IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 24 September 2023, accepted 22 October 2023, date of publication 16 November 2023,
date of current version 18 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3333678

RESEARCH ARTICLE

Improving the Efficiency of Semantic Image Retrieval Using a Combined Graph and SOM Model

NGUYEN MINH HAI[1,2,3], TRAN VAN LANG[4], AND THANH THE VAN[5]

[1]Institute of Applied Mechanics and Informatics, Vietnam Academy of Science and Technology (VAST), Hanoi 10072, Vietnam
[2]Graduate University of Science and Technology, Vietnam Academy of Science and Technology (VAST), Hanoi 10072, Vietnam
[3]Faculty of Physics, HCMC University of Education, Ho Chi Minh City 72711, Vietnam
[4]Journal Editorial Department, Ho Chi Minh City University of Foreign Languages and Information Technology (HUFLIT), Ho Chi Minh City 72511, Vietnam
[5]Faculty of Information Technology, HCMC University of Education, Ho Chi Minh City 72711, Vietnam

Corresponding author: Tran Van Lang (langtv@huflit.edu.vn)

ABSTRACT Extracting similarity and semantic images from images is a hot topic that is used in various semantic retrieval systems. The GP-Tree, a hierarchical clustering tree, is used in the article to retrieve semantic images. The GP-Tree is then used to create a graph of neighboring leaf nodes, resulting in a smaller search list and a reduced chance of missing related similar images. In addition, the neighbor graph is used to build a Self-Organizing Map (SOM) to improve clustering efficiency and image retrieval performance. To improve high-level semantic retrieval efficacy, we also propose extending the current ontology structure. The ontology framework with rich domains represents most of the basic objects. With this ontology, the sets of images are added for the enrichment and can be used for many different sets of images. The SPARQL query is automatically generated from visual words for querying on the ontology. The query result on the ontology is a set of similar images and the definition of its semantics. The proposed method is tested on the WANG and ImageCLEF datasets, and the results are compared to previous publications on the same dataset, demonstrating its efficacy.

INDEX TERMS Semantic-based image retrieval, content-based image retrieval, hierarchical clustering, self-organizing map, ontology.

I. INTRODUCTION

Image data is becoming more popular because it is one of the most effective ways to express, share, and remember information. Furthermore, image databases on topics such as art, satellite imagery, tourism, biology, medicine, and so on are attracting a growing number of users, both professional and amateur. As a result, effective systems that can quickly retrieve relevant visual information from a large collection of images are in high demand. Content-based image retrieval (CBIR) systems [1], [2], [3], [4], which extract low-level image features (texture, color, shape, and so on) to describe their visual content, have piqued the interest of researchers worldwide. The retrieval process involves combining the visual features of a given query image with those of the image collection to produce visually similar results. Extensive experiments on CBIR systems, however, have shown that low-level content frequently fails to describe high-level semantic concepts in human thinking [5]. The "semantic gap" refers to the gap between the limited description of low-level image features and the richness of user semantics [6], [7].

Recent research has addressed the "semantic gap" problem by combining visual words of various types of descriptors [8], [9], [10], [11]. Other significant studies [12], [13] rely on incorporating spatial image attributes into image retrieval. The effectiveness of these techniques in improving the performance of CBIR systems by reducing the "semantic gap" problem has been demonstrated experimentally. However, CBIR systems frequently encounter time complexity, high computational costs, and a lack of user-friendly semantics. To address these limitations, researchers

The associate editor coordinating the review of this manuscript and approving it for publication was Wentao Fan.

have focused on Semantics-Based Image Retrieval (SBIR). Diversifying search results based on relevance levels is an effective solution in cases where the user intention and context are unknown. This solution entails retrieving images that cover as many relevant topics as a vague query may have. The results obtained can then be ranked [14]. Two post-processing strategies were used: clustering-based strategy and diversification-based strategy.

Based on semi-supervised learning techniques, we previously developed a multi-branch tree, GP-Tree, for clustering feature vectors to store automatically indexed images [14]. However, in GP-Tree, each node splitting can separate similar elements into different branches, rendering the search for the most similar nodes ineffective. As a result, improving the retrieval efficiency of GP-Tree is required. By finding all related clusters, searching on a graph can overcome the disadvantage of missing similar data on the tree. However, because the graph must track all clusters, this advantage causes the graph to consume a lot of memory. Thus, in our previous study [15], we proposed an improvement to GP-Tree by generating a graph from clusters of leaf nodes in GP-Tree using the nearest-neighbor clustering technique to provide a smaller search list and avoid missing related data. Searching on the graph, however, still relies on measurement, and when the amount of data is too large, image search may encounter inaccuracies, resulting in lower accuracy than expected. To improve the performance of finding similar image sets even further, we propose creating a Self-Organizing Map (SOM) [16] based on the nearest-neighbor graph to search for the winning clusters for more accurate image classification.

Based on an analysis of the strengths and weaknesses of existing methods and our previous research direction, this paper proposes an improved GP-Tree approach to improve the efficiency of semantic image retrieval. The primary contribution of our work is to improve the accuracy of extracting similar images from the input query image based on GP-Tree, thereby building a semantic image query system with the following process:

- Building neighbor cluster graph Graph-GPTree, which is a combination of cluster graph and GP-Tree, is proposed as follows: each leaf node on GP-Tree tree in the experimental process will find clusters neighbors, thereby forming a cluster graph related to each other in terms of measure and hierarchical relationship (parent-child). Neighbor cluster graphs allow faster search times and less memory overhead than traditional cluster graphs
- Building SgGP-Tree model which is a combination of GP-Tree, Graph-GPTree and SOM network to overcome cluster selection problems of Graph-GPTree graph; Since the clustering criterion of the Graph-GPTree graph is metric, it can lead to errors if the tree performs many node splitting and the number of layers is large, and the SOM network overcomes these problems of the cluster graph because winning cluster selection criteria.
- Improving high-level semantic retrieval efficiency by enriching the RDF triple language ontology framework [17] by adding the RDF triple semantic structure. This structure is an ontology framework based on the RDF triple language, which describes and interacts with semantic information in data. From there, develop an image retrieval model based on SgGP-Tree, which allows querying similar image sets and semantic classification of the queried images. This model automatically generates SPARQL queries from classifications and semantic queries. Through this process, the model provides semantically related sets of images, metadata, semantic annotation, and hierarchical concept classification. These contributions make advancements in information extraction from graph data, improve semantic retrieval performance, and provide an automated way to search and discover information in the field of image retrieval.

The following is the rest of the article: Section II presents literature review; Section III presents image retrieval based on the combination of graph and SOM network; Section IV presents the semantic image retrieval model; Section V discusses an application of the SBIR-GP retrieval system based on the proposed model, the execution of experiments on popular image datasets to compare with other methods, and Section VI concludes and future development directions.

II. LITERATURE REVIEW

Many research groups have been formed in recent years with the goal of improving the effectiveness of semantic image retrieval based on built ontologies [17], [18], [19], [20], image retrieval based on relevance feedback techniques [21], and ontology-based image retrieval applied to text retrieval, multimedia data, or determining relationships between images through image annotations and features [22], [23]. However, due to the difference between computational representation in machines and natural language in humans, the obtained similar image set does not fully meet the expectations of users. Many related research works have been published with the goal of reducing semantic gaps to improve image retrieval performance, such as:

A paper on semantic image retrieval was introduced by Hirwane [21]. To build a semantic query model for images, the author introduced techniques for relevance feedback, classification, and semantic similarity measures. The author only used data mining techniques in this work, not search models, to improve the effectiveness of semantic image retrieval. Spanier et al. [24] created a multi-method ontology called MMO (Multi-Modality Ontology) to reduce the semantic gap between images using the OPF (Object Properties Filter) attribute filter. However, the author group only constructed an ontology based on a small sample dataset and a specific image domain, without creating a structure to store image datasets. The SemVisIR image retrieval system, proposed by Allani et al. [18], combines low-level image features and high-level semantics. Using clustering algorithms, the image dataset is stored in an automatically generated pattern graph. SemVisIR has modeled the visual aspects of images using region graphs and assigned them to ontology modules that were generated automatically.

Bchir et al. [25] used feature vectors extracted from object regions to perform segmentation for faster image search. The authors used this method to create a semantic mapping between visual features and high-level semantics. Jabeen et al. [19] created an image search model by clustering visual features and combining image classifier semantics. However, clustering low-level visual features may result in image clusters with different semantics, resulting in skewed search results for the query semantics of the image. As a result, a semantic classification method based on low-level features should be used, and these features should be transformed into image semantics.

Mafla et al. [26] proposed a method that combines visual and locally synthesized material features for detailed image classification and retrieval. This approach has the advantage of leveraging textual information to extract meaningful information from images. By mining text cues, it is possible to develop more comprehensive computer vision models that can better understand the context. The researchers tested their model on two sets of images from the Con-Text Dataset: the Drink Bottle Dataset, achieving accuracies of 64.52% and 62.91% respectively.

In another study, Wang et al. [27] presented an integrated ontology framework for remote sensing images. This ontology is an extension of the semantic sensor network ontology (SSN) using the OWL language. However, in applications involving multi-source data, there are often significant semantic challenges.

In addressing the problem of retrieving the bioCADDIE 2016 biomedical image dataset, Xu et al. [28] proposed a semantic similarity approach based on Ontology. The authors used the MeSH method to extract concepts from the bioCADDIE image set. To retrieve a similar set of images, they employed two measures, Wu-Palmer and Resnik, to quantify the semantic similarity between concepts.

Yu [20] proposed a semantic text processing and retrieval ontology model. The process of creating a semantic ontology for information retrieval entails entering query information, sending it to the ontology to find the corresponding semantic concept, and returning the query results to the user. The authors conducted experiments by extracting word concepts from 1000 scientific papers to generate ontology concepts and literals in ten groups of 100 papers each containing query terms or keywords. They also proposed a genetic algorithm that used word frequency calculations to return search results. The experiment demonstrated that the proposed model's information retrieval performance was feasible. The study, however, did not apply to image search problems, did not propose a model for automatically or semi-automatically building ontologies to enrich ontology data, and did not perform flexible queries to meet user needs.

Zhong et al. [23] proposed using image annotations and features to determine the relationship between images. By classifying image objects, attributes, and determining the relationship between image classes and object classes, the authors created an ontology framework to access the relationship of images. The authors introduced the HowNet structure in this paper and expanded on it by combining classification principles to build relationships between image objects. An ontology framework for processing semantic image relationships was created based on the semantic model. However, this is only the first step in developing an ontology application for images while automatically integrating HowNet into ontology-based semantics.

A multimodal feature-based image retrieval model was proposed by Pustu-Iren et al. [29]. The image-text relationship is determined by scene text detection, which maps image numbers to the corresponding text. Using deep neural networks, the method successfully embedded shape and structural information into the image. However, because the input image has not yet been partitioned or clustered by machine learning methods, the time to query a similar image to the input image is not yet optimal.

Hu et al. [30] proposed a semantic image retrieval model using an interest selection-based image classification method. In the experiment, the proposed method focuses on explaining the weighted eigenvectors of interest points and performing related mouse-click tests to recognize the classification of scene objects. The experimental results revealed that the object's first and second interest selections have a significant impact on target classification in the experimental context; the IWS-SVM method has the best overall effectiveness in classifying target objects in four types of experimental scenes; and the interest point method can improve the effectiveness of image information retrieval. However, because the proposed model does not yet include image dataset clustering, retrieval of similar image datasets with the same semantics does not achieve high performance. Shi et al. [31] proposed a query-based synthesis model guided by retrieval. Although the proposed method synthesizes realistic images and outperforms existing methods, inference speed remains a limitation, and image retrieval speed is time-consuming, making real-time inference impossible.

Recent approaches have focused on mapping low-level features to semantic concepts using supervised or unsupervised machine learning techniques; building data models such as graphs, trees, or deep neural networks to store low-level image content; and developing ontologies to identify high-level concepts, among other things. The SBIR problem, on the other hand, is heavily reliant on reliable external resources such as automatically captioned images, ontologies, and training datasets. Hai et al. [14] also developed a semi-supervised learning-based method for storing images that are automatically indexed based on low-level image features. In this paper, a GP-Tree was built, with each node clustered based on similarity measures using hierarchical clustering to efficiently search for a set of similar images and classify the input query image, semantically querying images based on ontology. The GP-Tree is a multi-branch tree that clusters feature vectors, stores large amounts of data, and has a fast image retrieval speed. However, GP-Tree may split similar elements into separate branches each time a node is split, so the process of searching for the most similar branch will not find those similar elements that have been moved to a different branch. As a result, improving retrieval efficiency on the GP-Tree is required.

III. IMAGE RETRIEVAL BASED ON GRAPH AND SOM

GP-Tree is a multi-branch tree that clusters feature vectors to store low-level features of images and enable fast image retrieval [14]. However, each time a node is split, GP-Tree may separate similar elements into separate branches, making it difficult to search for the most similar elements that have been moved to different branches. Therefore, the retrieval performance is not optimal, and improving the retrieval efficiency on GP-Tree is necessary. To address this issue and enhance the retrieval efficiency on GP-Tree, we propose the following improvements:

(1) Creating a neighbor cluster graph called Graph-GPTree: This graph is constructed from the neighbors of leaf nodes in the GP-Tree. When a node is split, the neighbors of the new leaf nodes are marked using a predefined criterion. As a result, searching the neighbor cluster graph will avoid missing similar data elements as in GP-Tree, improving image retrieval accuracy. Additionally, the weight vector set is trained during the process of splitting nodes to create the neighbor cluster graph.

(2) SgGP-Tree model: This is a self-organizing map (SOM) network that is assembled from the Graph-GPTree neighbor cluster graph based on the weight vector set trained on GP-Tree. The model aims to find the best clusters based on the proposed representative class and to improve retrieval efficiency.

In previous studies, we have built the GP-Tree [14] and the Graph-GPTree [15], a graph-based neighbor clustering method. Experimental results on the Graph-GPTree showed superior accuracy compared to the GP-Tree, demonstrating the effectiveness of our proposed improvement. The Graph-GPTree has solved most of the issues of the GP-Tree, improving the image retrieval performance. However, the criteria for selecting clusters on the graph is based on distance, which may lead to measurement errors when the tree splits a node multiple times. This is because when splitting a leaf, the two new leaves may not generate neighbors, but the representative elements of the split leaf (most elements) have been allocated to the two new leaves. Therefore, additional criteria are needed to select the winning leaf based on the weight of the representative elements of that leaf. As a result, the SOM network was built on both the GP-Tree and the Graph-GPTree, called the SgGP-Tree, to form a hybrid tree-graph-SOM model. In this section, we present an overview of the hierarchical clustering tree structures of the GP-Tree and the Graph-GPTree that we have previously studied, followed by a description of the structure of the SgGP-Tree hybrid model.

A. GP-TREE

The GP-Tree is a multi-branching hierarchical clustering tree [14] consisting of a root node, internal nodes, and leaf nodes. Each leaf in the GP-Tree contains a cluster of similar images. The image retrieval process is performed by traversing the tree from the root and selecting a branch if the representative element of that branch has the closest similarity score to the query image. This process is then repeated with the next child node if it is not a leaf; otherwise, the leaf contains a set of images that are similar to the query image.

Based on the analyzed images, the storage location of the images (URL) is allocated to the leaf nodes of the GP-Tree organized as follows.

Definition 1: Data element

The data element ρ at a leaf node is a tuple (f, τ, μ), denoted as $\rho = (f, \tau, \mu)$, where $f = (f_1, f_2, \ldots, f_n), f_i \in [0, 1], \forall i = \overline{1, n}$ is the feature vector of the image; τ is the path to the storage file on the disk (URL), and μ is the class of the image.

Definition 2: Representative element

The representative element σ in an internal node (including the root node) is a pair (c, l), denoted as $\sigma = (c, l)$. Here, $l = (l_1, l_2, \ldots, l_k)$ consists of k links to k nodes that are connected to this internal node.$c = (c_1, c_2, \ldots, c_n)$ is the center value corresponding to n features, where each c_i is the average of k center values of k connected nodes.

Definition 3: GP-Tree

The GP-Tree consists of:

- *A root node is a set C_0 with n_0 representative elements as defined in **Definition 2**, where $C_0 = \{\sigma_i^0 = (c_i^0, l_i^0)/\forall i = \overline{1, n_0}\}$.*
- *A set T consisting of N_T internal nodes, each of which is a set C_k consisting of n_k representative elements, denoted by $T = \{C_k = \{\sigma_i^k = (c_i^k, l_i^k)/\forall i = \overline{1, n_k}\}/\forall k = \overline{1, N_T}\}$*
- *A set L consisting of N_L leaf nodes, each of which is a set L_l consisting of m_l data elements as defined in **Definition 1**, denoted by $L = \{L_l = \{\rho_i^l = (f_i^l, \tau_i^l, \mu_i^l)/\forall i = \overline{1, m_l}\}/\forall l = \overline{1, N_L}\}$*

Comment: The number of images stored in a GP-Tree as defined in **Definition 3** is $\sum_{l=1}^{N_L} n_l$.

Definition 4: Selecting a data element's branch.

At each node σ in any given tree, the data element ρ selects the nearest branch σ_m^k based on similarity measure:

$$\sigma_m^k = argmin \left\{ \left\| \rho, \sigma_i^k \right\|_2, \forall i = \overline{1, n_k} \right\}$$

Theorem 1: There exists a unique path from the root to a leaf to insert the data element ρ into the leaf.

Proof: Let η be any GP-Tree node and let ρ be an element to be added to the GP-Tree. If $\eta \in T$, the element ρ can always choose a sub-branch to create a path to a leaf node,

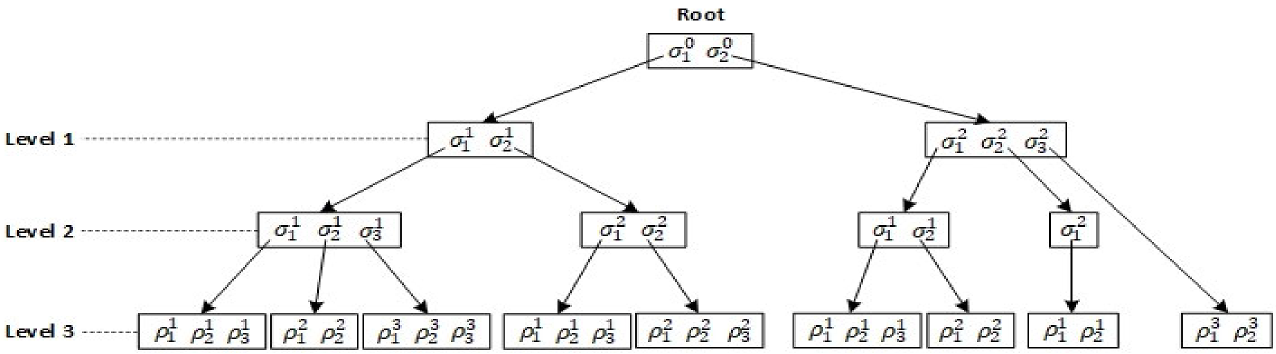


FIGURE 1. Illustrates the three levels of the GP-Tree.

according to **Definition 4**. Continue searching for the next sub-branch if the sub-branch is an internal node, and if $\eta \in L$, then the leaf node L_l containing the element ρ has been found. As a result, there is always a leaf node containing the element ρ.

According to **Definition 3**, there is always a C k to connect every leaf node L lL to lk. Assume there are two distinct leaf nodes $\mathrm{L_u}, \mathrm{L_v} \in \mathrm{L}$ such that $\rho \in \mathrm{L_u} \wedge \rho \in \mathrm{L_v}$. That is, there are two ways to get from $\mathrm{C_k}$ to the leaf node that contains the element ρ. Only one sub-branch can be chosen to go to the next child node, according to **Definition 4**. As a result, the preceding assumption is illogical. As a result, $\mathrm{L_u} \equiv \mathrm{L_v}$. As a result, the element ρ is stored in only one leaf node on the tree.

Fig. 1 shows a three-level hierarchical clustering tree known as the GP-Tree. According to **Definition 3**, GP-Tree is a growing tree that is constructed by adding an element ρ to a leaf node depending on a threshold θ and a distance d between the element to be added and the representative element, in the following cases:

(1) If $d \leq \theta$, then ρ belongs to the current leaf node.
(2) If $d > \theta$, then a new leaf node is created at the current parent node and ρ is added to the newly created leaf node.

Based on **Definition 3**, if the number of elements in any leaf node L_l is greater than the maximum number of elements M in a leaf node, this leaf node is split into two leaf nodes, creating a parent node linked to these two leaf nodes and the parent node becomes a child node of the current parent node; then the ρ_i^l elements are distributed to the two newly created leaf nodes. The process of splitting a leaf node into two new leaf nodes is as follows:

- Let L_l and L_r be the two new leaf nodes after splitting the leaf node L_s. Determine the center of the leaf node L_s as the average value of the feature vectors in L_s.
- Choose an element ρ_i that is far from the center of the leaf node L_s as the center of the leaf node L_l. Next, choose an element ρ_j, the farthest element from ρ_i, as the center of the leaf node L_r. Then, create a new parent node (an internal node) of L_l, L_r and add two center elements ρ_i and ρ_j to this new parent node.
- The elements in the leaf node L_s are allocated to the two new leaf nodes based on the nearest node selection rule using the Euclidean distance. Update the center element at the parent node cluster and recursively perform to the root.

Based on **Definition 3** and **Theorem 1**, GP-Tree is a multi-branched tree structure and grows in the direction of leaf. The GP-Tree is created by adding each data element to the structure of the tree. The added element only chooses a single direction on the tree to determine the leaf node to store it; therefore, if moving from the root node to the leaf node, only one suitable leaf node is chosen for storage. This adding process will perform node splitting and the tree will grow to contain the initial dataset.

B. GRAPH-GPTREE

To improve query efficiency on the GP-Tree, we propose an enhancement to the GP-Tree with the neighbor cluster graph, called Graph-GPTree. The Graph-GPTree is built based on the set of leaf nodes of the GP-Tree [15]. Each time a leaf node is split, the system marks the corresponding neighbor levels for the newly split leaf nodes. Thus, the structure of the Graph-GPTree still follows the rules of the GP-Tree, and the clustering problem on the tree is simplified into a graph clustering problem whose main task is to link related clusters, thereby minimizing the omission of similar elements when splitting nodes. **Fig. 2** illustrates the structure of the Graph-GPTree.

The Graph-GPTree $G = (V, E)$ is an undirected graph where the vertex set V consists of clusters of leaf nodes in the GP-Tree, and the edge set $E \subseteq V \times V$ consists of links between pairs of leaf nodes at different neighbor levels. The neighbor levels between any two leaf nodes L_i and L_j are defined as follows:

Definition 5: Neighbor levels

- *1st-level neighbor: Let* $v_p = \left(v_1^p, v_2^p, \ldots, v_n^p\right), v_q = \left(v_1^q, v_2^q, \ldots, v_n^q\right)$ *be the center vectors of two leaf nodes* L_p *and* L_q, *where* $v_j^p = \sum_{i=1}^{m_p} f_{ij}^p, \forall j = \overline{1, n}$; $v_j^p = \sum_{i=1}^{m_p} f_{ij}^p, \forall j = \overline{1, n}$. *If* $\left\| v_p, v_q \right\|_2 < \theta$, *where* θ *is*

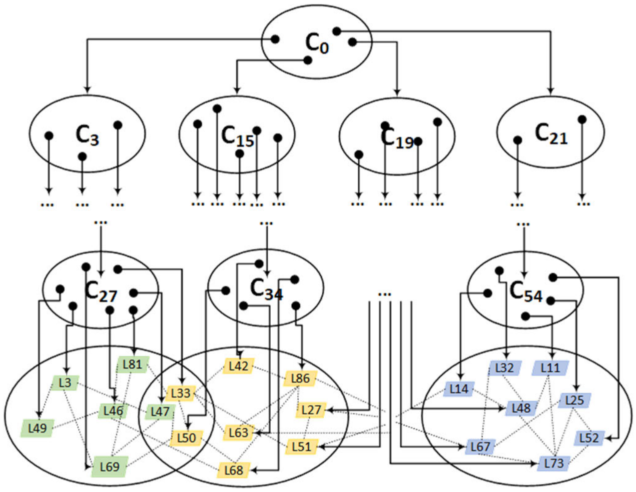


FIGURE 2. Illustrates the structure of the Graph-GPTree.

a predetermined threshold value, then L_p and L_q are marked as 1st-level neighbor with each other.

- *2nd-level neighbor: Let r and s be the number of class labels of images appearing in two leaf nodes L_t and L_k; c_t and c_k are the class labels that appear the most in the respective leaf nodes, where $c_t = argmax\{count(\eta_i.c_j)|\eta_i \in L_t, i = 1..|L_t|, j = 1..r\}$, $c_k = argmax\{count(\eta_i.c_j)|\eta_i \in L_k, i = 1..|L_k|, j = 1..s\}$. If $c_t \equiv c_k$, then L_t and L_k are marked as 2nd-level neighbor with each other.*

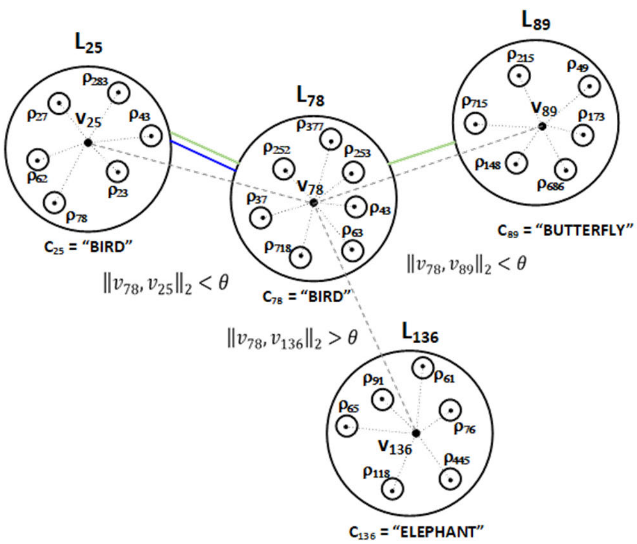


FIGURE 3. Describe an example of the neighbor levels of a leaf node in the neighbor cluster graph.

Fig. 3 is an example of the neighbor cluster graph as defined in **Definition 5** from node L_{78}. The representative class of this leaf node is $\mu_{78} =$ "*BIRD*". The neighbor levels are represented by the connecting edges, with 1st-level edges shown as solid green lines and 2nd-level edges shown as blue lines. Thus, we have:

- Since $\|v_{78}, v_{25}\|_2 < \theta$, L_{78} is a 1st-level neighbor of L_{25}. Moreover, the representative class of L_{25} is $\mu_{25} =$ "*BIRD*", so $\mu_{25} \equiv C_{78}$, and hence L_{25} is also a 2nd-level neighbor of L_{78};
- Since $\|v_{78}, v_{89}\|_2 < \theta$, L_{89} is a 1st-level neighbor of L_{78};

Since $\|v_{78}, v_{136}\|_2 > \theta$ and the representative class of L_{136} is $\mu_{136} =$ "*ELEPHANT*", L_{136} is not a neighbor of L_{78}.

The image retrieval on the Graph-GPTree is performed as follows: first, the query image is feature-extracted and searched on the GP-Tree to find the branch with the closest similarity measure and the most appropriate leaf node. This step helps to limit the search space on the Graph-GPTree, saving memory space during the search process. Then, the search is performed on the neighbor leaf nodes on the Graph-GPTree to restrict the omission of distributed data in the leaf clusters that do not belong to the found branch, thereby increasing the accuracy of the image retrieval.

Theorem 2: The algorithm for splitting a leaf node in a GP-Tree and creating the Graph-GPTree has a time complexity of $O(m)$.

Proof: The *foreach* loop, which is the step for distributing the elements in the leaf node L_s to the two new leaf nodes, has the most time complexity in the algorithm. Because a leaf node can have a maximum of M elements, the loop will run M times to add the data elements to the new leaf nodes, with an execution time of M. As a result, the algorithm's time complexity is $O(m)$.

Based on **Definition 5** and **Theorem 2**, the Graph-GPTree neighbor graph is created to connect leaf nodes with similar elements according to predefined criteria, thereby enhancing the efficiency of retrieving similar image sets with the input image and improving image retrieval on the GP-Tree.

C. THE SgGP-TREE COMBINED NETWORK MODEL

SgGP-Tree is a hybrid of the GP-Tree, the Graph-GPTree neighbor graph, and SOM. Adjusting the weights in the SOM network during training improves the SOM cluster [16]. However, for large input image datasets, the weight adjustment process is expensive, and randomly initializing weights can result in completely different maps. Furthermore, because SOM is static, adding new data to the map after SOM training will misclassify input data; at this point, SOM must be trained from scratch.

To overcome the limitations of SOM, the grSOM is proposed, which is built from the Graph-GPTree graph. The grSOM network is built from clusters of Graph-GPTree leaf nodes and input weight vectors extracted during GP-Tree training, and it has the following advantages: (1) The GP-Tree is used to train a set of input weight vectors for grSOM. Because grSOM has stable weights, high accuracy, and weights that are not adjusted too much during training, it has a shorter training time than traditional SOM networks. (2) The grSOM network is more adaptable and can be expanded after training. The grSOM network is constructed from individual clusters of Graph-GPTree graph leaf nodes, so if a new leaf

Algorithm for Separating Leaf Nodes in the GP-Tree and Creating the Graph-GPTree

Input: Threshold θ, Leaf node L_s, Graph-GPTree;
Output: Graph-GPTree
Begin
 # Find the two furthest elements in a leaf node
 $c_s = \frac{1}{m_s}\sum_{i=1}^{m_s} \rho_i^s.f_i^s$;
 $\rho_i^l = \text{argmax}\left\{\left\|c_s, \rho_i^s.f\right\|_2, i = 1..m_k\right\}$;
 $\rho_j^r = \text{argmax}\left\{\left\|\rho_i^l.f, \rho_i^s.f\right\|_2, , i = 1..m_k\right\}$;
 # Create new two leaf nodes
 $L_l, L_r \longleftarrow$ initializetwonewleafnodes
 $L_l = \{L_l\} \cup \rho_i^l$; $L_r = \{L_r\} \cup \rho_j^r$;
 # Allocates elements to two new leaf nodes
 For $\rho_i^s \in L_s$ **do**
  If $\left\|\rho_i^s, \rho_i^l\right\|_2 < \left\|\rho_i^s, \rho_j^r\right\|_2$ **then**
   $L_l = \{L_l\} \cup \rho_i^s$;
  Else
   $L_r = \{L_r\} \cup \rho_i^s$;
  Endif
 EndFor
 # Create center elements for two nodes: L_l & L_r
 $\sigma_l^h.c^h = \frac{1}{m_l}\sum_{i=1}^{m_l} \rho_i^l.f_i^l$; $\sigma_r^h.c^h = \frac{1}{m_r}\sum_{i=1}^{m_r} \rho_i^r.f_i^r$;
 # Update presentation elements to parent
 $\sigma_h^k = \sigma_h^k \cup \left\{\sigma_l^h, \sigma_r^h\right\}$;
 # Determine the 1st-level neighbors of the two newly split leaf nodes
 If $\left\|\sigma_l^h.c^h, \sigma_r^h.c^h\right\|_2 < \theta$ **then**
  $\Psi_1.L_l = \Psi_1.L_l \cup \{L_r\}$; $\Psi_1.L_r = \Psi_1.L_r \cup \{L_l\}$;
 Endif
 # Determine the 2nd-level neighbors of the two newly split leaf node
 $\gamma_l = \text{argmax}\left\{\text{count}\left(\rho_i^l.\mu_i^l\right), i = 1..|m_l|\right\}$
 $\gamma_r = \text{argmax}\left\{\text{count}\left(\rho_j^r.\mu_j^r\right), j = 1..|m_r|\right\}$
 if $\gamma_l = \gamma_r$ **then**
  $\Psi_2.L_l = \Psi_2.L_l \cup \{L_r\}$; $\Psi_2.L_r = \Psi_2.L_r \cup \{L_l\}$;
 Endif
 $\text{Graph} - \text{GPTree} = \text{Graph} - \text{GPTree} \cup \{\Psi_1, \Psi_2\}$;
 Return Graph-GPTree;
End

node appears, it will be trained on the tree with its own weights rather than training the entire network from scratch.

The combined model of the GP-Tree, Graph-GPTree, and grSOM network is called the SgGP-Tree and is illustrated in **Fig. 4**. On the basis of the SOM network, the grSOM network structure is defined as follows:

Definition 6: grSOM Network

The grSOM network is a SOM network with inputs being feature vectors of images $f = (f_1, f_2, \ldots f_m)$, where each vector f_i has n dimensions $f_i = (v_1, v_2, \ldots, v_n), f_i \in \{0, 1\}$, and the output layer consists of neurons containing the set of leaf nodes of the GP-Tree. The input and output layers are fully connected by weight vectors $W_i = (w_1, w_2, \ldots w_n)$, $w_i \in \{0, 1\}$.

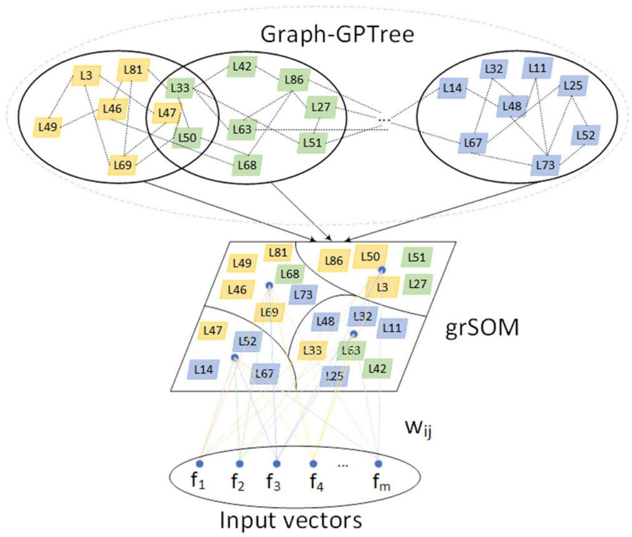


FIGURE 4. Illustrates the SgGP-Tree combined network model.

The purpose of SgGP-Tree network is to classify input data. The SgGP-Tree training process is a weight-training process. As previously discussed, instead of selecting a weight at random, a set of weight vectors trained on the GP-Tree is used. The following is the definition of this weight vector:

Definition 7: Weight vector

Let w be the weight vector of the data elements ρ at the leaf node. The weight vector w is the center of the feature vectors of the most frequently appearing classes in the leaf node, and it is defined as follows:

$$w = \frac{\sum_{i=1}^{n} f_i}{n}$$

where f_i is the feature vector value of n most frequently appearing classes.

The training process for the grSOM network, as defined in **Definition 6**, consists of the following steps: (1) Allocating leaf nodes from the Graph-GPTree network into the SgGP-Tree network; (2) Initializing the initial weights w_i from the weight set obtained during the GP-Tree training process; (3) Randomly selecting a feature vector f_i as a training sample; (4) Finding the winning neuron using the *sigmoid* function; and (5) Updating the weights using the Gradient descent method. (6) Continue from step 3 until the training process is finished.

The trained weight vector is used to select the winning cluster, which is the best cluster found on the grSOM network. The winning cluster is defined as follows:

Definition 8: Winning Cluster

*Let f_k be the input feature vector of the grSOM network, and let the leaf nodes L_i, L_j have weight vectors as defined in **Definition 7**, denoted as W_i and W_j, respectively. If $sigmoid(\|f_k, W_i\|_2) < sigmoid(\|f_k, W_j\|_2)$, then L_i is the winning cluster. The winning cluster L_i directly connects to the winning weight vector W_i.*

The final weight vector set used is the best weight set determined from each training cycle. The weight update condition is performed according to the formula:

$$d^{(t+1)}(w, f_i) \leq d^t(w, f_i)$$

The weights are updated using the Gradient Descent method according to the formula::

$$w_j^{k+1} = w_j^k - \lambda \frac{\partial E}{\partial w_j}$$

where λ is the learning rate, $\lambda \in (0, 1)$; $E = \frac{1}{2}\sum_{i=1}^{N} \left(t_i - f\left(\sum_j w_j \cdot sigmoid\left(\|W_i, W_j\|_2\right)\right)\right)^2$ is the error function, and t_i is the expected value for the node with the i-th output. The threshold θ is adjusted to a small enough value at the end of the training process to update the winning weight vectors and their nearest neighbors.

The purpose of the grSOM network is to classify input data. The grSOM network is trained using competitive learning to select a winning cluster for representative classification, as defined in **Definition 8**. As a result, the system employs the *Sigmoid* function as an activation function in order to identify the winning cluster. The formula for the sigmoid function is:

$$f(x) = \frac{1}{1 + e^{-x}}$$

The sigmoid function was proposed because it accepts real values in the range [0, 1], which is appropriate for the grSOM network where the desired output values are in the [0, 1] range.

IV. THE SEMANTIC IMAGE RETRIEVAL MODEL

An ontology framework for a set of images has been built [32] based on the RDF/XML triple language to enable semantic image retrieval. Using an ImageCLEF [33] image dataset, the proposed image ontology framework is a rich image domain capable of extending domains, inheriting individual image ontology systems, and best describing image relationships with their classification.

A. ONTOLOGY FRAMEWORK FOR IMAGE DATASET

The ontology framework includes components such as: (1) Classes and class hierarchy inherited and updated from the ImageCLEF image dataset; (2) Attributes and relationships between classes and between individuals with classes; (3) Definitions of classes, attributes, and relationships; (4) Individuals are images from the ImageCLEF image dataset; (6) Ontology synonym dictionary.

Fig. 5 is an example of an ontology visualized in Protégé, including classes, class hierarchy, individuals, and relationships between individuals and classes. Each image represents an instance of one or more classes in the ontology.

B. ENRICHING ONTOLOGY FRAMEWORK

Enriching an ontology entails adding data, semantic descriptions, and expanding the ontology's structure to include more information and richer semantics. Enriching the image ontology framework involves adding classes, class hierarchy,

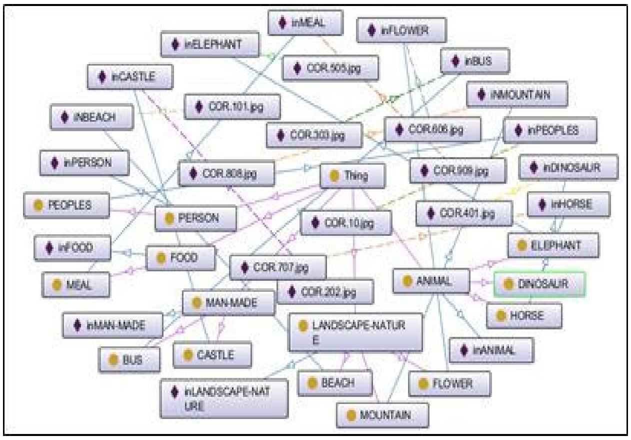


FIGURE 5. Illustrates an example of a visual ontology in Protégé.

properties, relationships, individuals, and semantic descriptions from various image datasets. Image data, on the other hand, is massive and constantly updated in order to improve and enrich the image ontology. As a result, the Protégé ontology cannot manage such large knowledge bases efficiently. As a result, it is proposed to enrich the image ontology semi-automatically, on a regular basis, with the capability of domain extension, inheriting existing ontology systems, or merging ontologies.

Sample data for classes, class hierarchy, properties, literals for class individuals and image individuals, and definitions for classes are prepared to semi-automatically enrich an ontology. These sample data are gathered automatically from various sources, including image datasets and the World Wide Web. The data is then edited and updated by experts in relevant fields before the ontology is generated automatically. RDF/OWL data models are created using the RDF and OWL libraries. Notation 3 syntax is used to store the ontology (N3). N3 is a non-XML serial string of RDF models that is intended to be more compact and readable than XML RDF [34]. The RDF semantic data model is supported by the N3 data model.

The addition of data to an ontology enriches its semantic descriptions and expands its structure. The initial ontology framework is automatically supplemented with classes, class hierarchies, attributes, relationships, entities, and semantic descriptions of a given set of images. **Fig. 6** depicts the model for data augmentation in the ontology framework, the process of ontology enrichment, which includes:

- **Step 1:** Images are classified based on the SgGP-Tree model (3) from experimental image sets (1) or any images from WWW (2).
- **Step 2:** The classifications are checked for duplicates with classes, image entities, attributes, and so on from the ontology framework (4), and then combined with semantic concepts from WORDNET (5) to enrich data samples of the ontology framework (6).
- **Step 3:** To further enrich the semantics, new classes, image entities, and attributes (7) are added to the ontology framework (8).

In **Step 2**, the ontology data is used to check for duplicates for a given classification. If the image class already exists, only the image instances are added to the ontology to enrich it. If the image class does not already exist, experts are brought in to determine the hierarchical relationship for the new classification. If the new class has a relationship with an existing class in the ontology, the parent-child hierarchy is checked. The new class is then added to the existing class as a parent or child. If no such relationships exist, a new hierarchy is created for this class.

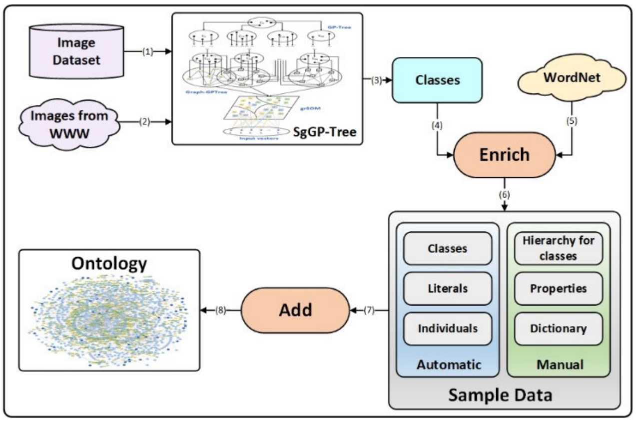


FIGURE 6. Ontology framework enrichment model.

Given an ontology framework $O_G = \{C^G, H_C^G, I^G, R^G, P^G, L^G, D^G\}$, where is C^G a set of classes, H_C^G is the class hierarchy, I^G is a set of individuals, R^G is a set of user-defined relations, P^G is a set of data properties, L^G is a set of literal descriptions, and D^G is the ontology dictionary. The process of adding data to the ontology framework from a given set of images follows the following rules:

- The augmented ontology framework must have a similar structure:

$$O_G = \{C', H'_C, I', R', P', L', D'\},$$

- Classes are added to O_G when: $\forall c' \in C' \wedge c' \notin C^G$: $C^G = C^G \bigcup c'$.
- Given class hierarchie H_C^G $(c_P: c)$ and H'_C $(c'_P : c')$, and sets of subclasses $c \subset c_P$, $c' \subset c'_P$, class hierarchies are added to the ontology if:
  - If $c_P \equiv c'_P \bigwedge c \neq c'$, then $c' \subset c_P$, and H_C^G $(c_P: c, c')$;
  - If $c_P \neq c'_P$, then $H_C^G = H_C^G \cup H'_C$
- Individuals I', litera descriptions L', properties P', and relations R' are added if:
  - $\forall i' \in I' \wedge i' \notin I^G, I = \{I' \cup I^G\}$;
  - $\forall l' \in L' \wedge l' \notin L^G, L = \{L' \cup L^G\}$;
  - $\forall p' \in P' \wedge p' \notin P^G, P^G = \{P^G \cup P'\}$;
  - $\forall r' \in R' \wedge r' \notin R^G, R^G = \{R^G \cup R'\}$.
- The definitions $V_i \in D'$ for new words of classes, attributes, and relationships that are added are taken from the WORDNET lexical database if they do not already exist in D.

These definitions are obtained from the semantic WORDNET dictionary definitions. **Fig. 7** depicts an image dataset that has been categorized to determine image classes, which are then compared to ontology classes. No new concepts are added if a class already exists. However, if a new class is identified, the WORDNET definitions are used to enrich the semantic meaning of the ontology.

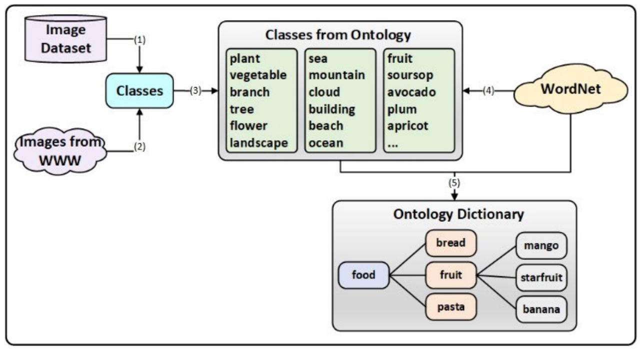


FIGURE 7. Adding new co ncepts for a new classification to the ontology dictionary.

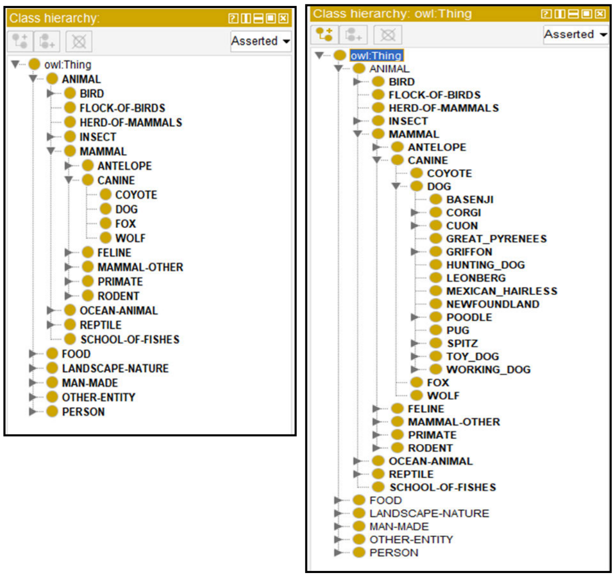
The process of adding data to the ontology framework must ensure that the structure and inheritance of existing information are correct and consistent.

Fig. 8 shows a class hierarchy for an image after the ontology has been enriched. At first, the ontology framework is constructed using the ImageCLEF image set, where the "Dog" class lacks any sub-hierarchies. Nevertheless, by incorporating the Wang image dataset, the "Dog" class is augmented with various subclasses derived from the image set. Consequently, the ontology framework is enriched with additional data. To showcase the effectiveness of ontologies, an ontology-based image search system, named SBIR-GP, is developed.

C. THE ONTOLOGY-BASED SEMANTIC IMAGE RETRIEVAL MODEL

SBIR-GP is an ontology-based image retrieval system that combines the SgGP-Tree machine learning structure and ontology. The SBIR-GP image retrieval system architecture is divided into two stages:

- The pre-processing phase in which features are extracted from images in the dataset, images are segmented to create concept sub-classes, and images are organized for storage on the SgGP-Tree. Simultaneously, a semi-automatic ontology based on the RDF triple language is built and enriched.
- The image retrieval phase searches the SgGP-Tree for sets of similar images, classifies the images using the k-NN algorithm and grSOM to extract visual vector, and generates SPARQL queries to query the ontology to retrieve the high-level semantic output of the input



(a) (b)

FIGURE 8. **Class hierarchy before (a) and after (b) when adding data to ontology.**

query image. The SgGP-Tree structure is a hybrid of the GP-Tree, the Graph-GPTree cluster graph, and grSOM.

Fig. 9 illustrates the ontology-based semantic query system, consisting of two specific phases as follows:

❖ ***Pre-processing phase:***

- **Step 1:** From the image dataset, perform low-level feature extraction and segmentation technique [34] (1);
- **Step 2:** Create a dataset (2) made up of feature vectors and image classifications derived from the segmentation and feature extraction processes.
- **Step 3:** From the data samples, create a combined model of GP-Tree and Graph-GPTree. As initial weights for the SOM network fitted to the graph, a set of weight vectors is trained.
- **Step 4:** Semantic descriptions from image collections (4) and the WWW (5) are used to enrich an ontology framework that has already been built.

❖ ***Image retrieval phase:***

- **Step 1:** The system extracts low-level features from an input query image (6).
- **Step 2:** Using the feature vector, the system queries SgGP-Tree (7): retrieving the most appropriate leaf node on the GP-Tree, then retrieving the set of neighboring leaf nodes of that node on the Graph-GPTree; simultaneously performing a search on grSOM (8) to find the winning cluster, then taking the neighbors of the winning cluster to find the best similar image set (9);
- **Step 3:** Find the visual word vector using the k-NN classification algorithm (10) on a similar image set (11).
- **Step 4:** The SPARQL query is automatically generated from the visual word vector (12); the query (13) is executed on the built ontology (14).
- **Step 5:** The result of the semantic image query process on the ontology includes metadata, URIs (15), a set of similar images and their semantics (16).

From this query process, it is shown that the final set of similar images found is the intersection result of queries on GP-Tree, Graph-GPTree, and grSOM, thus achieving the best image query efficiency among the proposed models. At the same time, classifying on grSOM according to the winning cluster will result in better image classification, leading to more accurate query results on the ontology from these classifications. The result of this query process is a set of similar images, semantic annotations, high-level semantic descriptions, and URIs/IRIs of the images. In the semantic image retrieval architecture, ontology plays a critical role in extracting high-level semantic meaning of the image, in addition to efficient data organization using SgGP-Tree. As a result, an ontology framework [32] is inherited and enriched with additional datasets in this paper to supplement additional hierarchical layers (taxonomy) and concepts for the new layers.

V. THE SEMANTIC IMAGE RETRIEVAL SYSTEM

The SBIR-GP image retrieval system [1] is built to query images based on semantics using the SgGP-Tree and ontology. Given an input image, the SBIR-GP system extracts feature vectors and retrieves similar images based on content sequentially on the SgGP-Tree to retrieve a set of similar images. From the set of similar images based on content, the SBIR-GP performs image classification to extract visual word vectors. At the same time, SPARQL queries (UNION or AND) are automatically generated to query the ontology. For each image in the set of similar images, semantic descriptions are provided with metadata for image annotation, URI identification, etc. At the same time, the semantic concepts of the visual words are extracted from the Wordnet.

The dotNET Framework 4.8 underlies the experimental environment, which was created in C#. Graphs were created using Matlab 2015 to create. An Intel(R) CoreTM i7-9200H processor running at 3.4GHz, 16GB of Memory, and Windows 10 Professional make up the computer setup utilized in the experiment. Image datasets like WANG and ImageCLEF, which are shown in **Table 1**, were utilized in the experiments.

TABLE 1. **Information of experimented image datasets.**

No.	Dataset	Number of images	Categories	Size
1	WANG	10.800	80	62.2 MB
2	ImageCLEF	20.000	276	1.64 GB

The paper employs a number of metrics, including accuracy, recall, F-measure, and query time, to assess the

[1] https://github.com/minhhaidhsp/SBIR-GP

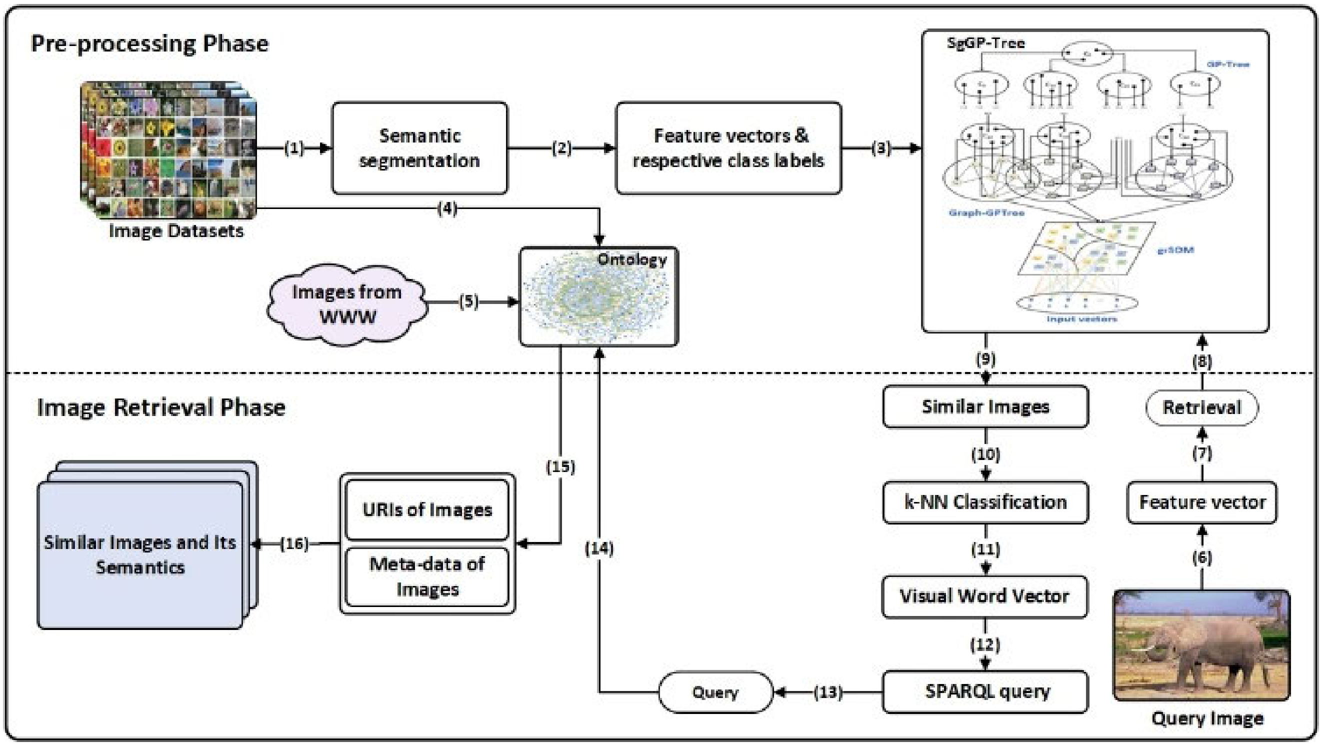


FIGURE 9. The SBIR-GP Semantic image retrieval model.

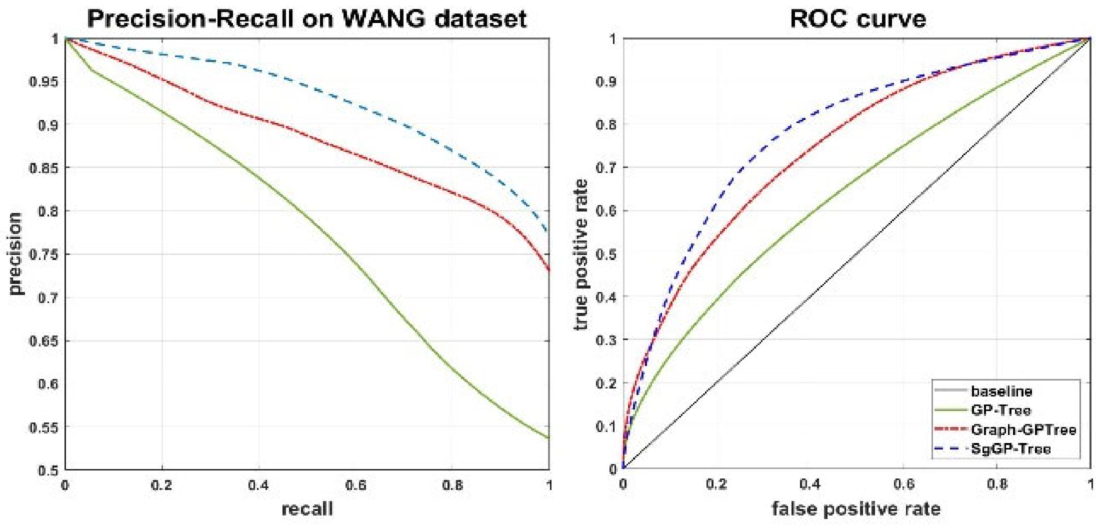


FIGURE 10. Image retrieval performance on GP-Tree, Graph-GPTree, and SgGP-Tree for the WANG image dataset.

efficiency of picture retrieval (in milliseconds). The average performance values and search times of the Wang and ImageCLEF datasets on GP-Tree, Graph-GPTree, and SgGP-Tree are reported in **Tables 2 and 3** based on the performance values that have been obtained.

From the tables above, it can be seen that improving the GP-Tree yields better retrieval accuracy performance for both Wang and ImageCLEF datasets. The Graph-GPTree neighbor graph has better performance than GP-Tree but lower than the SgGP-Tree. However, the retrieval time of GP-Tree is faster than Graph-GPTree and SgGP-Tree.

In addition, to evaluate the retrieval system results, a characteristic curve known as ROC (Receiver Operating Characteristic) is performed. The area under the curve AUC (Area Under the Curve), limited in the ROC space, is a measure of the retrieval accuracy, and the larger the area, the higher the accuracy. Combining accuracy and coverage generates another metric called the Precision-Recall curve (PR curve) to

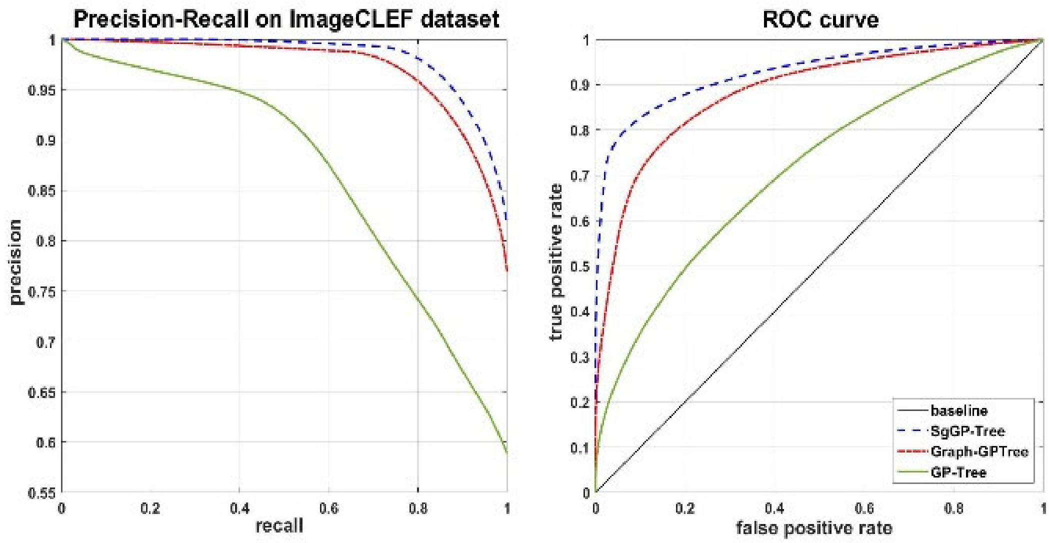


FIGURE 11. Image retrieval performance on GP-Tree, Graph-GPTree, and SgGP-Tree for the ImageCLEF image dataset.

TABLE 2. Retrieval performance of SBIR-GP retrieval system on the wang dataset.

Method	Avg. precision	Avg. recall	Avg. F-measure	Avg. query time (ms)
GP-Tree	0.607822	0.4896184	0.5450811	39.746901
Graph-GPTree	0.766473	0.667659	0.713353	202.793863
SgGP-Tree	0.800408	0.703982	0.748713	696.185404

TABLE 3. Retrieval performance of SBIR-GP retrieval system on the ImageCLEF dataset.

Method	Avg. precision	Avg. recall	Avg. F-measure	Avg. query time (ms)
GP-Tree	0.6062	0.4094	0.4748	44.0891
Graph-GPTree	0.8398	0.7807	0.8064	239.2858
SgGP-Tree	0.8716	0.8224	0.8437	868.5069

evaluate the effectiveness of the image retrieval system. The AUC of the PR curve is similar to the AUC of the ROC curve, meaning that the larger the area under the curve, the higher the accuracy. Based on the experimental data, Precision-Recall curves and ROC curves are performed to evaluate the accuracy of the SBIR-GP retrieval system (**Fig. 10, Fig. 11**).

In the PR curve graph, each curve represents an image folder of each image dataset. The Precision-Recall curves show that the area under the curve of SgGP-Tree is the largest, followed by Graph-GPTree, and the lowest is GP-Tree, indicating that the proposed improvements have enhanced the accuracy. In the ROC curve graph, a baseline diagonal line divides the ROC space into two parts. The points above the diagonal line represent the correct classification results, while the points below the diagonal line represent the incorrect

TABLE 4. Image classification performance of SBIR-GP retrieval system on different image datasets.

Method	Wang		ImageCLEF	
	Precision	Time (ms)	Precision	Time (ms)
GP-Tree	0.6893	14.2307	0.6913	15.9623
Graph-GPTree	0.7711	186.4156	0.8661	214.8923
SgGP-Tree	0.8315	498.1982	0.9345	719.4184

classification results. The ROC curve of the system has points above the baseline line, indicating good image classification results. The image classification results of Graph-GPTree are better than GP-Tree but not as good as SgGP-Tree. The performance of image retrieval on the GP-Tree of the Wang and ImageCLEF datasets shows that the proposed improvement method in the paper is effective.

To evaluate the effectiveness of the image classification in the image retrieval system based on the GP-Tree with k-NN algorithm, we used precision and semantic classification time on the GP-Tree. The more accurate the image classification is, the more accurate the retrieval result on the ontology is. **Table 4** summarizes the accuracy and classification time on the GP-Tree, Graph-Tree, and SgGP-Tree for Wang and ImageCLEF image datasets. **Tables 4** show that the classification accuracy of Graph-GPTree is better than GP-Tree but lower than SgGP-Tree. Therefore, our proposed method to improve GP-Tree has enhanced the image classification accuracy, resulting in better accuracy when querying the ontology.

To evaluate the accuracy and effectiveness of the SBIR-GP image retrieval system, we compared its performance with other research works on the same image dataset. **Table 5** presents the comparison results of our proposed method with other research works on the Wang dataset (consisting of 10,800 images). The comparison results show that our proposed method achieved higher MAP than the mentioned methods, thus being more effective on the Wang image dataset.

Table 6 presents the MAP comparison results of our proposed method with other research works on the ImageCLEF dataset (20,000 images), including: (1) Seymour Z. et al. [35] introduction of a novel method for a human-text matrix using these word vectors to improve tag retrieval results for both user-generated tags and expert labels; (2) HDLA (Hybrid Deep Learning Architecture) method models high-order correlations between visual words to reduce the semantic distance in image search [36]; (3) SDCH method (Semantic Deep Cross-modal Hashing) uses CNN network to extract features and deep hash function to get image semantics [37]; (4) Consistency Preserving Adversarial Hashing (CPAH) method aims to exploit semantic consistency, features are extracted based on CNN network [38].

TABLE 5. Comparison of accuracy among methods on WANG dataset.

Method	Mean Average Precision (MAP)
Color Difference Histogram + HSV+entropy **[39]**	0.703
Fusion feature ResNet-34 + PCA + CNN **[40]**	0.5067
Image signature + BoSW **[41]**	0.78
Texture features + CFBPNN **[42]**	0.60
DSFH (low feature + deep feature VGG-16) **[43]**	0.66
SBIR-GP	**0.8645**

TABLE 6. Comparison of accuracy among methods on ImageCLEF dataset.

Method	Mean Average Precision (MAP)
Seymour Z. et al. [35]	0.420
HDLA (hybrid deep learning architecture) [36]	0.797
SDCH (Semantic Deep Cross-modal Hashing) method [37]	0.803
CPAH (Consistency Preserving Adversarial Hashing) [38]	0.8324
SBIR-GP	**0.8964**

From the above tables, it is evident that our proposed method achieves higher accuracy compared to other retrieval methods on the same dataset. This indicates that our method can effectively extract features to distinguish detailed object features in images such as contours, segments, area, perimeter, etc. Our proposed method is effective in addressing the problem of query and semantic analysis for images with single and multiple objects.

VI. CONCLUSION

This paper proposes methods to improve the performance of image retrieval on GP-Tree. First, a model combining the neighbor graph with GP-Tree, called Graph-GPTree, was created to connect similar elements that are branched out during the node splitting process on GP-Tree. Next, a model combining grSOM and Graph-GPTree, called SgGP-Tree, was created to improve the efficiency of image retrieval. The SgGP-Tree model adds a criterion for selecting winning leaf nodes, making clustering better and image retrieval more accurate. The experiments were performed on the WANG image dataset (10,800 images) and the ImageCLEF dataset (20,000 images). The SBIR-GP system has superior accuracy compared to our previous proposals. Test performance is compared with other methods on the same image dataset to evaluate the proposed model, method, and algorithm. The comparison results show that the SBIR-GP retrieval system is more accurate than other studies on the same image dataset. This demonstrates that our proposals in this article are effective and appropriate.

REFERENCES

[1] G. W. Jiji and P. J. DuraiRaj, "Content-based image retrieval techniques for the analysis of dermatological lesions using particle swarm optimization technique," *Appl. Soft Comput.*, vol. 30, pp. 650–662, May 2015.
[2] H. F. Atlam, G. Attiya, and N. El-Fishawy, "Integration of color and texture features in CBIR system," *Int. J. Comput. Appl.*, vol. 164, no. 3, pp. 23–29, Apr. 2017.
[3] X.-Y. Wang, B.-B. Zhang, and H.-Y. Yang, "Content-based image retrieval by integrating color and texture features," *Multimedia Tools Appl.*, vol. 68, no. 3, pp. 545–569, Feb. 2014.
[4] S. G. Barkalle and P. Jain, "An effective content based image retrieval system based on global representation and multi-level searching," in *Proc. 2nd Int. Conf. Trends Electron. Informat. (ICOEI)*, May 2018, pp. 1329–1335.
[5] G. Pass and R. Zabih, "Comparing images using joint histograms," *Multimedia Syst.*, vol. 7, no. 3, pp. 234–240, May 1999.
[6] G.-H. Liu and J.-Y. Yang, "Content-based image retrieval using color difference histogram," *Pattern Recognit.*, vol. 46, no. 1, pp. 188–198, Jan. 2013.
[7] X.-Y. Wang, Y.-W. Li, H.-Y. Yang, and J.-W. Chen, "An image retrieval scheme with relevance feedback using feature reconstruction and SVM reclassification," *Neurocomputing*, vol. 127, pp. 214–230, Mar. 2014.
[8] M. Yousuf, Z. Mehmood, H. A. Habib, T. Mahmood, T. Saba, A. Rehman, and M. Rashid, "A novel technique based on visual words fusion analysis of sparse features for effective content-based image retrieval," *Math. Problems Eng.*, vol. 2018, pp. 1–13, Jan. 2018.
[9] Z. Mehmood, F. Abbas, T. Mahmood, M. A. Javid, A. Rehman, and T. Nawaz, "Content-based image retrieval based on visual words fusion versus features fusion of local and global features," *Arabian J. Sci. Eng.*, vol. 43, no. 12, pp. 7265–7284, Dec. 2018.
[10] U. Sharif, Z. Mehmood, T. Mahmood, M. A. Javid, A. Rehman, and T. Saba, "Scene analysis and search using local features and support vector machine for effective content-based image retrieval," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 901–925, Aug. 2019.
[11] A. Sarwar, Z. Mehmood, T. Saba, K. A. Qazi, A. Adnan, and H. Jamal, "A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine," *J. Inf. Sci.*, vol. 45, no. 1, pp. 117–135, Feb. 2019.
[12] Z. Mehmood, S. M. Anwar, and M. Altaf, "A novel image retrieval based on rectangular spatial histograms of visual words," *Kuwait J. Sci.*, vol. 45, no. 1, pp. 54–69, 2018.
[13] Z. Mehmood, N. Gul, M. Altaf, T. Mahmood, T. Saba, A. Rehman, and M. T. Mahmood, "Scene search based on the adapted triangular regions and soft clustering to improve the effectiveness of the visual-bag-of-words model," *EURASIP J. Image Video Process.*, vol. 2018, no. 1, pp. 1–16, Dec. 2018.
[14] H. M. Nguyen, V. Thanh The, and T. V. Lang, "A method of semantic-based image retrieval using graph cut," *J. Comput. Sci. Cybern.*, vol. 38, no. 2, pp. 193–212, Jun. 2022.
[15] N. M. Hai, T. V. Lang, and V. T. Thanh, "Semantic-based image retrieval using hierarchical clustering and neighbor graph," in *Proc. World Conf. Inf. Syst. Technol.* Cham, Switzerland: Springer, 2022, pp. 34–44.

[16] C. S. Wickramasinghe, K. Amarasinghe, and M. Manic, "Parallalizable deep self-organizing maps for image classification," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Nov. 2017, pp. 1–7.

[17] J. Filali, H. B. Zghal, and J. Martinet, "Towards visual vocabulary and ontology-based image retrieval system," in *Proc. 8th Int. Conf. Agents Artif. Intell.*, vol. 2, 2016, pp. 560–565.

[18] O. Allani, H. B. Zghal, N. Mellouli, and H. Akdag, "Pattern graph-based image retrieval system combining semantic and visual features," *Multimedia Tools Appl.*, vol. 76, no. 19, pp. 20287–20316, Oct. 2017.

[19] S. Jabeen, Z. Mehmood, T. Mahmood, T. Saba, A. Rehman, and M. T. Mahmood, "An effective content-based image retrieval technique for image visuals representation based on the bag-of-visual-words model," *PLoS One*, vol. 13, no. 4, Apr. 2018, Art. no. e0194526.

[20] B. Yu, "Research on information retrieval model based on ontology," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, pp. 1–8, Dec. 2019.

[21] M. Hirwane, "Semantic based image retrieval," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 6, pp. 120–122, Apr. 2017.

[22] M. N. Asim, M. Wasim, M. U. G. Khan, N. Mahmood, and W. Mahmood, "The use of ontology in retrieval: A study on textual, multilingual, and multimedia retrieval," *IEEE Access*, vol. 7, pp. 21662–21686, 2019.

[23] B. Zhong, H. Li, H. Luo, J. Zhou, W. Fang, and X. Xing, "Ontology-based semantic modeling of knowledge in construction: Classification and identification of hazards implied in images," *J. Construct. Eng. Manage.*, vol. 146, no. 4, Apr. 2020, Art. no. 04020013.

[24] A. B. Spanier, D. Cohen, and L. Joskowicz, "A new method for the automatic retrieval of medical cases based on the RadLex ontology," *Int. J. Comput. Assist. Radiol. Surgery*, vol. 12, no. 3, pp. 471–484, Mar. 2017.

[25] O. Bchir, M. M. Ben Ismail, and H. Aljam, "Region-based image retrieval using relevance feature weights," *Int. J. Fuzzy Log. Intell. Syst.*, vol. 18, no. 1, pp. 65–77, Mar. 2018.

[26] A. Mafla, S. Dey, A. F. Biten, L. Gomez, and D. Karatzas, "Fine-grained image classification and retrieval by combining visual and locally pooled textual features," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 2939–2948.

[27] C. Wang, X. Zhuo, P. Li, N. Chen, W. Wang, and Z. Chen, "An ontology-based framework for integrating remote sensing imagery, image products, and in situ observations," *J. Sensors*, vol. 2020, pp. 1–12, Nov. 2020.

[28] X. Wang, Z. Huang, and F. van Harmelen, "Ontology-based semantic similarity approach for biomedical dataset retrieval," in *Proc. Int. Conf. Health Inf. Sci.* Cham, Switzerland: Springer, 2020, pp. 49–60.

[29] K. Pustu-Iren, G. Bruns, and R. Ewerth, "A multimodal approach for semantic patent image retrieval," in *Proc. 2nd Workshop Patent Text Mining Semantic Technol. (PatentSemTech) Co-Located 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Aachen, Germany: RWTH, 2021, pp. 45–49. [Online]. Available: https://api.semanAcscholar.org/CorpusID:236776781

[30] W. Hu, Y. Sheng, and X. Zhu, "A semantic image retrieval method based on interest selection," *Wireless Commun. Mobile Comput.*, vol. 2022, pp. 1–6, Feb. 2022.

[31] Y. Shi, X. Liu, Y. Wei, Z. Wu, and W. Zuo, "Retrieval-based spatially adaptive normalization for semantic image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11214–11223.

[32] N. T. U. Nhi, T. M. Le, and T. The Van, "A model of semantic-based image retrieval using C-Tree and neighbor graph," *Int. J. Semantic Web Inf. Syst.*, vol. 18, no. 1, pp. 1–23, Feb. 2022.

[33] B. Ionescu, H. Müller, M. Villegas, A. G. S. de Herrera, C. Eickhoff, V. Andrearczyk, Y. D. Cid, V. Liauchuk, V. Kovalev, S. A. Hasan, Y. Ling, O. Farri, J. Liu, M. Lungren, D.-T. Dang-Nguyen, L. Piras, M. Riegler, L. Zhou, M. Lux, and C. Gurrin, "Overview of ImageCLEF 2018: Challenges, datasets and evaluation," in *Proc. Int. Conf. Cross-Language Eval. Forum Eur. Lang.* Cham, Switzerland: Springer, 2018, pp. 309–334.

[34] K. Alaoui and M. Bahaj, "Semantic oriented data modeling based on RDF, RDFS and OWL," in *Proc. Int. Conf. Adv. Intell. Syst. Sustain. Develop.* Cham, Switzerland: Springer, 2020, pp. 411–421.

[35] Z. Seymour and Z. Zhang, "Image annotation retrieval with text-domain label denoising," in *Proc. ACM Int. Conf. Multimedia Retr.*, Jun. 2018, pp. 240–248.

[36] K. S. Arun and V. K. Govindan, "A hybrid deep learning architecture for latent topic-based image retrieval," *Data Sci. Eng.*, vol. 3, no. 2, pp. 166–195, Jun. 2018.

[37] A. Yang, X. Yang, W. Wu, H. Liu, and Y. Zhuansun, "Research on feature extraction of tumor image based on convolutional neural network," *IEEE Access*, vol. 7, pp. 24204–24213, 2019.

[38] L. Xie, J. Wang, B. Zhang, and Q. Tian, "Fine-grained image search," *IEEE Trans. Multimedia*, vol. 17, no. 5, pp. 636–647, May 2015.

[39] H. Qazanfari, H. Hassanpour, and K. Qazanfari, "Content-based image retrieval using HSV color space features," *Int. J. Comput. Inf. Eng.*, vol. 13, no. 10, pp. 533–541, Sep. 2019.

[40] K. Kanwal, K. T. Ahmad, R. Khan, A. T. Abbasi, and J. Li, "Deep learning using symmetry, FAST scores, shape-based filtering and spatial mapping integrated with CNN for large scale image retrieval," *Symmetry*, vol. 12, no. 4, p. 612, Apr. 2020.

[41] A. Ouni, E. Royer, M. Chevaldonné, and M. Dhome, "Leveraging semantic segmentation for hybrid image retrieval methods," *Neural Comput. Appl.*, vol. 34, no. 24, pp. 21519–21537, Dec. 2022.

[42] S. Dhingra and P. Bansal, "Relative examination of texture feature extraction techniques in image retrieval systems by employing neural network: An experimental review," in *Proc. Int. Conf. Artif. Intell. Appl. (ICAIA)*. Singapore: Springer, 2021, pp. 337–349.

[43] G.-H. Liu and J.-Y. Yang, "Deep-seated features histogram: A novel image retrieval method," *Pattern Recognit.*, vol. 116, Aug. 2021, Art. no. 107926.

NGUYEN MINH HAI received the B.Sc. degree in computer science teacher education from the HCMC University of Education, Vietnam, in 2006, and the master's degree in data transmission and computer networks from the Posts and Telecommunications Institute of Technology, Vietnam, in 2011. He is currently pursuing the Ph.D. degree in computer science with the Institute of Information Technology, Vietnam Academy of Science and Technology (VAST), Vietnam. His research interests include data mining, image processing, and image retrieval.

TRAN VAN LANG received the B.Sc. degree in mathematics and the Ph.D. degree in mathematics–physics from the HCMC University of Natural Sciences, Vietnam, in 1982 and 1996, respectively. He was a member of the Directorship of the Institute of Applied Mechanics and Informatics, Vietnam Academy of Science and Technology (VAST); the Dean of the Information Technology Faculty, Lac Hong University (LHU) and Nguyen Tat Thanh University (NTTU); and a Researcher in computational mathematics with the Dorodnitsyn Computing Center, Russian Academy of Science. He has been an Associate Professor in computer science with the Graduate University of Science and Technology, VAST. He was a Senior Principal Research Scientist in computer science with VAST, who has been interested in the development of bioinformatics and parallel computing in Vietnam. His work has appeared in Vietnamese and international journals and proceedings. His research interests include bioinformatics, parallel and distributed computing, computational intelligence, scientific computation, and computational mathematics. He is currently a Editor-in-Chief of *HUFLIT Journal of Science* (HJS) and the Head of Journal Editorial Department of Ho Chi Minh City University of Foreign Languages and Information Technology (HUFLIT).

THANH THE VAN received the B.Sc. degree in mathematics and computer science from the University of Science-Vietnam National University HCMC, in 2001, the master's degree in computer science from Vietnam National University HCMC, in 2008, the Ph.D. degree in computer science from the University of Science, Hue University, Vietnam, in 2016. His research interests include image processing, image mining, and image retrieval.