Journal of Computer Science and Cybernetics, V.36, N.1 (2020), 49–67
DOI 10.15625/1813-9663/36/1/14347

A SELF-BALANCED CLUSTERING TREE FOR SEMANTIC-BASED IMAGE RETRIEVAL

NGUYEN THI UYEN NHI[1,3], VAN THE THANH[2], LE MANH THANH[1]

[1] *Faculty of Information Technology, University of Science - Hue University, Vietnam*

[2] *Office of Scientific Research Management and Postgraduate Affairs, HCMC University of Food Industry, Vietnam*

[3] *Faculty of Statistics and Informatics, University of Economics, The University of Danang, Vietnam; nhintu@due.edu.vn*

Abstract. The image retrieval and semantic extraction play an important role in the multimedia systems such as geographic information system, hospital information system, digital library system, etc. Therefore, the research and development of semantic-based image retrieval (SBIR) systems have become extremely important and urgent. Major recent publications are included covering different aspects of the research in this area, including building data models, low-level image feature extraction, and deriving high-level semantic features. However, there is still no general approach for semantic-based image retrieval (SBIR), due to the diversity and complexity of high-level semantics. In order to improve the retrieval accuracy of SBIR systems, our focus research is to build a data structure for finding similar images, from that retrieving its semantic. In this paper, we proposed a data structure which is a self-balanced clustering tree named C-Tree. Firstly, a method of visual semantic analysis relied on visual features and image content is proposed on C-Tree. The building of this structure is created based on a combination of methods including hierarchical clustering and partitional clustering. Secondly, we design ontology for the image dataset and create the SPARQL (SPARQL Protocol and RDF Query Language) query by extracting semantics of image. Finally, the semantic-based image retrieval on C-Tree (SBIR_CT) model is created hinging on our proposal. The experimental evaluation 20,000 images of ImageCLEF dataset indicates the effectiveness of the proposed method. These results are compared with some of recently published methods on the same dataset and demonstrate that the proposed method improves the retrieval accuracy and efficiency.

Keywords. SBIR; Image retrieval; Similar image, C-tree; Ontology.

1. INTRODUCTION

Recently a collection of digital images has been rapidly increasing and continues to enhance in future with the development of the Internet. Image data plays an important role in many multimedia systems such as geographic information systems (GISs),

This paper is selected from the reports presented at the 12th National Conference on Fundamental and Applied Information Technology Research (FAIR'12), University of Sciences, Hue University, 07–08/06/2019.

hospital information systems (HISs), digital library systems (DLSs), biomedicine, education and entertainment, etc. This yields an exigent demand for developing highly effective image retrieval systems to satisfy human needs. Many image retrieval systems have been developed, such as Text-based Image Retrieval (TBIR) [24], Content-based Image Retrieval (CBIR [8, 10]). These systems which retrieve images by keywords, text or visual contents still lack the semantic analysis of images [1, 3], so the search results usually return the images unrelated, performance of image retrieval is still far from user's expectations. To overcome the above disadvantages in TBIR and CBIR, semantic based image retrieval (SBIR) is proposed. SBIR extracts features to identify meaning of images; then, it retrieves the related images in visual features and extracts semantics of contents of these images [2, 12, 23]. There are two challenges with this approach. The first challenge of SBIR is to extract visual features after that map it into semantics to describe content of image [20, 28]. The second challenge is to describe semantics and build models for image retrieval [11, 15]. The advanced techniques in SBIR include mainly the following categories: (1) using object ontology to define high-level concepts [17, 19], (2) using machine learning methods to associate low-level features with high-level semantics [6, 7], (3) using both the visual content of images and the textual information obtained from the Web for WWW image retrieval [14, 18], etc. However, the SBIR problem is still partially resolved because the proposed approaches strongly depend on an external reliable resource such as automatically annotation images, ontology, and learning datasets. There is still no general approach for SBIR, due to the diversity and complexity of high-level semantics. Therefore, SBIR has attracted great interest in recent years. Many researchers have found that tree structure is an extensively researched area for classification tasks and has great potential in image semantic learning [11, 15]. Cluster tree keeps the tree simple by controlling its size and complexity, since a cumbersomely large tree leads to misclassifications.

The problems discussed above provide the motivation to develop an SBIR system with high-level semantics derived using cluster tree learning. In this paper, we build a self-balanced clustering tree structure, named C-Tree, to store visual feature vectors of images. C-Tree is a combination of methods including hierarchical clustering and partitional cluster, which creates a data model that supports the retrieval process. This data model is created by semi-supervised learning techniques. C-Tree has been built for classification tasks, and keeps the tree simple by controlling its size and complexity. Besides, semantically relevant images will be retrieved in lesser amount of time. Every image in the database is segmented into different regions, represented by their color, texture features, spatial location, shape, etc. To associate low-level region features with high-level image concepts, we propose a C-Tree based image semantic learning algorithm. SBIR based on C-Tree (SBIR_CT) is built. The experiment of SBIR_CT is executed on ImageCLEF dataset [29, 30]. We identify the semantics of similar images on ontology, which describes semantics of visual features of images.

The contributions of the paper include: (1) building an automatic clustering model by proposing a self-balanced clustering tree structure (C-Tree) to store low-level visual content of the images; (2) proposing model and algorithms of SBIR_CT to retrieve semantics of similar images; (3) building ontology for image dataset on the basis of triple language RDF (Resource Description Framework) [16, 17] and creating a SPARQL command [31, 32] to retrieve similar images based on visual word vector; (4) constructing the SBIR_CT system based on proposed model and algorithms to implement the evaluation on ImageCLEF dataset.

The rest of this paper is as follows. Section 2 gives a brief overview of related approaches to high-level semantic image retrieval systems. In Section 3 we present algorithms for building self-balanced clustering C-Tree. In Section 4, we describe the components of SBIR_CT system and create ontology for image dataset. In Section 5, we build the experiment and evaluate the effectiveness of the proposed method. Conclusions and future works are presented in Section 6.

2. RELATED WORKS

Semantic-based image retrieval has become an active research topic in recent times. There were many techniques of image retrieval, which have been implemented aiming to reduce the "semantic gap" by modeling high-level semantics, such as techniques to build a model for mapping between low-level features and high-level semantics [2, 21], query techniques based on ontology to accurately describe semantics for images [18, 25], techniques for classification data [12, 13, 17], etc.

In 2008, Liu Y., et al. [15] proposed a region-based image retrieval system with high-level semantic learning. A method to employ decision tree induction for image semantic learning, named DT-ST, was introduced. During retrieval, a set of images whose semantic concept matches the query is returned. Their semantic image retrieval system allowed users to retrieve images using both query by region of interest and query by keywords, and experimented on 5000 COREL images. However, the experiments in this paper were conducted using query by single specified region.

In 2013, Sarwar S. et al. [23] proposed an ontology based image retrieval framework from a corpus of natural scene images by imparting human cognition in the retrieval process. Domain ontology had been developed to model qualitative semantic image descriptions and retrieval, thereafter could be accomplished either using a natural language description of an image containing semantic concepts and spatial relations. This system is tested on 300 natural scene images from the SCULPTEUR Project, which are manually classified.

Poslad S. and Kesorn K. (2016) [21] proposed a Multi-Modal Incompleteness ontology-based (MMIO) system for image retrieval based upon fusing two derived indexes. The two indexes were fused into a single indexing model: The first index exploits low-level features extracted from images to represent the semantics of visual content, by restructuring visual word vectors into an ontology model. The second index relied on a textual description to extract the concepts, and properties in ontology.

Y. Cao et al. [4] used CNN to classify images and create binary-featured vectors. On this basis, the authors have proposed a DVSH model to identify a set of semantic analog images. However, this method must implement two processes for classifying visual and semantic features. If an image lacks one of these features, the same image is retrieved incorrectly. This method has not yet been mapped from visual features to high-level semantics of images. However, this method must perform two classification processes of visual and semantic features. If an image lacks one of these two features, the retrieved similar images are inaccurate. Furthermore, the method has not yet mapped from visual features to semantics of images.

In 2017, Allani Olfa et al. [2] proposed pattern-based image retrieval system SemVisIR, which combined semantic and visual features. They organized the image dataset in a graph of patterns which are automatically built for the different domains by clustering algorithms.

SemVisIR modeled the visual aspects of images through graphs of regions and assigning them to automatically built ontology modules for each domain. Their system was implemented and evaluated on ImageCLEF. The performance of this method is not high compared to the previous methods, because the semantics of images are retrieved directly on the ontology.

Hakan Cevikalp et al. [5] proposed a method for large-scale image retrieval by using binary hierarchical trees and transductive support vector machines (TSVM). TSVM classifier was used to separate both the labeled and unlabeled data samples at each node of the binary hierarchical trees. The method had been experimented on ImageCLEF and compare the effectiveness with other methods. However, this method had not yet implemented semantic queries for images and had not yet classified the semantics of images.

M. Jiu et al. (2017) [13] proposed a novel method that learns deep multi-layer kernel networks for image annotation. The system was created by semi-supervised learning (SSL) that learns deep nonlinear combinations. SSL models the topology of both labeled and unlabeled data resulting into better annotation performances. The SVM technique is applied to layering images at the output layer to extract a semantic level according to visual information for similar pocket-based images from BoW (Bag-of-Words). The method is evaluated on ImageCLEF dataset. In this method, neural network is fixed the number of layers, so the classification of deep learning technique is limited.

Zahid Mehmood et al. (2018) [14] proposed a novel image representation based on the weighted average of triangular histograms of visual words using support vector machine. The proposed approach was added the image spatial contents to the inverted index of the BoVW (Bag-of-Visual-Words) model, to reduce semantic gap. Image annotations automatically based on classification scores. The method was tested on the COREL dataset.

The recent approaches focused on methods for mapping low-level features to semantic concepts by using supervised or unsupervised machine learning techniques [27, 28]; building data models to store low-level contents of images; building ontology to define the high-level concepts, etc. On the basis of inheriting and overcoming limitations of related works, we propose methods to improve performance of SBIR. The SBIR_CT system in this article is implemented by: (1) using queries by multiple regions, (2) automatically classifying image semantics, (3) retrieving semantics based on ontology.

3. A SELF-BALANCED CLUSTERING TREE

In this section, we build a self-balanced clustering tree structure, named C-Tree, to create an automatic clustering data mining model for feature vectors of dataset.

3.1. The data of C-Tree

In this paper, each image is segmented into different regions according to Hugo Jair Escalantes method [8, 15]. Each region is extracted a feature vector including: Region area, width and height; Features of locations including mean and standard deviation in the x and y-axis; Features of shape including boundary/area, convexity; Features of colors in RGB and CIE-Lab space including average, standard deviation and skewness, etc. Each feature vector is assigned a label and mapped to a semantic class to describe visual semantics for each image region. Each image is extracted with many feature vectors and many semantic descriptions.

For our ImageCLEF dataset, there are 276 classes. Each of these 276 classes is given a concept label from 0, 1,..., to 275 in sequence. The input attributes of C-Tree are the low-level region features and the output is the concepts from classes.

Figure 1. Original image and segmented image

3.2. C-Tree structure

C-Tree is a multi-branch tree consisting of a set of vertices and edges. Vertices of C-Tree include a root node, a set of internal nodes, and a set of leaf nodes. C-Tree edges are the links l from parent node to child node, which are quantified by the similarity measure. The C-Tree is a tree that grows in height in the root direction. Each node of the C-Tree stores a set of elements E. Each element E stores a vector feature f of an image region, a concept label c, and a link l to a child node or an identifier id of the image, $E = \langle f, c, l, id \rangle$. If $id = null$, $l \neq null$ then we have an element of the internal node InE. In contrast, $id \neq null$, $l = null$, we have an element of leaf node lvE. C-Tree is organized in a clustering structure based on Minkowski measure to cluster feature vectors of image regions. C-Tree is defined as follows.

Definition 1. Let C-Tree be a clustering tree, which is connected in a parent-child relationship due to the regions representing the similar measure of feature vectors.

a) A root node is the topmost node without a parent, containing elements of internal node $InE : root = \{inE_i\}$, where $inE = \langle f_c, c_k, l \rangle$, f_c is feature vector of the center of child node, which has the link l, c_k is the set of concept labels of child node;

b) Internal node $inNode$ is a node with at least one child, containing elements of internal node InE, set of internal nodes I is: $I = \{inNode\}$, where $inNode = \{inE_i | i \geq 1\}$;

c) Leaf node $lvNode$ is a node without a child node, contains elements of leaf node lvE, set of leaf nodes L is $L = \{lvNode\}$, where $lvNode = \{lvE_i | i \geq 1\}$, $lvE = \langle f, c, id \rangle$;

d) Two nodes at the same level if they have the same parent node;

e) p_Node is called the parent of c_Node if p_Node has an element, which is linked to c_Node;

Based on Definition 1, the creation of the C-Tree is described according to the following rules.

Definition 2. Rules for creating C-Tree

a) At the beginning, C-Tree has only one empty root node;

b) Each element is added to a leaf node of the C-Tree, basing on the rules of the nearest branch selected in similarity measure;

c) A leaf is split into k-leaves if the number of elements exceeds M, these new leaves are linked by k-new elements of parent node based on Definition 1(a). If this parent node is full, it is split by (d) rule;

d) A node is split into k-nodes if the number of elements exceeds M; at the same time, k- new elements of parent node are created.

Because image data is constantly increasing, so C-Tree must be able to grow. C-Tree height is $h = \log_M(N)$, for M, N are the maximum numbers of elements of a node and the maximum number of nodes.

Figure 2 describes the structure of a self-balanced clustering tree, including a root, set of internal nodes, and set of leaf nodes. A leaf node contains feature vectors, image identifiers of regions. The internal node contains the feature vectors of the center child nodes and the links with those child nodes.

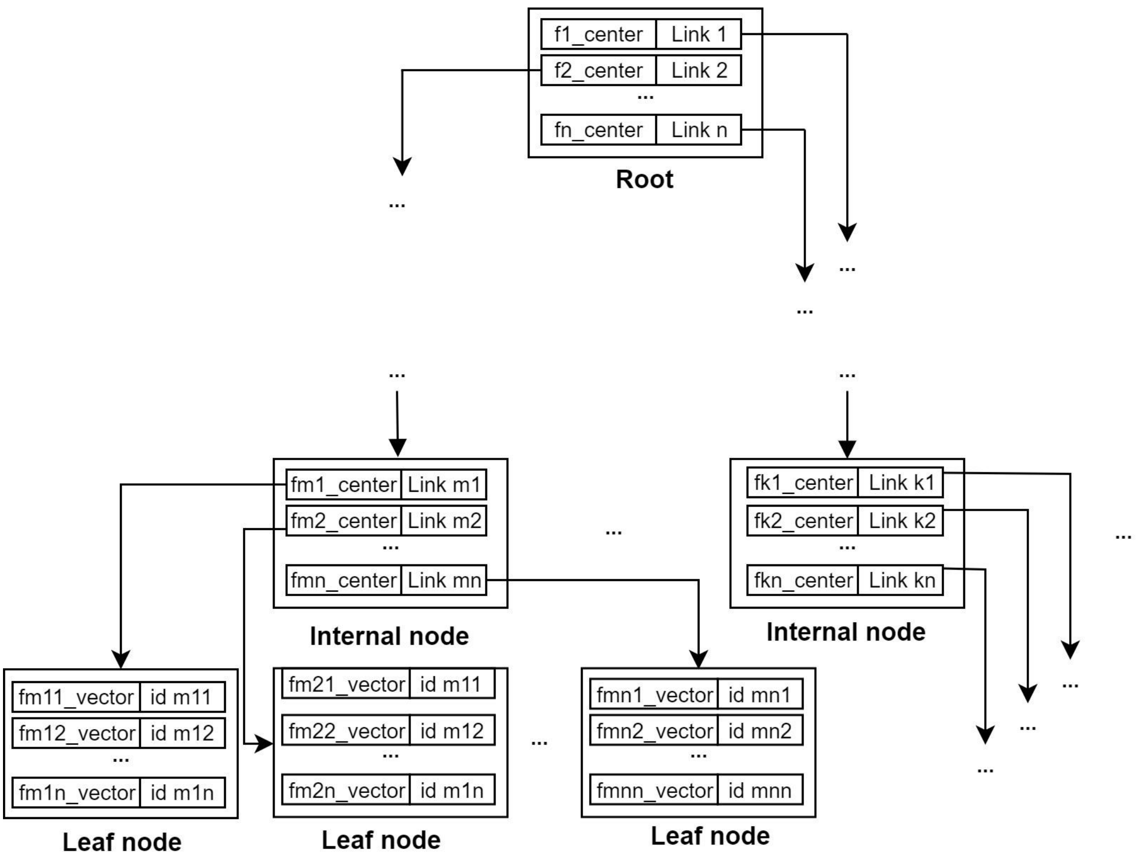


Figure 2. Structure self-balanced clustering C-Tree

Theorem 1. *The C-Tree is a multi-branched tree that balances in height from the root to the leaf node in all directions.*

Proof. According to Definition 2, when a leaf node is split into k-leaf node, the parent node element is formed. In addition, when an internal node is split the elements of the adjacent parent node is formed. Moreover, C-Tree grows in the root direction, so the height of the leaf nodes increases equally. Therefore, C-Tree is a height-balanced tree in every direction from root to leaf node. ∎

Theorem 2. *For each feature vector:*

(i) *There always exists only one leaf node in the C-Tree to store vector f;*

(ii) *The feature vector f is stored on the most suitable leaf node based on similarity measure;*

Proof:

(i) At each internal node of the C-Tree, we select only one direction to find location, which stores the feature vector f. Therefore, if browsing from the root node to the leaf node, only the most appropriate leaf node is selected to store the vector f. In case the node is split into k-cluster, the vector f is distributed to a single cluster according to the algorithm K-means, meaning that the vector f belongs to only one leaf node.

(ii) Because every time we add a vector f to the C-Tree, we have to browse from the root node and find the nearest branch, so we can only find one next child. Therefore, we can find only one leaf with the closest center, meaning that the leaf node is the most suitable for adding vector f. ∎

3.3. Algorithms creating C-Tree

The creating C-Tree process is based on inserting and splitting nodes to cluster feature vectors and the identifier of the images with the metadata of those images. Therefore, algorithms for creating C-Tree include: Splitting the node, updating the cluster center, and inserting an element into the tree.

3.3.1. Splitting a node on C-Tree

Each element $E = \langle f, c, id \rangle$ is inserted into the appropriate leaf node, so C-Tree updates the center. If the element's number of node is greater than the limit value M of each node, the split node process will be performed and the C-Tree grows balanced (according to Theorem 1).

When C-Tree executes the split process, each node is split into k-nodes by selecting k elements of farthest node to create k new node, then distribute the feature vectors of the node to the newly node based on the Minkowski measure. After each feature vector distribution into new clusters, the cluster center is updated. The element of parent node is the center of the child node. When the parent node is full, proceed to split the parent node into k-nodes.

Algorithm: SN

Input: A split node
Output: C-Tree clustering after split
Function SN(v);
Begin
//Select k elements with the furthest distribution according to Minkowski measure
$E_c = \{E_i | Minkowski(E_k.f, E_t.f) \leq Minkowski(E_i.f, E_j.f);\ i,j = 1..k;\ k,t = 1..count\}$;
$Create\ node\ v_i = \{E_i\}$;
For $f \in v$ **do**
$pos = argmin\{Minkowski(f, v_i.E[m].f) | i = 1..k;\ m = 1..v_i.count\}$;
$v_{pos}.count = v_{pos}.count + 1$;
$v_{pos}.f = f$;
EndFor
If $(v_{center}! = null)$ **then** $(v_{parent} = avg(v_i))$;
$UCE(v_{parent})$;
End
If $(v_{parent}.count > M)$ **then** $SN(v_{parent})$;
End.

Proposition 1. *The SN Algorithm executes splitting a node on a C-Tree with complexity $O(M \times N)^2$, where M, N are respectively maximum number elements in a node and maximum number nodes of C-Tree.*

Proof. When a node is split, in the worst case, the SN Algorithm must call recursively from leaf node to root, i.e. all N nodes of C-Tree must be browsed. Each time the node is split, the SN Algorithm must perform M comparisons to distribute to k-clusters. Therefore, the complexity of the SN Algorithm is $O(M \times N)^2$. ∎

3.3.2. Updating the cluster center on C-Tree

Updating the cluster center is to create a path from the leaf node to the root. Therefore, this update is performed from a node v to the root and executed basing on the UCE Algorithm as follows.

Algorithm 2 UCE

Input: node v
Output: C-Tree clustering after updating
Function $UCE(v)$;
Begin
If $(v.Element_{parent}! = null)$ **then**
$f_v = avg\{v.E[i].f | i = 1..count\}$;
$v.Element_{parent}.f = f_v$;
EndIf
If *(v.parent!=null)* **then**
$v = v.parent$;
$UCE(v)$;
Endif

End.

Proposition 2. *The UCE Algorithm has a complexity $O(M \times N)$, where M, N are respectively maximum number elements in a node and maximum number nodes of C-Tree.*

Proof. In the worst case, the UCE Algorithm must update the center of the node from leaf node to the root and traverses the elements of each node and N nodes of C-Tree. Therefore, the complexity of the UCE Algorithm is $O(M \times N)$.

3.3.3. Inserting an element into the C-Tree

For each element $E = \langle f, c, id \rangle$ is inserted into the C-Tree, it will take priority to follow the cluster with the nearest similarity measure. This process will be approved until a suitable leaf node is found due to Minkowski measure.

Algorithm 3 INF

Input: feature vector f and node v

Output: C-Tree clustering after inserting

```
Function INF(f, v);
Begin
    If (v is Leaf) then
            v.count=v.count+1;
            v.E[count].f=f;
            v.E[count].id=id;
            v.E[count].l=null;
            If (v.Element_parent!=null) then UCE(v);
            EndIf
            If (v.count > M) then SN(v);
            Endif
            return C-Tree;
    Else
            pos = argminMinkowski(f, v.E[i].f)|i = 1..count;
            v=v.E[pos].l;
            INF(f,v);
    EndIf
End.
```

Proposition 3. *The complexity of the INF Algorithm is $O(M \times N)$, where M, N are respectively maximum number elements in a node and maximum number nodes of C-Tree.*

Proof. The INF Algorithm in turn executes the browse from the root to the leaf node, through the M elements of node and N nodes of C-Tree. Therefore, the complexity the INF Algorithm is $O(M \times N)$.

4. THE SEMANTIC-BASED RETRIEVAL IMAGE SBIR_CT SYSTEM

4.1. The architecture of SBIR_CT system

The general architectural model of SBIR_CT system is described in Figure 3. The SBIR_CT system consists of two phases including: (1) extracting feature vectors of image datasets to generate data for training a self-balanced clustering tree based on the K-means algorithm and Minkowski measure; building ontology for the image dataset; (2) for each query image, visual features are extracted to query on C-Tree, the set of similar images and visual word vector are generated. Then, the SPARQL command is generated automatically from visual word vector to query on ontology.

4.1.1. Pre-processing phase of SBIR_CT

Each image in the dataset is segmented into different regions, which are extracted feature vectors to generate inputs for training a self-balanced clustering tree based on the K-means algorithm and Minkowski measure. At the same time, ontology is built for the image dataset. The process of pre-processing phase consists of the following steps:
Step 1. Extract data sample including feature vectors f and semantic category w of each region corresponding to each image in dataset;
Step 2. Train a self-balanced clustering tree structure, named C-Tree, to store data samples based on K-means algorithm and Minkowski measure;
Step 3. Build ontology as RDF triple language to describe semantics for image dataset.

4.1.2. Image retrieval phase of SBIR_CT

The process of the query phase includes the following steps:
Step 1. For each query image IQ, the feature vectors of regions are extracted and retrieved on C-Tree; the result is a set of similar images and visual word vector.
Step 2. Create a SPARQL query based on the visual word vector and retrieve on ontology to produce a set of URIs and the metadata of images;
Step 3. Arrange similar images by similarity measure of the query image.

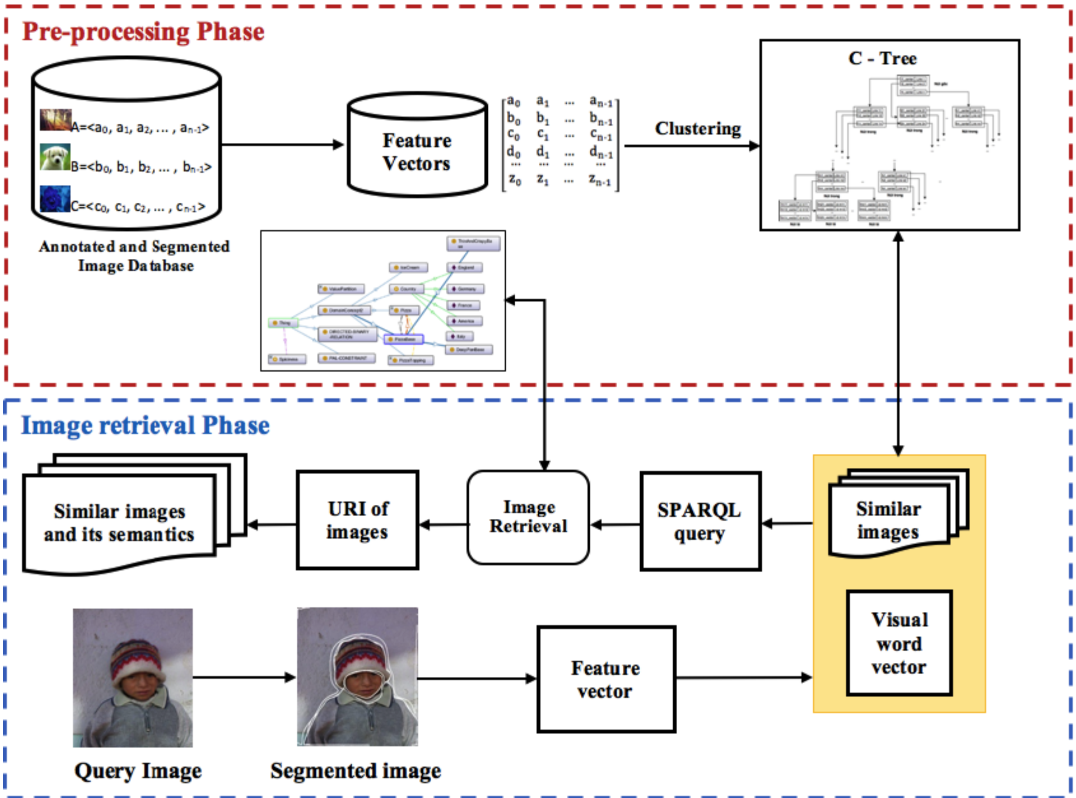


Figure 3. Model of semantic-based image retrieval SBIR_CT

4.2. Visual word vector

Each image is a set of visual feature vectors of each region and a set of labels assigned to each vector. These labels are mapped into concept classes to give a visual word. Each image is represented by a set of visual words. The image retrieval on C-Tree creates a set of similar images and a set of visual words that represent this dataset. Visual word vector is based on a set of visual words, taking words with the highest frequency. The number words of the visual word vector equals the number of visual words of the query image.

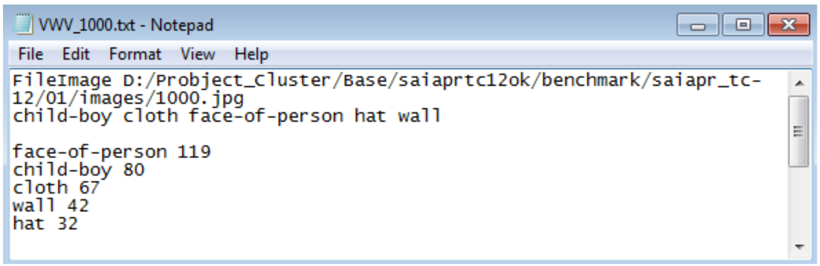


Figure 4. Illustration of a visual word vector

Figure 4 is an illustration of the visual word vector a set of similar images, which is generated from retrieval image process. This image is segmented into 5 regions with equivalent visual words for each region such as: child boy, cloth, wall, hat, face-of-person. The retrieval images process of 1000.jpg on C-Tree produces a set of similar images and visual word vectors. Visual word vector is stored in text files with 5 vocabularies, which have the most frequency in the set of similar images: face-of-person (119), child-boy (80), cloth (67), wall (42), hat (32).

4.3. Image retrieval on C-Tree

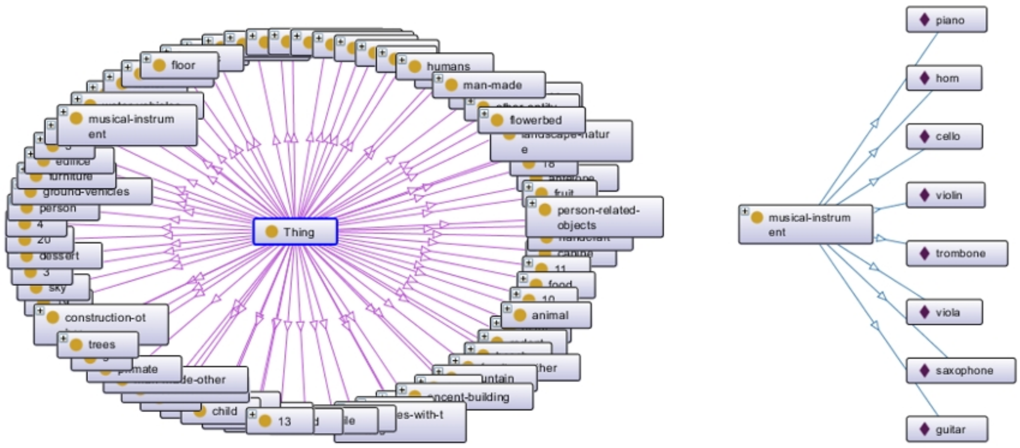
The query process is performed based on the regions of the query image to search for a set of similar images and visual word vector of the images. Retrieval image algorithm on C-Tree is described as follows.

Algorithm 4 IRCT

Input: feature vector f of query image I_Q, C-Tree
Output: Set of similar image SI
Function $IRCT(f, I_Q, v)$
Begin
  v=Root;
  If *(v is Leaf)* **then**
    $SI = v_i.E|i = 1..count$;
    Return SI;
  Else
    For $(f \in v)$ **do**
      $m = argmin\{Minkowski(f, v_i.f)|i = 1..v.count\}$;
    EndFor
    $v = v.E[m].l$;
    $IRCT(f, I_Q, v)$;
  EndIf
End.

4.4. Creating ontology of image dataset

An ontology is a collection of concepts and relations defined on these concepts, which represent the knowledge in a certain domain and provide reasoning and inference mechanisms [23]. In this paper, we propose the semantic based image retrieval using the low level feature of CBIR combined with semantic representations of ontology. The main purpose on ontology is to represent the image in semantics. We implemented mapping of content descriptions of images into semantics based on ontology.



(a) (b)

Figure 5. (a) Ontology of ImageCLEF created by Protg - (b) Ontology music-instrument

Ontology is described on Protg and an example for ontology of music-instrument in Figure 5. Based on the visual word vector, the SPARQL command is automatically generated for retrieval on ontology. The query result is a set of URIs with the image semantics and metadata of the similar image dataset. Figure 6 illustrates the query generated from the visual word vector.

```
PREFIX cla: <http://www.sgu.edu.vn/fit/imageclef/class#>
PREFIX cat: <http://www.sgu.edu.vn/fit/imageclef/catetories#>
PREFIX img: <http://www.sgu.edu.vn/fit/imageclef/imagename#>
PREFIX des: <http://www.sgu.edu.vn/fit/imageclef/description#>

SELECT DISTINCT *
WHERE
{
    {select * where { cla:child-boy img:imageID ?img.}}
    UNION
    {select * where { cla:face-of-person img:imageID ?img.}}
    UNION
    {select * where { cla:cloth img:imageID ?img.}}
    UNION
    {select * where { cla:wall img:imageID ?img.}}
    UNION
    {select * where { cla:hat img:imageID ?img.}}
}
```

Figure 6. Example of a query using SPARQL

5. EXPERIMENTS

5.1. Experimental application

To evaluate our approach, based on the proposed algorithms, we build the image retrieval system SBIR_CT to retrieve semantics of image dataset (Figure 7). Our proposal has been implemented and evaluated in order to measure the image retrieval effectiveness. We used the ImageCLEF dataset. This dataset consists of 20,000 annotated and segmented images collected from a wide variety of domains, such as sports and actions, people, animals, cities, landscapes, and so forth, and stores in 41 folders (from 0-th folder to 40-th folder). Besides, it provides category annotations generated from segmentation tasks with 276 concepts. Each region is assigned to a label, which is mapped with a semantic concept.

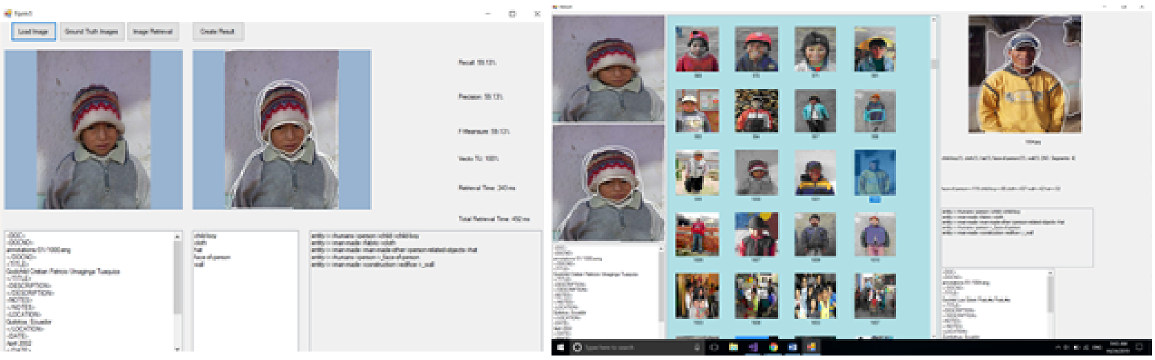


Figure 7. The SBIR_CT system for semantic retrieval image

In our experiment, the SBIR_CT system is built on the dotNET Framework 4.5 platform, the $C\#$ programming language. The graphs are built on MathLab. The SBIR_CT system is performed in two phases: preprocessing phase and query phase, which are implemented on computers with Intel (R) CoreTM i7-8750H processors, CPU 2.70GHz, RAM 8GB and Windows 10 Professional operating systems. Figure 7 describes the SBIR_CT system for semantic image retrieval.

5.2. Experimental results

In order to assess the effectiveness of proposed method, we used the following as evaluation metrics: precision, recall, F-measure. The formulas of these values are as follows:

$$\text{precision} = \frac{|\text{relevant images} \cap \text{retrieved images}|}{|\text{retrieved images}|}, \tag{1}$$

$$\text{recall} = \frac{|\text{relevant images} \cap \text{retrieved images}|}{|\text{relevant images}|}, \tag{2}$$

$$F\text{−measure} = 2 \times \frac{(\text{precision} \ \times \text{recall})}{(\text{precision} + \text{recall})}. \tag{3}$$

We obtained experimental results for image retrieval performance of the proposed method on ImageCLEF dataset in Table 1, which has 7092 query images; the averages of performance are: recall 0.4403, precision 0.6510, F-measure 0.5227, and average query time 73.0605 ms.

Table 1. Performance of image retrieval of the proposed method on ImageCLEF dataset

Folders	No. images	Avg. recall	Avg. precision	Avg. F-measure	Avg. query time (ms)
00-10	2239	0.412843042	0.63972223	0.49943441	82.2642317
11-20	1820	0.459227484	0.61276569	0.52322946	76.7232867
21-30	1491	0.412109099	0.63408214	0.49720632	73.5502254
31-40	1542	0.477112611	0.71750647	0.57088284	59.7042889
AVG	7092	0.440323059	0.65101913	0.52268826	73.0605082

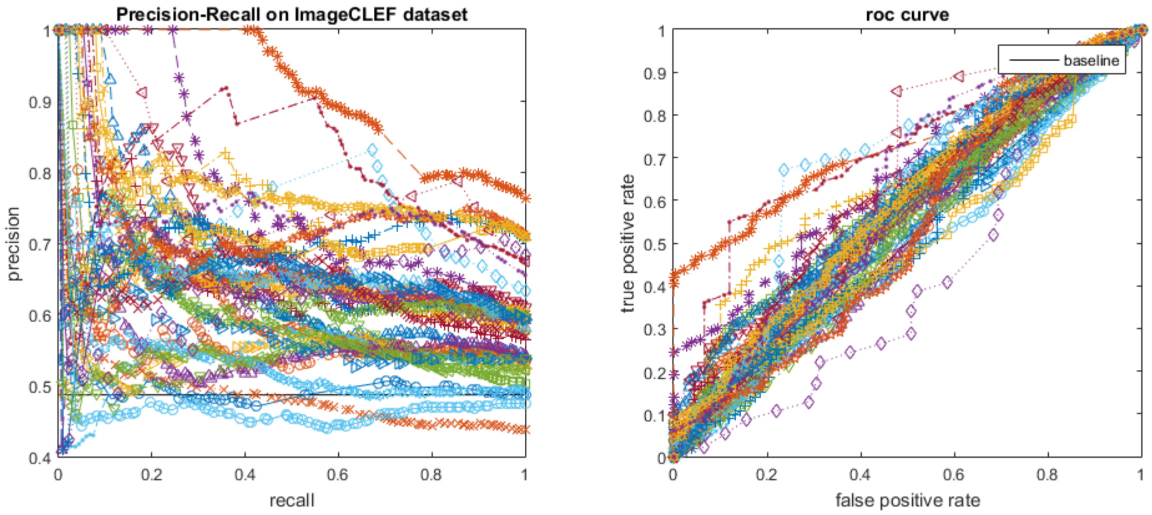


Figure 8. The graph of Precision-Recall and ROC of SIR-DL on ImageCLEF dataset

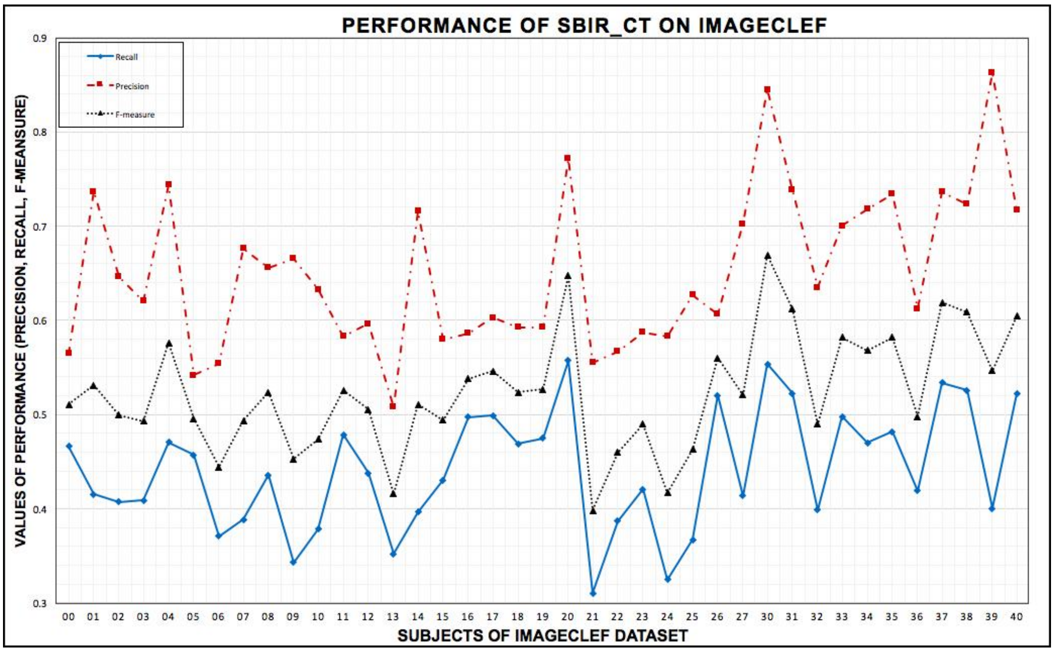


Figure 9. The mean averages of precision, recall and F-measure on the ImageCLEF dataset

Figure 8 shows the curves of Precision-Recall and ROC for the ImageCLEF dataset. Each curve describes a set of query images, which are retrieved. The graph shows that the area under the Precision-Recall curve is not high, because the accuracy of the query system is concentrated in the 0.4 to 0.7 range, but there are also image sets for the degree of accuracy within the high-performance areas [0.8, 1.0]. A receiver operating characteristic curve, or ROC curve, is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The diagonal divides the ROC space. Points above the diagonal represent good classification results; points below the line represent bad results. The ROC curve graph of our proposed system shows that more values fall within the true positive region than the false positive. Our proposed method is effective and potential to improve the performance of semantic-based image retrieval. This shows that the self-balanced clustering tree does well in data classification.

Figure 9 describes the mean average precision, recall, F-measure of 40 folders in ImageCLEF dataset. This graph shows that the precision of the retrieval is at an average level, with many subjects of image dataset for high precision. In particular, the precision of folder 39 is the largest at 0.8625. The precision of folder 13 is lowest at 0.5137. The precision of the SBIR_CT system is higher than the Recall, because the recall is quite low, the F-measure is not high. In image retrieval, recall is the fraction of the relevant images that are successfully retrieved. Therefore, the proposed method needs further improvement in the future to increase the recall of retrieval image.

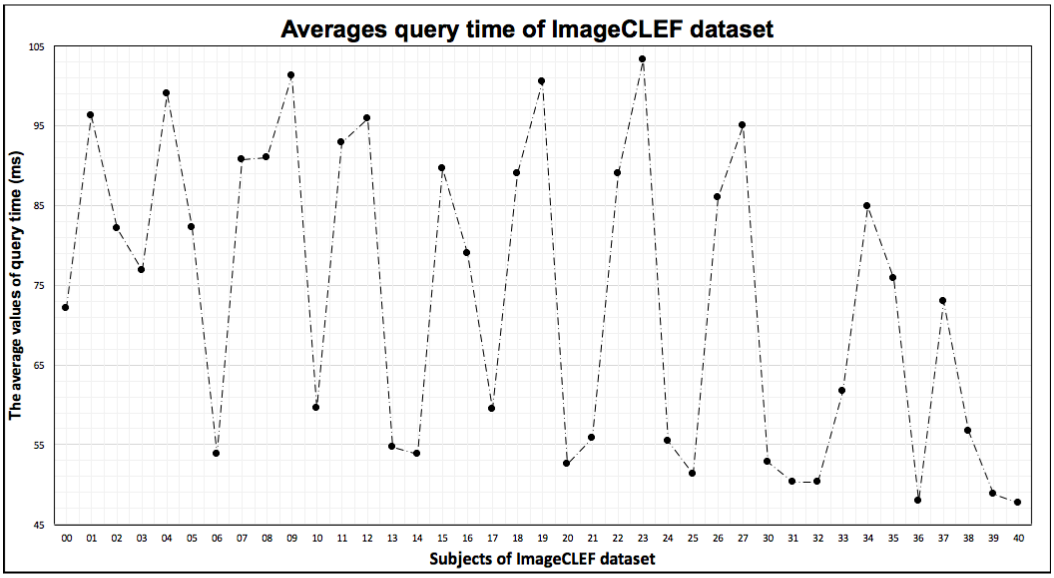


Figure 10. The average query time of subjects on the ImageCLEF dataset

In addition, Figure 10 shows the average query time of the ImageCLEF dataset. The average query time for each subset of images is low. The highest average query time is 102.8ms, and the lowest average query time is 47.62ms. This indicates that the semantic-based image retrieval on C-Tree is efficient in terms of time. The values of Mean Average Precision (MAP) of proposed method are compared with other methods on the same dataset. They are described in Table 2, which shows that the accuracy of SBIR_CT is higher than that of other methods.

Table 2. Comparison of mean average precision (MAP) of methods on ImageCLEF dataset

Methods	Mean Average Precision (MAP)
H. Cevikalp, 2017 [5]	0.4678
O. Allani, 2017 [2]	0.3460
M. Jiu, 2017 [13]	0.5970
Y. Cao, 2016 [4]	0.7236
SBIR_CT	0.6510

However, the MAP of Y.Caos method [4] is higher than that of the proposed method of this paper. In Y. Caos method, the authors perform image retrieval relied on CNN. In this method, two vectors are created including the image vector and the sentence vector. This system only searches for similar images and it does not create semantic of image content as well as does not query on ontology. So this method only performs the first stage of the semantic image retrieval. In our proposed method, we extracted semantics of image from low-level visual feature vectors based on C_Tree. This process creates a set of similar images with their semantics and visual word vector and query on ontology. Then we automatically create a query based on SPARQL language and query on ontology. We compared this work to show the difference between two problems, including the image retrieval based on semantic and the semantic-based image retrieval.

The comparison results show the accuracy and effectiveness of the proposed model and algorithm. Therefore SBIR_CT can be developed to improve the efficiency of semantic image retrieval systems.

6. CONCLUSIONS AND FUTURE WORKS

In this paper, we implement a semantic-based image retrieval system SBIR_CT based on self-balanced clustering C-Tree. The proposed model is based on semi-supervised learning techniques by combining the methods of hierarchical clustering and partitional clustering. At the same time, we developed a method for extracting semantic images on ontology. The retrieval process on C-Tree finds similar images and visual word vector; then the SPARQL command is automatically generated to query on ontology. The result of this process is a set of URIs, metadata and semantics of similar images. We implemented our SBIR_CT system based on the proposed methods, model and algorithms. The experiments are evaluated on ImageCLEF dataset with the precision at 65.10%, the recall at 44.59% and the F-measure at 49.73%. Experimental results are compared with other methods on the same image dataset. The experimental results show that proposed methods are correct and effective. Our proposal contributes to significantly increasing the relevance of retrieval results with semantic concepts and reducing “semantic gap”. SBIR_CT system can be developed and improved to increase image retrieval efficiency. In a future work, we intend to improve our algorithm image classification by using deep learning techniques and build ontology from image collections on WWW.

ACKNOWLEDGMENT

The authors would like to thank the Faculty of Information Technology, University of Science - Hue University for their professional advice for this study. We would also like to thank HCMC University of Food Industry, University of Education, and research group SBIR HCM, which are sponsors of this research. This work has been sponsored and funded by Ho Chi Minh City University of Food Industry. We also thank anonymous reviewers for helpful comments on this article.

REFERENCES

[1] O. Allani, N. Mellouli, H. Baazaoui, H. Akdag, Ben H. Ghezala, "A relevant visual feature selection approach for image retrieval", in *Proceedings of the 10th International Conference on Computer Vision Theory and Applications (VISAPP)*, 2015 (pages 377–384). Doi: 10.5220/0005306303770384

[2] O. Allani, H. B. Zghal, N. Mellouli, H. Akdag, "Pattern graph-based image retrieval system combining semantic and visual features", *Multimedia Tools and Applications*, vol. 76, no. 19, 2017.

[3] H. Bannour, C. Hudelot, "Building and using fuzzy multimedia ontologies for semantic image annotation," *Multimedia Tools Applications,* vol. 72, no. 3, pp. 2107–2141, 2014.

[4] Y. Cao, M. Long, "Deep visual-semantic hashing for cross-modal retrieval," *Inter. Conf. on Knowl. Discovery and Data Mining, SIGKDD*, California, USA: ACM, 2016 (pp. 1445–1454).

[5] H. Cevikalp, M. Elmas, S. Ozkan, "Large-scale image retrieval using transductive support vector machines," *Computer Vision and Image Understanding*, vol. 173, August 2018, pages 2–12.

[6] M. Crucianu, M. Ferecatu, N. Boujemaa , "Relevance feedback for image retrieval: a short survey", Report of the DELOS2 European Network of Excellence (FP6), 2014.

[7] B. Demir, L. Bruzzone, "A novel active learning method in relevance feedback for content-based remote sensing image retrieval", *IEEE Trans Geosci Remote Sens,* vol. 53, no. 5, pp. 2323–2334, 2015.

[8] J. Eakins, M. Graham, "Content-based image retrieval", Technical Report, University of Northumbria at Newcastle, 1999.

[9] H. J. Escalante, C. A. Hernndez, J. A. Gonzalez, A. Lpez-Lpez, M. Montes, E. F. Morales, L.E. Sucar, "The segmented and Annotated IAPR TC-12 Benchmark", *Computer Vision and Image Understanding*, vol. 114, no. 4, pages 419–428, April 2010.

[10] T. Gevers, A. Smeulders, "Content-based image retrieval by viewpoint-invariant color indexing", *Image Vision Computing,* vol. 17, no. 7, pp. 475–488, 1999.

[11] L. Gomez, Y. Patel, M. Rusiol, D. Karatzas, C. V. Jawahar, "Self-supervised learning of visual features through embedding images into text topic spaces", *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017 (pp. 4230–4239).

[12] M. Grubinger, "Analysis and evaluation of visual information systems performance", School of Computer Science and Mathematics, Faculty of Health, Engineering and Science, Victoria University, Melbourne, Australia, Tech. Rep., 2007.

[13] M. Jiu, H. Sahbi, "Nonlinear deep Kernel learning for image annotation", *IEEE Transactions on Image Processing,* vol. 26, no. 4, pp.1820–1832, 2017.

[14] D. Lin, "Automatic retrieval and clustering of similar words," Department of Computer Science University of Manitoba Winnipeg, Manitoba, Canada R3T 2N2, Tech. Rep.

[15] Y. Liu, D. Zhang, G. Lu, "Region-based image retrieval with high-level semantics using decision tree learning," *Pattern Recognition*, vol. 41, no. 8, pp. 2554–2570, 2008.

[16] B. Linse, F. Bry, D. Plexousakis, G. Gottlob, "RDF querying: Language constructs and evaluation methods compared," *Reasoning Web International Summer School,* Springer, Berlin, Heidelberg (pp. 1–52).

[17] P. Haase, J. Broekstra, A. Eberhart, R. Volz, "A comparison of RDF query languages," *International Semantic Web Conference,* Springer, Berlin, Heidelberg (pp. 502–517).

[18] V. Mezaris, I. Kompatsiaris, M.G. Strintzis, "Region-based image retrieval using an object ontology and relevance feedback", *EURASIP J. Adv. Signal Process.*, 2004. https://doi.org/10.1155/S1110865704401188

[19] R.I. Minu, K.K. Thyagharajan, "Multimodal ontology search for semantic image retrieval," *ICTACT Journal On Image And Video Processing,* vol. 3, no. 1, p. 473, 2012.

[20] J. Moehrmann, G. Heidemann, "Semi-automatic image annotation," *International Conference on Computer Analysis of Images and Patterns.* Springer, Berlin Heidelberg, 2012 (pp. 266273).

[21] S. Poslad, K. Kesorn, "A multi-modal incompleteness ontology model (MMIO) to enhance information fusion for image retrieval," *Information Fusion*, vol. 20, pp. 225–241, 2014.

[22] F. Pollinger, A. Bauch, K. Meiners-Hagen, M. Astrua, M. Zucco, S. Bergstrand, "Metrology for long distance surveying: A joint attempt to improve traceability of long distance measurements," *IAG 150 Years.* Springer, Cham, 2015 (pp. 651–656).

[23] S. Sarwar, Z. U. Qayyum, S. Majeed, "Ontology based image retrieval framework using qualitative semantic image descriptions," *Procedia Computer Science*, vol. 22, pp. 285–294, 2013.

[24] I.K. Sethi, I.L. Coman, "Mining association rules between low-level image features and high-level concepts," *Proceedings Volume 4384, Data Mining and Knowledge Discovery: Theory, Tools, and Technology III,* 2001. https://doi.org/10.1117/12.421083

[25] A. Singh, A. Yadav, A. Rana, "K-means with three different distance metrics," *International Journal of Computer Applications*, vol. 67, no. 10, 2013.

[26] A. Vailaya, M.A.T. Figueiredo, A.K. Jain , H.J. Zhang, "Image classification for content-based indexing," *IEEE Trans. Image Process.,* vol.10, no. 1, pp. 117–130, 2001.

[27] Z. Yang, C. C. Jay Kuo, "Learning image similarities and categories from content analysis and relevance feedback," *MULTIMEDIA '00: Proceedings of*

the 2000 ACM Workshops on Multimedia, November 2000 (pages 175–178). https://doi.org/10.1145/357744.357927

[28] X.S. Zhou, T.S. Huang, "CBIR: from low-level features to high- level semantics," *Proc. SPIE 3974, Image and Video Communications and Processing,* April 19, 2000 (pp. 426-431). https://doi.org/10.1117/12.382975

[29] M. Villegas, H. Mller, A. Gilbert, L. Piras, J. Wang, K. Mikolajczyk, B. Acar, "General overview of ImageCLEF at the CLEF 2015 labs," *International Conference of the Cross-Language Evaluation Forum for European Languages. CLEF 2015: Experimental IR Meets Multilinguality, Multimodality, and Interaction,* Springer, Cham., 2015 (pp. 444–461).

[30] B. Ionescu, H. Mller, M. Villegas, H. Arenas, G. Boato, D. T. Dang-Nguyen, B. Islam, "Overview of ImageCLEF 2017: Information extraction from images," *International Conference of the Cross-Language Evaluation Forum for European Languages.* Springer, Cham., 2017 (pp. 315–337).

[31] J. Prez, M. Arenas, C. Gutierrez, "Semantics and complexity of SPARQL," *International Semantic Web Conference.* Springer, Berlin, Heidelberg, 2006, November (pp. 30–43).

[32] B. Quilitz, U. Leser, "Querying distributed RDF data sources with SPARQL," *European Semantic Web Conference.* Springer, Berlin, Heidelberg, 2008, June (pp. 524–538).

Received on August 22, 2019
Revised on December 26, 2019