology* 2(6): 323–332.

Bartolini, I., Ciaccia, P. and Patella, M. (2010). Query processing issues in region-based image databases, *Knowledge and Information Systems* 25(2): 389–420.

Chappell, T. and Geva, S. (2013). Efficient top-k retrieval with signatures, *Proceedings of the 18th Australasian Document Computing Symposium, ADCS'13, Brisbane, Australia*, pp. 10–17.

Wang (2016). <http://wang.ist.psu.edu>.

COREL (2016). <http://www.corel.com>.

Microsoft (2016). <http://research.microsoft.com>.

Kim, S., Park, S. and Kim, M. (2003). Central object extraction for object-based image retrieval, *Image and Video Retrieval, CIVR 2003, Urbana-Champaign, IL, USA*, pp. 39–49.

Kompatsiaris, I. and Srinatzis, M.G. (2000). Spatiotemporal segmentation and tracking of objects for visualization of video conference image sequences, *IEEE Transactions on Circuits and Systems for Video Technology* 10(8): 1388–1402.

Kumar, H.C.S., Raja, K.B., Venugopal, K.R., and Patnaik, L.M. (2009). Automatic image segmentation using wavelets, *International Journal of Computer Science and Network Security* 9(2): 305–313.

Le, T.M. and Van, T.T. (2013). Image retrieval system based on EMD similarity measure and S-tree, *Intelligent Technologies and Engineering Systems, ICITES-2012, Changhua, Taiwan*, pp. 139–146.

Li, Y., Jin, J.S. and Zhou, X. (2005). Video matching using binary signature, *Proceedings of the IEEE International Symposium on Intelligent Signal Processing and Communication Systems, Hong-Kong, China*, pp. 317–320.

Liu, G.-H. and Yang, J.-Y. (2013). Content-based image retrieval using color difference histogram, *Pattern Recognition* 46(1): 347–357.

Liu, L., Lu, Y. and Suen, C.Y. (2015). Variable-length signature for near-duplicate image matching, *IEEE Transactions on Image Processing* 24(4): 1282–1296.

Liu, Y., Zhang, D., Lu, G. and Ma, W.-Y. (2007). A survey of content-based image retrieval with high-level semantics, *Pattern Recognition* 40(1): 262–282.

Manolopoulos, Y., Nanopoulos, A. and Tousidou, E. (2003). *Advanced Signature Indexing for Multimedia and Web Applications*, Springer Science Business Media, New York, NY.

- Marques, O. and Furht, B. (2002). *Content-Based Image and Video Retrieval*, Springer Science + Business Media, New York, NY/London.
- Mezaris, V., Kompatsiaris, I. and Strintzis, M.G. (2004). Still image segmentation tools for object-based multimedia applications, *International Journal of Pattern Recognition and Artificial Intelligence* **18**(4): 701–725.
- Muneesawang, P., Zhang, N. and Guan, L. (2014). *Multimedia Database Retrieval: Technology and Applications*, Springer, Cham/Heidelberg.
- Nascimento, M.A. and Chitkara, V. (2002). Color-based image retrieval using binary signatures, *SAC 2002, Madrid, Spain*, pp. 687–692.
- Nascimento, M.A., Tousidou, E., Chitkara, V. and Manolopoulos, Y. (2002). Image indexing and retrieval using signature trees, *Data & Knowledge Engineering* **43**(1): 57–77.
- Ozkan, S., Esen, E. and Akar, G.B. (2014). Visual group binary signature for video copy detection, *Proceedings of the IEEE International Conference on Pattern Recognition, ICPR-2014, Stockholm, Sweden*, pp. 3945–3950.
- Singha, M. and Hemachandran, K. (2012). Content based image retrieval using color and textual, *Signal & Image Processing: An International Journal* **3**(1): 39–57.
- Tang, Z., Zhang, X., Dai, X., Yang, J. and Wu, T. (2013). Robust image hash function using local color features, *AEU—International Journal of Electronics and Communications* **67**(8): 717–722.
- Van, T.T. and Le, T.M. (2014a). Color image retrieval using fuzzy measure hamming and S-tree, *Advances in Computer Science and its Applications, CSA-2013, Vietnam*, pp. 615–620.
- Van, T.T. and Le, T.M. (2014b). Image retrieval based on binary signature and S-kGraph, *Annales Universitatis Scientiarum Budapestinensis de Rolando Eötvös Nominatae, Sectio Computatorica* **43**(2): 105–122.
- Van, T.T. and Le, T.M. (2014c). RBIR based on signature graph, *Proceedings of the IEEE International Conference on Computer Communication and Informatics, ICCCI-2014, Coimbatore, India*, pp. 1–4.
- Wang, X.-Y., Wu, J.F. and Yang, H.Y. (2010). Robust image retrieval based on color histogram of local feature regions, *Multimedia Tools and Applications* **49**(2): 323–345.
- Wang, X.-Y., Yang, H.-Y., Li, Y.-W. and Yang, F.-Y. (2013). Robust color image retrieval using visual interest point feature of significant bit-planes, *Digital Signal Processing* **23**(4): 1136–1153.
- Yoo, H.-W., Jung, S.-H., Jang, D.-S. and Na, Y.-K. (2002). Extraction of major object features using VQ clustering for content-based image retrieval, *Pattern Recognition* **35**(5): 1115–1126.
- Zhou, W., Li, H., and Tian, Q. (2014). *Academic Press Library in Signal Processing*, Vol. 5, Elsevier, Oxford.



**Thanh The Van** was born in 1979. He received the B.Sc. degree in mathematics and computer science from the University of Science/HCMC National University, Vietnam, in 2001. In 2008, he obtained an M.Sc degree in computer science from Vietnam National University. Since 2012, he has been a Ph.D. candidate at Hue University, Vietnam. His research interests include image processing and image retrieval.



**Thanh Manh Le** was born in 1953. He received a Ph.D. degree in computer science from Budapest University (ELTE), Hungary, in 1994. He became an associate professor at Hue University, Vietnam, in 2004. His research interests include databases, knowledge bases and logic programming.

Received: 11 July 2015

Revised: 27 December 2015

Re-revised: 15 February 2016

Accepted: 2 March 2016