Multi-label Remote Sensing Image Retrieval using a Semi-Supervised Graph-Theoretic Method

Bindita Chaudhuri[1], Begüm Demir[2], *Senior Member, IEEE,* Subhasis Chaudhuri[1], *Fellow, IEEE,* and Lorenzo Bruzzone[2], *Fellow, IEEE*

***Abstract*—Conventional supervised content-based remote sensing image retrieval systems require a large number of already annotated images to train a classifier for obtaining high retrieval accuracy. Most systems assume that each training image is annotated by a single label associated to the most significant semantic content of the image. However, this assumption does not fit well with the complexity of RS images, where an image might have multiple land-cover classes (i.e., multi-labels). Moreover, annotating images with multi-labels is costly and time consuming. To address these issues, in this paper we introduce a semi-supervised graph-theoretic method in the framework of multi-label RS image retrieval problems. The proposed method is based on four main steps. The first step segments each image in the archive and extracts the features of each region. The second step constructs an image neighborhood graph and uses a correlated label propagation algorithm to automatically assign a set of labels to each image in the archive by exploiting only a small number of training images annotated with multi-labels. The third step associates class labels with image regions by a novel region labeling strategy, whereas the final step retrieves the images similar to a given query image by a subgraph matching strategy. Experiments carried out on an archive of aerial images show the effectiveness of the proposed method when compared to the state-of-the-art RS CBIR methods.**

***Index Terms*—Content-based image retrieval, correlated label propagation, multi-label categorization, region adjacency graph, remote sensing, semi-supervised learning, subgraph matching.**

I. INTRODUCTION

With the advancement of satellite technology, the volume of remote sensing (RS) image archives and the amount of information that can be extracted from them have largely increased. As a result, content-based image retrieval (CBIR) has recently become an important topic of research in RS in order to keep up with the growing need of automatization. A CBIR system generally has two main steps [1]: 1) *Feature extraction* in which the images are described and represented by a set of features, and 2) *Image matching* in which the query image is matched based on the features with all the images in the archive and the most relevant images are retrieved. Accordingly, the performance of the CBIR systems depends on the capability and effectiveness of: i) the extracted features in characterizing the semantic content of the images; and ii) the retrieval algorithms in evaluating the similarity among the considered features. In the RS literature, global image representations based on a set of local descriptors have been found effective. As an example, in [2] local descriptors extracted by the scale invariant feature transform (SIFT) and their bag-of-visual-words representations have been introduced. In [3], bag-of-morphological-words representations of local morphological texture descriptors are computed in the context of CBIR. Local Binary Pattern (LBP) and Local Phase Quantization (LPQ) descriptors, introduced in [4], are defined by initially assigning a binary code to each image pixel by thresholding its neighboring sample values and then computing a histogram of the codes. After extracting the image descriptors, image retrieval is achieved using the k-Nearest Neighbor (k-NN) algorithm or an optimized search strategy [2]–[4], where the retrieval system ranks the images based on their feature similarity with the query image and then displays the most similar images in the order of similarity. All the above mentioned descriptors are potentially effective in modeling RS image content, but they do not model the possible primitives (such as different land-cover classes) present in images and their relationships. This may result in a large semantic gap between the low-level features and the high-level semantic concepts present in RS images.

To narrow down the semantic gap and improve the retrieval performance, few promising supervised and unsupervised methods have been developed in RS. In [5], a system to efficiently and accurately perform content-based shape retrieval of objects from an RS image archive is presented. In [6], [7], methods which model images using graphs (which effectively capture both region characteristics and the spatial relationships among the regions) and compare the similarity among the images using graph matching techniques have been introduced. In our previous paper [6], we have introduced an unsupervised RS image retrieval approach in which each image in the archive is modeled as an attributed relational graph, where the nodes represent region properties and the edges represent the spatial relationships among the regions. Image similarity is then estimated using an inexact graph matching strategy, which jointly exploits a subgraph isomorphism algorithm (for node matching) and a spectral embedding algorithm (for edge matching). This approach has been found to be very promising,

Manuscript received August 23, 2016; revised February 15, 2017 and July 24, 2017; accepted September 11, 2017. Date of publication * *,2017. This work was supported in part by the India-Trento Program for Advanced Research (ITPAR).

B. Chaudhuri was and S. Chaudhuri is with the Vision and Image Processing Laboratory, Department of Electrical Engineering, Indian Institute of Technology Bombay, Mumbai, 400076 India. E-mail: binditac@ee.iitb.ac.in, sc@ee.iitb.ac.in

B. Demir and L. Bruzzone are with the Remote Sensing Laboratory, Department of Information Engineering and Computer Science, University of Trento, Trento, I-38123 Italy. E-mail: begum.demir@unitn.it, lorenzo.bruzzone@ing.unitn.it

Digital Object Identifier *

but it may lead to less accurate retrieval results for query images with highly complex semantic content. In [7], image regions are obtained in a supervised manner using pixel-based classifiers and an iterative split-and-merge technique and scenes are then modeled by attributed relational graphs. As also presented in [8], supervised retrieval methods, which require a large number of already annotated images to train the considered classifier, have been found very effective to reduce the semantic gap. However, most of the existing supervised methods consider each of the training images to be annotated by only a single label describing the most significant connotation of the image in terms of land-cover classes. Fig. 1(a) shows an image that was categorized as *baseball diamond* in a well-known RS archive (see Section III for the details on the archive), but one can see that it contains 3 different primitive classes, such as *grass*, *buildings* and *bare soil*. On the other hand, Fig. 1(b) and 1(c) contain the same primitive classes but, in the considered archive, they were categorized as *dense residential* and *medium residential* respectively. We would like to point out that single (broad category) labels to categorize images may be sufficient for some particular applications, but region level description is required for more complex applications. For example, to distinguish between a beach and a residential area, it is not necessary to separately label 'sea', 'sand', 'buildings', etc. However, it is very difficult to distinguish between categories like 'dense residential' (e.g., see Fig. 1(b)) and 'medium residential' (e.g., see Fig. 1(c)) using broad category labels only, as this requires information about the spatial arrangement of the buildings, pavement etc. Moreover, the multiple class labels (multi-labels) given to an image usually have co-occurrence relationships, i.e., land-cover classes are correlated with each other. For example, land-cover classes such as buildings and pavement are more likely to be assigned to the same image and are thus highly correlated. Hence, the characteristics and geometric arrangement of the multiple primitive classes present in the images are crucial for an efficient retrieval. Accordingly, CBIR methods that properly exploit the multi-labels in RS images are required.

(a) *grass*, *buildings*, *bare soil*, *pavement* (b) *buildings*, *pavement*, *trees*, *cars* (c) *buildings*, *pavement*, *trees*, *cars*

Fig. 1: Examples of images in the considered archive and the multi-labels associated with them.

To address these problems, multi-label learning methods have been recently found very promising in computer vision literature for multi-label image search and retrieval problems, where multiple class labels are simultaneously assigned to each image [9]–[14]. In [9], one versus all Support Vector Machine (SVM) classifiers are used for multi-label scene categorization, where each classifier is trained to solve a binary classification problem defined by one primitive class against all the others. In [10], image regions are classified using one versus one SVM classifiers and an optimal feature subset is selected from standard visual features using a genetic algorithm. The multi-label annotations of the query image are refined using the PageRank algorithm and text-based retrieval is performed at the end. In [11], multi-label k-NN is introduced, which adopts a maximum aposteriori principle to determine the label set of the query image based on the statistical information derived from the label sets of its k nearest neighbors. In [12], canonical correlation analysis, which learns a common subspace and finds the correlation between image features and textual tags, is exploited to perform cross-modal retrieval. In [13], simultaneous recognition and localization of multiple classes in images are performed using random forests, dense pixel matching and genetic algorithm optimization. A comparative study of multi-label classification methods for image annotation and retrieval problems is given in [14]. However, the use of multi-label image retrieval methods is seldom considered in RS. As an example, in [15] multi-label RS image retrieval system is presented. This system initially produces a classification map of the query image and of each image in the archive by using an object-based SVM classifier and neglects the images that do not include the same land-cover classes as the query image. Then, the remaining images are represented by graphs based descriptors and a graph matching technique is applied to retrieve the most similar images to the query image. However, this system has the limitation that in order to perform region classification, it requires a reliable pixel-based training set that is representative of all the land cover classes within each image in the archive (which is critical in large RS archives). We would like to emphasize that pixel-based RS image classification is appropriate for land-cover maps generation problems (where there is a need to classify only a single image), but it is not practical and efficient in real RS image search and retrieval applications, especially when huge archives of RS images are considered. Moreover, collecting annotations on multi-labels is time consuming and costly.

To overcome the above-mentioned critical issues, in this paper we propose a semi-supervised graph-theoretic method in the framework of multi-label RS image retrieval problems. The proposed method is based on four main steps: i) Image segmentation and feature extraction; ii) Multi-label image categorization; iii) Automatic region labeling; and iv) Image retrieval. The method requires that the user initially selects a small fraction of images in the archive as training images, assigns multiple class labels to each image depending on the primitive classes present in it and also annotate a few regions of those images with the corresponding labels. Then, the first step segments each image in the archive into semantically meaningful regions and extracts features from these regions. The second step aims to automatically assign multi-labels to each image in the archive by using a graph-based semi-supervised algorithm. In this algorithm, a neighborhood graph is initially constructed taking all the images in the archive, where the nodes represent the images and the edges represent the neighbors of the images in terms of similarity. Then, the class labels of the training images are propagated using a

correlated label propagation method to the unlabeled images to associate them with multiple labels. Once the user selects an image for query, the system determines the multi-labels associated with the query image and filters out all the images from the archive which do not have those labels. In the third step, the small amount of region labels of the training images is used to associate the multi-labels of each image to the corresponding regions in the image. The fourth step consists of constructing a vertex-labeled edge-weighted undirected graph for each image and matching the graphs using the spectral matching algorithm [16]. Finally, the images similar to the query image are retrieved in the order of graph similarity values. The proposed method is semi-supervised because in the second step, the structure of both labeled and unlabeled data in the feature space (i.e., region descriptors of each image) is used together with labels of regions in the training images. The novelty of the proposed system is the introduction of a semi-supervised graph-theoretic method in the framework of content-based multi-label remote sensing image retrieval.

In order to evaluate the performance of the proposed method, we consider a benchmark archive that is frequently used in RS for CBIR problems (see Section III). In the current literature, the images within the considered archive have been annotated by using only one category label, whereas to use it in our experiments we have relabeled and annotated them with multi-labels by visual inspection. Experiments carried out on the multi-labeled archive demonstrate the effectiveness of the proposed method in terms of the retrieval accuracy. The rest of the paper is organized as follows. Section II describes the proposed method explaining the various steps involved. Section III illustrates the benchmark archive and the experimental settings. Experimental results and discussion are given in Section IV, while Section V draws the conclusion of the work.

II. PROPOSED METHOD

A. Problem formulation

Given a query image $\mathbf{X}_q$ and an archive $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_I\}$ of I RS images, the aim of the proposed semi-supervised method is to retrieve all images from the archive having a similar pattern of regions as the query image. At first, a few images are randomly chosen from $\mathbf{X}$ to form a set $\mathbf{T}$ of training images. Let $\mathcal{L} = \{1, 2, \ldots, |\mathcal{L}|\}$ be the set of all possible class labels associated with the images in the considered archive. Each training image $\mathbf{X}_t \in \mathbf{T}$ is then associated with a vector $\mathbf{L}_t = [l_t^1, l_t^2, \ldots, l_t^{|\mathcal{L}|}]$ of labels, where $l_t^c = 1$ if $\mathbf{X}_t$ contains the class label $c \in \mathcal{L}$ and $l_t^c = 0$ otherwise. Let $\mathbf{r}_t^p$ be the p^{th} labeled region in the set of P labeled regions of $\mathbf{X}_t \in \mathbf{T}$, where P is a small number. Each class label present in $\mathbf{X}_t$ is then assigned to at least one of the P regions until all the P regions are labeled. Complete list of the notations and the related definition used in this paper is given in Appendix. In order to retrieve the visually most similar images to $\mathbf{X}_q$ from $\mathbf{X}$ using the multi-label information of the images, the proposed method is characterized by four main steps: i) Image segmentation and feature extraction; ii) Multi-label image categorization using the multi-labels of the training images, iii) Automatic region labeling using a few region labels of the training images, and iv) Image retrieval based on a subgraph matching strategy. Fig. 2 presents the block scheme of the proposed semi-supervised graph-theoretic method. The steps are explained in detail in the subsequent subsections.

B. Image segmentation and feature extraction

This step aims to obtain the regions and their features from each image in the archive. To obtain the regions, each image $\mathbf{X}_i$ is segmented into n_i semantically meaningful regions which form the set $\{r_i^1, r_i^2, \ldots, r_i^{n_i}\}$, where r_i^k is the k^{th} region of $\mathbf{X}_i$. Unsupervised segmentation of the images is achieved using the parametric kernel graph cut algorithm [17]. The algorithm initially involves implicit mapping of the non-linearly separable image data into a higher dimensional space using a kernel function. Then the objective functional is optimized using an iterative two-step process: 1) fixing the labeling and minimizing the functional with respect to the region parameters; and 2) given the region parameters, minimizing the functional with respect to the image partitioning. This ensures the convergence of the algorithm and gives optimally segmented regions. It is worth noting that the proposed method is independent of the choice of the segmentation algorithm. However, the accuracy of the segmentation results semantically affect the performance of the proposed CBIR technique. Hence an efficient unsupervised segmentation algorithm which segments images into semantically meaningful regions should be chosen. After segmentation, features are extracted from each region to create a feature vector $\mathbf{f}_i^k$ for each region r_i^k. The reader is referred to Section III for detailed information about the features used in our method.

C. Multi-label image categorization

This step aims to assign multi-labels to each image in the archive as well as to the query image based on the characteristics of the regions present in the image. The multi-label image categorization task is achieved in three substeps. Firstly, a neighborhood graph is constructed using the images in the archive as the nodes and their feature similarity relationships as the edges. Secondly, class labels are propagated from the training images to the unlabeled images to obtain their label scores using the edge weights determined from the image pair distances. Finally, the label scores of the unlabeled images are thresholded appropriately to convert them into binary values, thereby associating each image in the archive with class labels. Thus, this step adopts a semi-supervised approach to label the unlabeled images with a few multi-labeled images by exploiting the correlation that exists among the multi-labels. The substeps are described below.

1) Neighborhood graph construction: This substep aims to construct a graphical structure with the images as nodes such that each image is connected to only a few images in its neighborhood. A neighborhood graph for an archive is defined as a graph $\mathcal{G} = (\mathbf{X}, \mathsf{E}, \mathbf{W})$, where E is the set of graph edges and $\mathbf{W} \in \mathbb{R}^{I \times I}$ is the weight matrix of the graph containing edge information. Each node represents an image $\mathbf{X}_i$ in the

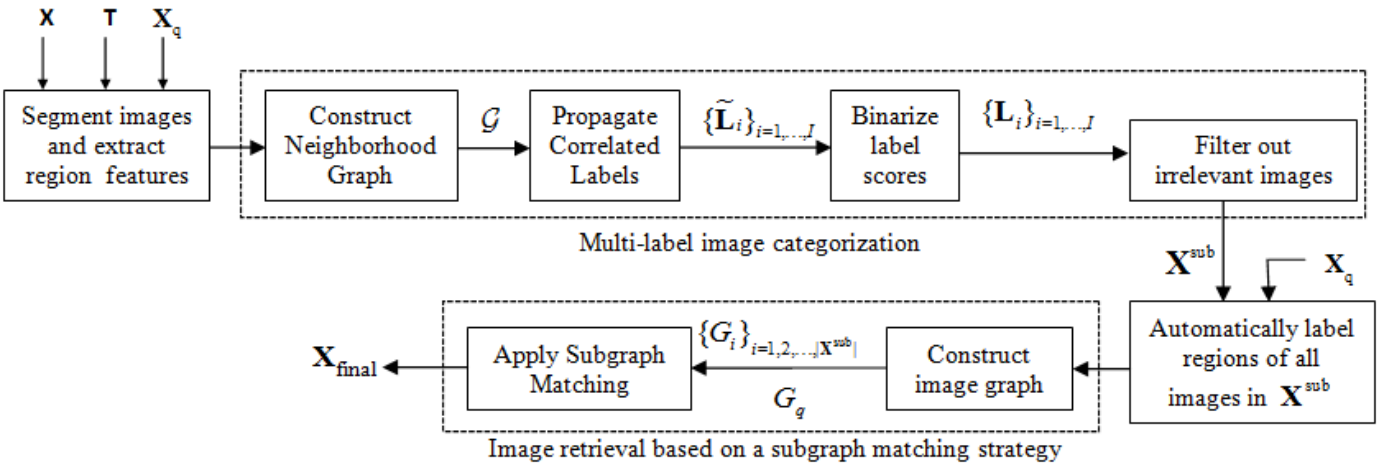


Fig. 2: Block diagram of the proposed semi-supervised graph-theoretic method

archive and $\mathbf{N}_i$ denotes the set of K neighboring images of $\mathbf{X}_i$ according to some similarity measure. An edge is defined between $\mathbf{X}_i$ and $\mathbf{X}_j$ if $\mathbf{X}_j \in \mathbf{N}_i$ and is then assigned a corresponding weight $\mathbf{W}(i,j)$ so as to minimize the following value [18]:

$$\zeta = \sum_{i=1}^{I} ||\mathbf{X}_i - \sum_{j:\mathbf{X}_j \in \mathbf{N}_i} \mathbf{W}(i,j)\mathbf{X}_j||^2 \quad (1)$$

$$\text{s.t. } \mathbf{W}(i,j) \geq 0 \; \forall \; i,j \text{ and } \sum_{j:\mathbf{X}_j \in \mathbf{N}_i} \mathbf{W}(i,j) = 1$$

The intuitive idea behind minimizing the above cost function is to approximate each image representation by a weighted linear sum of the image representations lying in its neighborhood. Hence, the weights $\mathbf{W}(i,j)$ are also known as the reconstruction weights. The values of $\mathbf{W}(i,j)$ for those values of i,j for which edges do not exist are set to 0.

In order to compute the reconstruction weight $\mathbf{W}(i,j)$ that satisfies (1) for all $\mathbf{X}_j \in \mathbf{N}_i$, the distances between $\mathbf{X}_i$ and each $\mathbf{X}_j$ are computed, then arranged in descending order (so that the most similar image to $\mathbf{X}_i$ is placed at the beginning of the order and the least similar one at the end) and then normalized to sum-to-one. Thus higher the similarity of an image with its neighbor, higher is the weight assigned to the corresponding edge connecting them in the graph $\mathcal{G}$. The same process is repeated for each $\mathbf{X}_i$, $i = 1, 2, \ldots, I$ to construct the final weight matrix $\mathbf{W}$ of the neighborhood graph. The features and the distance metric we have used to find the nearest neighbors of an image are given in Section III.

2) Correlated label propagation: After the calculation of the reconstruction weights of the neighborhood graph, we aim to transmit the class labels from the training images to the unlabeled images. Most of the prevalent approaches propagate the labels independently without taking into consideration the inherent correlation that exists among the multi-labels. In the proposed method, we adopt the scheme introduced in [18] to exploit this correlation and thereby propagate the labels all at a time. The label vector of an unlabeled image is obtained as a weighted combination of the label vectors of its neighboring images. This is due to the fact that an image most likely contains the same primitive classes as the ones present in its similar images with the most similar image having the greatest influence.

In order to form the weighted combinations, we create a matrix of the label vector values, as in [18], and update the values of the matrix by performing elementary row operations using the reconstruction weights in $\mathbf{W}$. We start by creating a matrix $\mathbf{Y} \in \{0,1\}^{I \times |\mathcal{L}|}$, whose i^{th} row is equal to the label vector $\mathbf{L}_i$ of $\mathbf{X}_i \in \mathbf{X}$. The label vector of an unlabeled image $\mathbf{X}_{t'} \in \mathbf{X} \setminus \mathbf{T}$ is initiated to $\mathbf{L}_{t'} = \{0\}^{1 \times |\mathcal{L}|}$. A matrix $\tilde{\mathbf{Y}}$ is then initialized with the values of $\mathbf{Y}$ and subsequently updated by performing the following iteration until the values of $\tilde{\mathbf{Y}}$ converge.

$$\tilde{\mathbf{Y}} = \beta \mathbf{W} \tilde{\mathbf{Y}} + (1-\beta)\mathbf{Y} \quad (2)$$

Here β $(0 < \beta < 1)$ is a parameter that determines the amount of label information each image receives from its neighbors compared to its existing label information. Hence after each iteration, the new label vector of an image is the weighted average of its original label vector and a weighted combination of the label vectors of the most similar images. After the convergence, $\tilde{\mathbf{Y}}(i,c)$ $\forall$ $i = 1, 2, \ldots, I$ denotes the c^{th} label score for $\mathbf{X}_i$. The convergence analysis of (2) is given in [18]. It is to be noted that although the values of the label vectors are known for the training images, they are changed to real-valued label scores during the iteration of (2). These changed values are used for label score binarization in the next substep. Each image $\mathbf{X}_{t'} \in \mathbf{X} \setminus \mathbf{T}$ is now associated with a vector $\tilde{\mathbf{L}}_{t'} = [\tilde{l}^1_{t'}, \tilde{l}^2_{t'}, \ldots, \tilde{l}^{|\mathcal{L}|}_{t'}]$ of real-valued label scores, where $\tilde{l}^c_{t'} = \tilde{\mathbf{Y}}(t', c)$.

3) Label score binarization: This is the last substep of the multi-label categorization algorithm which aims to discretize the label scores obtained from label propagation into binary values in order to indicate the presence or absence of a class label in a particular image. Our approach of label score binarization is inspired to [19]. For binarization, a threshold th is determined by taking the minimum of the label score values of all the training images. The elements of the label vector of each unlabeled image $\mathbf{X}_{t'} \in \mathbf{X} \setminus \mathbf{T}$ are then calculated as $l^c_{t'} = 1$ if $\tilde{l}^c_{t'} \geq th$ and $l^c_{t'} = 0$ otherwise. Thresholding of the

label scores results in converting the score vector $\tilde{\mathbf{L}}_{t'}$ to the label vector $\mathbf{L}_{t'}$ for each unlabeled image. Thus the multi-label categorization algorithm finally associates each image $\mathbf{X}_i \in \mathbf{X}$ with a label vector $\mathbf{L}_i$. When the query image $\mathbf{X}_q$ is given as input to the system, the retrieval system performs multi-label categorization of $\mathbf{X}_q$ to determine the class labels (vector $\mathbf{L}_q$) associated with it. The system then searches for all images $\mathbf{X}_i \in \mathbf{X}$ for which $\mathbf{L}_q \cdot \mathbf{L}_i \geq 1$ and forms a set $\mathbf{X}^{\text{sub}} \subset \mathbf{X}$ of the images satisfying the criterion. The remaining images in the archive are then filtered out and only the images in $\mathbf{X}^{\text{sub}}$ are considered for the subsequent steps of the method. The performance of this step, which greatly affects the retrieval results, depends on how well the system learns the characteristic features of the regions and identifies the primitive classes contained in the unlabeled images. For example, if a sand region in an image is not identified as sand and the query image contains the class *sand*, that image is not retrieved by the system as it is eliminated as an irrelevant image after this step. We have chosen our features carefully to avoid this problem.

D. Automatic region labeling

The third step of the proposed method is automatic region labeling, which aims to exploit a few labeled regions of the training images to assign an appropriate label to each region of the images $\mathbf{X}_i \in \mathbf{X}^{\text{sub}}$. It is worth noting that the region labels of only those training images which belong to the set $\mathbf{X}^{\text{sub}}$ are ultimately used while querying. In each training image $\mathbf{X}_t \in \mathbf{X}^{\text{sub}}$, labels are assigned to a small number of regions based on a manual visual analysis. Each of the remaining unlabeled regions in $\mathbf{X}_t$ is then assigned with the label of the most similar labeled region. The similarity of an unlabeled region with a labeled region is computed by the distance $d(\cdot,\cdot)$ between their corresponding feature vectors. The distance measures we have used in our method are given in Section III. This approach requires that each primitive class present in $\mathbf{X}_t$ has at least one corresponding labeled region for an accurate classification. Fig. 3 shows an example of how region labeling is done for a training image. After completing the labeling of regions of all the training images $\mathbf{X}_t \in \mathbf{X}^{\text{sub}}$, the prototype feature vector for each class label $c \in \mathcal{L}$ is computed by taking the average of the feature vectors of all the newly formed regions in the training images having the label c.

(a) (b) (c)

Fig. 3: Example of region labeling of a training image. (a) Original image with labeled regions, (b) Segmented image and (c) Region-labeled image. In (a), the regions with red squares are labeled as *airplane*, the regions with yellow stars are labeled as *grass*, the regions with green circles are labeled as *cars* and the regions with blue triangles are labeled as *bare soil*. In (b), each region with a different color indicates a different segment. In (c), the black regions denote *airplane*, the dark grey regions denote *cars*, the light grey regions denote *grass* and the white regions denote *bare soil*.

In the case of unlabeled images $\mathbf{X}_{t'} \in \mathbf{X}^{\text{sub}}$ having no labeled regions, all regions are labeled in the same manner. To label a region $r_{t'}^k$ of $\mathbf{X}_{t'}$, we compute the dissimilarity $d(\cdot,\cdot)$ between the feature vector $\mathbf{f}_{t'}^k$ and each of the prototype feature vectors of only those class labels which are associated with $\mathbf{X}_{t'}$. Then, $r_{t'}^k$ is assigned the label c for which the value of $d(\cdot,\cdot)$ is minimum. Thus, after region labeling, each class label c contained in $\mathbf{X}_{t'} \in \mathbf{X}^{\text{sub}}$ is assigned to one or more regions of $\mathbf{X}_{t'}$. Fig. 4 shows an example of how region labeling is done for an unlabeled image.

(a) (b) (c)

Fig. 4: Example of region labeling of an unlabeled image. (a) Original image, (b) Segmented image and (c) Region-labeled image. In (b), each region with a different color indicates a different segment. In (c), the black regions denote *buildings*, the darkest grey regions denote *tanks*, the medium grey region denotes *pavement*, the lightest grey region denotes *bare soil* and the white regions denote *trees*.

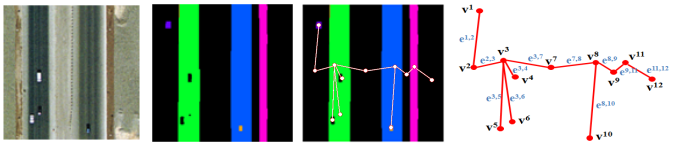
E. Image retrieval based on a subgraph matching strategy

The fourth and final step of the proposed method aims to construct a Region Adjacency Graph (RAG) for the query image $\mathbf{X}_q$ and each image $\mathbf{X}_i \in \mathbf{X}^{\text{sub}}$ and measure the similarity of the images using a subgraph matching strategy. A RAG of an image $\mathbf{X}_i$ is defined as $G_i = (V_i, E_i, \mathbf{A}_i)$, where $V_i = \{r_i^1, r_i^2, \ldots, r_i^{n_i}\}$ is the set of n_i nodes, $E_i = \{e_i^{(s,t)} \mid s,t \in \{1,2,\ldots,n_i\}\}$ is the set of edges that link the nodes and $\mathbf{A}_i \in \mathbb{R}^{n_i \times n_i}$ is the weighted adjacency matrix containing edge information. An edge $e_i^{(s,t)}$ exists if the regions r_i^s and r_i^t are adjacent to each other. To define the edges, we construct from the segmented $\mathbf{X}_i$ an adjacency matrix $\mathbf{A}_i$, whose initial entries are logical 0s and 1s, with $\mathbf{A}_i(r_i^s, r_i^t) = 1$ if $e_i^{(s,t)}$ exists and $\mathbf{A}_i(r_i^s, r_i^t) = 0$ otherwise. Each edge is then assigned an attribute as follows:

$$\mathbf{A}_i(r_i^s, r_i^t) = \alpha_1 \, ||\mathbf{f}_i^s - \mathbf{f}_i^t||_2 + \alpha_2 \left(||c_{r_i^s} - c_{r_i^t}||_2 + |\theta_{r_i^s} - \theta_{r_i^t}|\right) \quad (3)$$

where $c_{r_i^s}$ and $c_{r_i^t}$ are the centroids of the pixel co-ordinates within the regions r_i^s and r_i^t, respectively and $\mathbf{f}_i^s$ and $\mathbf{f}_i^t$ are the feature vectors of the regions r_i^s and r_i^t, respectively. $\theta_{r_i^s}$ and $\theta_{r_i^t}$ are the orientation angles of those regions (angle between the horizontal axis and the major axis of the ellipse having the same 2^{nd} moments as the region, such that $\theta \in [-90^0, +90^0]$) and $||\cdot||_2$ is the L_2 norm. The orientation angles are included for retrieval of images with high visual similarity and the choice of using the orientation angles depends on the final goal of retrieval. If the orientation angle is not relevant for the objects in the images in a given archive, it can be neglected. Each term is normalized before adding so that all the values are comparable when combined. It is worth noting that the

sequencing of the regions and class labels does not affect the shape of the constructed graphs. The weights α_1 and α_2 should be selected after taking into account the importance of the related variables in the retrieval. For undirected property of the graphs, we have made $\mathbf{A}_i$ symmetric, i.e. $\mathbf{A}_i(r_i^s, r_i^t) = \mathbf{A}_i(r_i^t, r_i^s)$. The diagonal elements of $\mathbf{A}_i$ are set as $\mathbf{A}_i(r_i^s, r_i^s) = c$ where c is the numeric value of the label assigned to the region r_i^s. In order to better understand this step, Fig. 5 shows a qualitative example of RAG construction for an image. Note that the graph shown in the figure is not a rigid structure, it is merely a diagrammatic representation for visual understanding.



(a) (b) (c) (d)

Fig. 5: Example of step-wise graph construction from an image - (a) original image; (b) segmented image; (c) node formation and edge attribute description; (d) created graph.

After graph construction, the method measures the similarity between $\mathbf{X}_q$ and each image $\mathbf{X}_i \in \mathbf{X}^{\text{sub}}$ by matching the corresponding graphs G_q and G_i and ranks the images in the order of graph similarity values. Spectral graph matching methods, which utilize the property of invariance of eigenvalues for isomorphic graphs, have recently gained popularity as effective graph matching techniques. In the proposed method, we have used the Spectral Matching (SM) [16] algorithm to solve the optimization problem. Although other methods available in literature [20]–[23] have been proved to give more accurate matching results compared to the SM algorithm, they require computational time which is several orders of magnitude higher than that taken by the SM algorithm. Hence the SM algorithm is chosen to provide the solution to the graph matching problem quickly and efficiently in order to achieve fast retrieval.

The SM algorithm formulates the problem of subgraph matching as the problem of finding the optimal indicator vector $\mathbf{y}^* \in \{0,1\}^{n_q n_i}$ which maximizes a quadratic score function $\mathbf{y}^* = \text{argmax}(\mathbf{y}^T\mathbf{M}\mathbf{y})$ where $\mathbf{M} \in \mathbb{R}^{n_q n_i \times n_q n_i}$ is the affinity matrix, subject to the one-to-one (or many-to-one depending on requirement) *matching constraint* and the *integer constraint* (which ensures that $\mathbf{y}$ can take only binary values). Each diagonal element of $\mathbf{M}$, i.e. $\mathbf{M}(us, us)$, measures the similarity between the regions (nodes) r_q^u of $\mathbf{X}_q$ and r_i^s of $\mathbf{X}_i$, and is hence computed as $\mathbf{M}(us, us) = \mathrm{e}^{-|n^{c'} - n^c|}$, where c' and c are the labels of r_q^u and r_i^s, respectively. Each non-diagonal element of $\mathbf{M}$, i.e. $\mathbf{M}(us, vt)$, measures the similarity between the edges $e_q^{(u,v)}$ and $e_i^{(s,t)}$, and is hence computed as $\mathbf{M}(us, vt) = \mathrm{e}^{-|\mathbf{A}_q^{(r_q^u, r_q^v)} - \mathbf{A}_i^{(r_i^s, r_i^t)}|}$. By this definition of the values of $\mathbf{M}$, $\mathbf{y}_{us}^* = 1$ if the region r_q^u corresponds (has the same label and similar linking edges) to the region r_i^s and $\mathbf{y}_{us}^* = 0$ otherwise. The SM algorithm returns the L1 normalized principal eigenvector of $\mathbf{M}$ as the solution $\mathbf{y}^*$. For a detailed description of the SM algorithm, we refer the reader to [16]. The graph similarity between G_q and G_i is then computed as $\text{GD}(G_q, G_i) = (\mathbf{y}^*)^T\mathbf{M}\mathbf{y}^*$. Higher the value of $\text{GD}(G_q, G_i)$, higher is the similarity between $\mathbf{X}_q$ and $\mathbf{X}_i$. Hence, the graph similarity values are arranged in descending order and the most similar images to $\mathbf{X}_q$ are retrieved in that order, which form the set $\mathbf{X}^{\text{final}}$ of retrieved images. The full algorithm of the proposed method is described in Algorithm 1.

Algorithm 1: Algorithm of the proposed semi-supervised graph-theoretic method

Input : An archive $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_I\}$, the set of training images $\mathbf{T}$, the set $\mathcal{L} = \{1, 2, \ldots, |\mathcal{L}|\}$ of class labels and a query image $\mathbf{X}_q$

Output: Images from the archive which are similar to $\mathbf{X}_q$

begin
- **for** i = 1 **to** I **do**
  - Segment $\mathbf{X}_i$ and compute $\mathbf{f}_i^k \; \forall \; r_i^k \in \{r_i^1, r_i^2, \ldots, r_i^{n_i}\}$
  - Calculate distance between $\mathbf{X}_i$ and $\mathbf{X}_{j \neq i}$ and find $\mathbf{N}_i$ (K nearest neighbors) for $\mathbf{X}_i$
- **end**
- Find the weights and construct the neighborhood graph $\mathcal{G}$.
- Find the label score vector $\tilde{\mathbf{L}}_{t'}$ of each $\mathbf{X}_{t'} \in \mathbf{X} \setminus \mathbf{T}$ by iteratively updating the label matrix $\tilde{\mathbf{Y}}$ as $\tilde{\mathbf{Y}} = \beta\mathbf{W}\tilde{\mathbf{Y}} + (1-\beta)\mathbf{Y}$ until convergence
- Binarize $\tilde{\mathbf{L}}_{t'}$ using an appropriate threshold th to obtain $\mathbf{L}_{t'}$.
- For $\mathbf{X}_q$, find $\mathbf{L}_q$ and form a set $\mathbf{X}^{\text{sub}}$ of images satisfying $\mathbf{L}_q \cdot \mathbf{L}_i \geq 1$
- **for** all training images in $\mathbf{X}^{\text{sub}}$ **do**
  - Label each of the remaining unlabeled regions of $\mathbf{X}_t$ with the label of the adjacent and the most similar labeled region.
- **end**
- Find the characteristic feature vector for each class label $c \in \mathcal{L}$
- **for** all unlabeled images in $\mathbf{X}^{\text{sub}}$ **do**
  - Label $r_{t'}^k = c$ if $\mathbf{f}_{t'}^k$ is most similar to the characteristic feature vector of c
- **end**
- **foreach** $i : \mathbf{X}_i \in \mathbf{X}^{\text{sub}}$ **do**
  - Create a vertex-labeled edge-weighted undirected graph $G_i = (V_i, E_i, \mathbf{A}_i)$
  - Calculate $\text{GD}(G_q, G_i)$ using the spectral matching algorithm
- **end**
- Sort the $\text{GD}(G_q, G_i)$ values in descending order and obtain $\mathbf{X}^{\text{final}}$.

end

III. DATASET DESCRIPTION AND EXPERIMENTAL SETUP

A. Dataset description

In order to evaluate the performance of the proposed semi-supervised method, we have conducted experiments on the UCMERCED archive of 2100 images, each of size 256×256 pixels, taken from aerial orthoimagery and broadly grouped into 21 different categories. Further information about the archive can be obtained in [2]. Since the proposed method is based on multi-label retrieval, we have relabeled the images in the archive. Each image in the archive has been manually labeled with one or more (maximum seven) labels based on visual inspection in order to create the ground truth data (the multi-labels are available at http://bigearth.eu/datasets). The

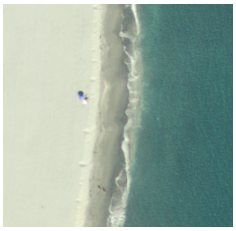
total number of distinct class labels associated with $\mathbf{X}$ for the considered archive is $|\mathcal{L}| = 17$. A few images are then randomly chosen from the labeled data to form the set $\mathbf{T}$ of training images and the labels of the remaining images are considered only while evaluating the retrieval performance. The retrieval is done on the basis of the choice of labels by the user. Fig. 6 shows an example image from each category and the multi-labels associated with them after relabeling the archive. Table I lists the class labels associated with the images for each of the broad categories in the archive. The category names are given based on our understanding of the categories with respect to the primitive classes present in their images. The original category names (if any) in the archive are given in brackets. We would like to point out that each image belonging to a broad category (in the first column of Table I) does not need to be associated with all the class labels in the corresponding row (in the second column of Table I) and may be associated only with some of them. The number of images present in the archive associated with each of the newly defined class labels is listed in Table II.

B. Experimental settings

In our experiments, the regularization parameter for the segmentation algorithm is experimentally chosen to be 0.75. After segmentation, the number of regions n_i in each $\mathbf{X}_i$ lies in the range between 2 and 50. Each region is described by the following features concatenated together to form a 232-dimensional feature vector:- 1) *Shape features* [24] (which include Fourier descriptors and contour-based shape descriptors); 2) *Intensity features* (which consist of the mean, standard deviation and skewness of the samples within each region in each spectral channel); 3) *Texture features* (which include entropy and spectral histogram [25]). Further details about the features can be found in [6]. Image similarity for neighborhood graph construction in the proposed method has been measured using the Earth Mover's Distance (EMD) [26], due to its ability to handle variable-length image representations and its robustness to inaccurate image segmentation. The weight for a feature vector $\mathbf{f}_i^k$ is taken to be the normalized area (in terms of number of pixels) in the corresponding region r_i^k of $\mathbf{X}_i$. To evaluate the efficiency of EMD in finding neighboring images, we have compared the results obtained by EMD with those obtained by four state-of-the-art image descriptors, i.e. GIST [27], SIFT, LBP and LPQ descriptors, each used with the chi-square distance. While the above-mentioned methods resulted in an average precision of less than 55% in finding correct image neighbors (according to the broad category labels associated with the images), EMD resulted in a precision of more than 70%. Although EMD is costly in terms of computational complexity compared to the other methods, achieving high accuracy with the multi-label categorization algorithm is of foremost importance for obtaining satisfactory performance of the proposed retrieval system. These reasons justify the choice of EMD for finding image neighborhoods as opposed to other state-of-the-art methods. The value of β in (2) is set to 0.99 as suggested in [28]. After the filtering of irrelevant images in the first step, the set $\mathbf{X}^{\text{sub}}$ contains an average of 900 images. In the region labeling step, we have labeled an average of 5 regions in each training image, which constitute only 10% of the regions in the image. To compute the distance $d(\cdot,\cdot)$ between the feature vectors of two regions for labeling, we have used different distance measures to evaluate similarities between node attributes: *City block* distance is considered for Fourier descriptors and intensity features, whereas *Euclidean* distance is considered for entropy and contour-based shape features and *Chi-square* distance is selected for spectral histograms. The weights α_1 and α_2 in (3) are empirically set to 0.8 and 0.2, respectively, in order to give more importance to the characteristics of the regions linked by an edge in determining the edge weight rather than to the geometric properties of the regions.

In order to evaluate the performance of the proposed Multi-Label Image Retrieval Method (denoted as MLIRM hereafter), we have considered four state-of-the-art methods for comparison: 1) the k-NN algorithm applied to the image descriptors obtained by the LPQ [4] (denoted as KNN); 2) the Attributed Relational Graph Modeling and Matching Method proposed in [6] (denoted as ARGMM); 3) the multi-label SVM method [9] (denoted as ML-SVM), which consists of a parallel architecture made up of 17 one versus all SVMs (one for each primitive class) to assign multi-labels to the query image and uses LPQ image descriptors to model the images; and 4) the multi-class SVM method [10] (denoted as MC-SVM), in which we train a one versus all SVM classifier for each primitive class to classify each region in an image using our region descriptors. The first two methods are unsupervised, whereas the latter two are supervised. It is worth noting that there are no multi-label image retrieval methods available in RS CBIR literature that can be used for comparison. To obtain the LPQ descriptors for an image, we have considered a 3×3 neighborhood for each sample, computed a binary code for each pixel position as described in [4] and finally formed an L1 normalized histogram of the codes to form a 256-dimensional descriptor for each image. The k-NN algorithm is then applied by using the G statistic similarity measure for image similarity computation [4]. For ARGMM, we have initially segmented and constructed an attributed relational graph for each image in the archive by extracting some features from the regions. Images similar to the query image are then determined by matching the graph of the query image with all the graphs in the archive using an inexact graph matching strategy. Then the images are ranked in the order of graph similarity values. Further details about the method can be obtained in [6]. ML-SVM and MC-SVM were implemented by using a radial basis function kernel and the kernel parameters have been chosen via 5-fold cross validation. For the ML-SVM, each of the 17 trained SVM classifiers independently predicts a vector. The vectors are then summed up to get the final label vector $\mathbf{L}_q$ of the query image, given the 256-dimensional LPQ descriptor of the query image as input. The number of training instances is 315, since we have chosen 15 training images from each of the 21 broad categories. The choice of 15 images is justified in Section IV. For the MC-SVM method, the trained classifiers predict the label of a region, given the 232-dimensional feature vector of the region. Since

(a) *airplane*, *bare soil*, *grass*, *cars*



(b) *sand*, *sea*

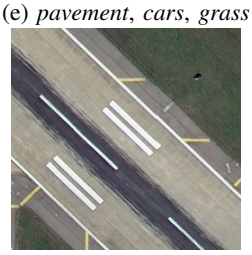
(c) *water*, *trees*, *bare soil*

(d) *trees*, *bare soil*

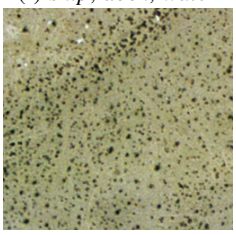
(e) *pavement*, *cars*, *grass*

(f) *ship*, *dock*, *water*

(g) *pavement*, *cars*, *bare soil*, *trees*



(h) *pavement*, *grass*



(i) *sand*, *chaparral*

(j) *bare soil*, *grass*, *trees*

(k) *court*, *grass*, *trees*

(l) *cars*, *pavement*

(m) *buildings*, *pavement*, *cars*

(n) *field*, *trees*

(o) *mobile home*, *pavement*, *trees*

(p) *pavement*, *cars*, *buildings*

(q) *buildings*, *bare soil*, *pavement*

(r) *tanks*, *pavement*

(s) *bare soil*, *grass*, *buildings*

(t) *buildings*, *pavement*, *cars*

(u) *buildings*, *pavement*, *trees*

Fig. 6: Example of an image from each category in the considered archive and its associated class labels.

TABLE I: Multi-labels associated with the images of each category in the archive.

Category names	Associated multi-labels
agricultural	field, trees
airport (airplane)	airplane, pavement, grass, buildings, cars
baseball diamond	bare soil, pavement, grass, trees, buildings
beach	sea, sand, trees
urban area (buildings)	buildings, pavement, cars
chaparral	sand, chaparral
dense residential	buildings, pavement, trees, cars
forest	trees, bare soil
freeway	pavement, cars, grass, trees, bare soil
golf course	grass, trees, bare soil
harbor	ship, dock, water
intersection	pavement, bare soil, cars, buildings, grass
medium residential	buildings, cars, trees, grass, pavement, bare soil
mobile home park	mobile home, pavement, cars, trees, bare soil
overpass	pavement, cars, bare soil, grass, trees
parking lot	cars, pavement, bare soil, grass
river	water, trees, bare soil
runway area (runway)	pavement, grass, bare soil
sparse residential	buildings, grass, bare soil, trees, sand, chaparral
storage tanks	tanks, bare soil, grass, pavement, buildings
tennis area (tennis court)	court, grass, trees, pavement, buildings

TABLE II: Number of images present in the archive for each class label.

Class label	Number of images
airplane	100
bare soil	633
buildings	696
cars	884
chaparral	119
court	105
dock	100
field	106
grass	977
mobile home	102
pavement	1305
sand	389
sea	100
ship	102
tanks	100
trees	1015
water	203

we concatenate feature vectors of various lengths into a single feature vector, we normalize the features according to [29]. The number of training regions in this method is 10108. Each region in the query image is classified into one of the 17 primitive classes by the trained classifiers, and the label vector $\mathbf{L}_q$ of the query image is obtained from the classes present in it. The similarity between two images is finally computed using the Hamming distance between their corresponding label vectors for both the methods.

Since we have considered multi-labels for each image in the archive, conventional single-label based retrieval performance evaluation metrics are not suitable for our multi-label retrieval system. Hence in our experiments with the multi-label information, results of each method are provided in terms of three special performance evaluation metrics [30]: 1) Accuracy, 2) Precision and 3) Recall. In order to define these metrics, let $\mathcal{L}_r \subset \mathcal{L}$ be the set of class labels present in the retrieved image $\mathbf{X}_r \in \mathbf{X}^{\text{final}}$. Similarly, let $\mathcal{L}_q \subset \mathcal{L}$ be the set of class labels present in $\mathbf{X}_q$. Accuracy is the ratio between the number of identical labels of $\mathbf{X}_q$ and $\mathbf{X}_r$ and the total number of unique labels of $\mathbf{X}_q$ and $\mathbf{X}_r$. While precision is defined as the fraction of identical labels of $\mathbf{X}_q$ and $\mathbf{X}_r$ in the label set $\mathcal{L}_r$, recall is defined as the fraction of identical labels of $\mathbf{X}_q$ and $\mathbf{X}_r$ in the label set $\mathcal{L}_q$. The equations defining these metrics are given in Table III. According to their definitions, the retrieval performance is increased when the accuracy, precision and recall values approach to 1.

TABLE III: Performance metrics used to evaluate the performance of the proposed method. For two sets, $\cap$ denotes intersection and $\cup$ denotes union of the sets.

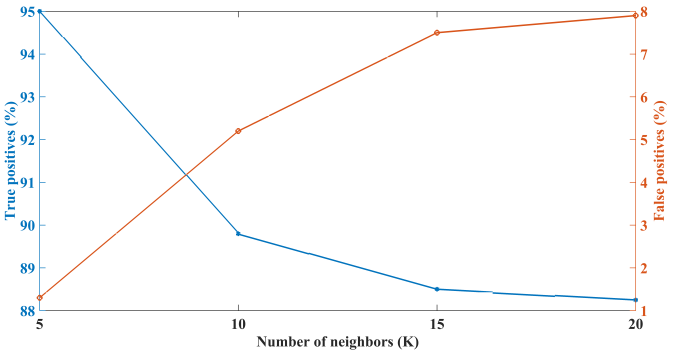
Performance metric	Formula
Accuracy	$\frac{1}{\lvert\mathbf{X}^{\text{final}}\rvert}\sum_{r=1}^{\lvert\mathbf{X}^{\text{final}}\rvert}\frac{\lvert\mathcal{L}_q\cap\mathcal{L}_r\rvert}{\lvert\mathcal{L}_q\cup\mathcal{L}_r\rvert}$
Precision	$\frac{1}{\lvert\mathbf{X}^{\text{final}}\rvert}\sum_{r=1}^{\lvert\mathbf{X}^{\text{final}}\rvert}\frac{\lvert\mathcal{L}_q\cap\mathcal{L}_r\rvert}{\lvert\mathcal{L}_r\rvert}$
Recall	$\frac{1}{\lvert\mathbf{X}^{\text{final}}\rvert}\sum_{r=1}^{\lvert\mathbf{X}^{\text{final}}\rvert}\frac{\lvert\mathcal{L}_q\cap\mathcal{L}_r\rvert}{\lvert\mathcal{L}_q\rvert}$

IV. EXPERIMENTAL RESULTS

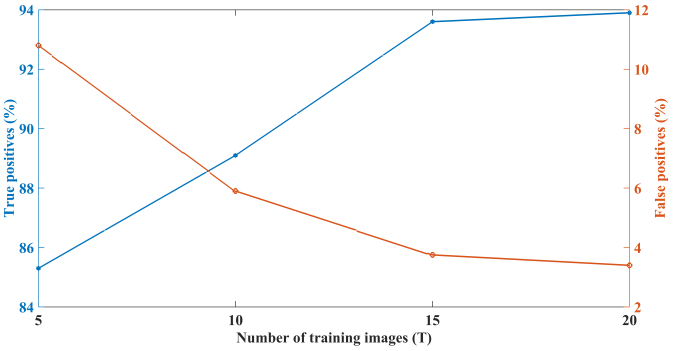
A. *Effects of varying the values of K and $|\mathbf{T}|$*

In order to study the effects of varying the values of K (number of neighbors of each image in the neighborhood graph $\mathcal{G}$) and $|\mathbf{T}|$ (number of training images) on the performance metrics, we have used two measures, namely false positives (FP) and true positives (TP), which have direct effect on the performance metrics. On one hand, the number of false positives for a particular class label $c \in \mathcal{L}$ is the number of images in the archive which are wrongly labeled as c, expressed as a percentage of images in the archive which do not have the label c. On the other hand, the number of true positives for a label c is defined as the number of images in the archive which are correctly labeled, expressed as a percentage of the total number of images in the archive having the label c. Both these values are calculated by taking the average over all the class labels $c \in \mathcal{L}$. Hence, the number of FP should ideally be very low and the number of TP should be very high. Fig. 7 shows the variation in the values of TP and FP versus variations of the values of the two parameters K and $|\mathbf{T}|$. From Fig. 7(a), one can observe that the number of TP decreases and the number of FP increases by increasing the value of K. This is because, as we consider a higher number of neighbors for an image, the chances of getting comparatively irrelevant images in the neighborhood increase, thereby degrading the system performance. On the other hand, Fig. 7(b) shows that the number of TP increases and the number of FP decreases by increasing the value of $|\mathbf{T}|$. The reason is that as we train

the system on a higher number of training images, the labeling of the unlabeled images becomes more accurate. However, the performances for $|\mathbf{T}| = 15$ and $|\mathbf{T}| = 20$ have been found to be quite similar. Considering these results, the values of $K = 5$ and $|\mathbf{T}| = 15$ have been found to give the best result in terms of accuracy of the multi-label categorization algorithm and are hence used in our experiments. It is important to note that the considered archive contains both large intra-category and inter-category variations in terms of region features, hence $|\mathbf{T}| = 15$ is a reasonable choice. A real archive is probably more complex than ours.



(a)



(b)

Fig. 7: Results of the TP and FP values obtained by the proposed method by varying (a) K and (b) $|\mathbf{T}|$.

B. Comparison of the overall retrieval performance with state-of-the-art methods

Table IV shows the values of accuracy, precision and recall obtained when the KNN, the ARGMM, the ML-SVM, the MC-SVM and the proposed MLIRM are used. These values are the average of the values obtained by considering each image in the archive as the query image and by retrieving the 20 most similar images. By analyzing the table, one can observe that the proposed MLIRM obtained significantly better metric values in the considered archive compared to the KNN, the ARGMM and the ML-SVM. As an example, the proposed method shows an improvement of 18.42% in accuracy and of 19.24% in precision over the ARGMM. The improvement over the KNN is even more remarkable and is of 42.44% in accuracy and of 34.03% in precision. The improvement in the recall value is of 11.6% over the ML-SVM and of 1.7% over the MC-SVM. These results are because of the ability of the proposed MLIRM to accurately model the image regions using the inherent multi-label information present in them. It is worth noting that the proposed method and the MC-SVM exploit the same region features, but the proposed method is more effective in terms of both computational time and retrieval accuracy. Further details regarding the computational efficiency are given in part E of this section.

TABLE IV: Results obtained for the KNN, the ARGMM, the ML-SVM, the MC-SVM and the proposed MLIRM.

Method	**Accuracy**(%)	**Precision**(%)	**Recall**(%)
KNN [4]	52.18	63.97	61.02
ARGMM [6]	63.56	72.34	69.87
ML-SVM [9]	67.39	76.52	71.93
MC-SVM [10]	73.94	85.17	78.91
Proposed MLIRM	**74.29**	**85.68**	**80.25**

Fig. 8 shows an example of images retrieved by the ARGMM, the MC-SVM and the proposed MLIRM when the query image is selected from the golf course category of the original archive. The retrieval order of each image is given above the related image and the multi-labels associated with the image are given below the related image. We would like to mention that the ARGMM and the MC-SVM are the best among the unsupervised and supervised methods, respectively, hence the retrieval results of the other two methods are not given.

The query image considered in Fig. 8(a) belongs to the golf course category and is associated with 3 primitive classes, namely *grass*, *trees* and *bare soil*. From the results one can see that all the images retrieved by the proposed MLIRM (see Fig. 8(d)) contain all the 3 primitive classes. On the contrary, the images retrieved by the ARGMM (see Fig. 8(b)) mostly contain 1 or 2 of the primitive classes only. For example, the 5^{th} image retrieved by the ARGMM originally belongs to the agricultural category of the UCMERCED archive. The results show how the label matching strategy solves the problem of spurious retrieval results due to feature matching. For the MC-SVM, the primitive classes present in the retrieved images (see Fig. 8(c)) are more or less the same as those present in the query image, but MC-SVM does not take into account the spatial arrangement of the regions in the images, which is evident from the retrieval results. Fig. 9 depicts another example of images retrieved by the ARGMM, the MC-SVM and the MLIRM related to a query image taken from the medium residential category. The retrieval results show that it is necessary to consider labels as well as the spatial arrangement of the regions in the query image in order to retrieve visually more similar images. By a visual analysis of all the obtained results, we can conclude that the proposed method accurately detects the multiple primitive classes associated with each query image and retrieves the visually most similar images from the archive.

C. Analysis of the effectiveness of the steps of the proposed method

1) Retrieval performance when only the first and second steps are executed: The multi-label image categorization step

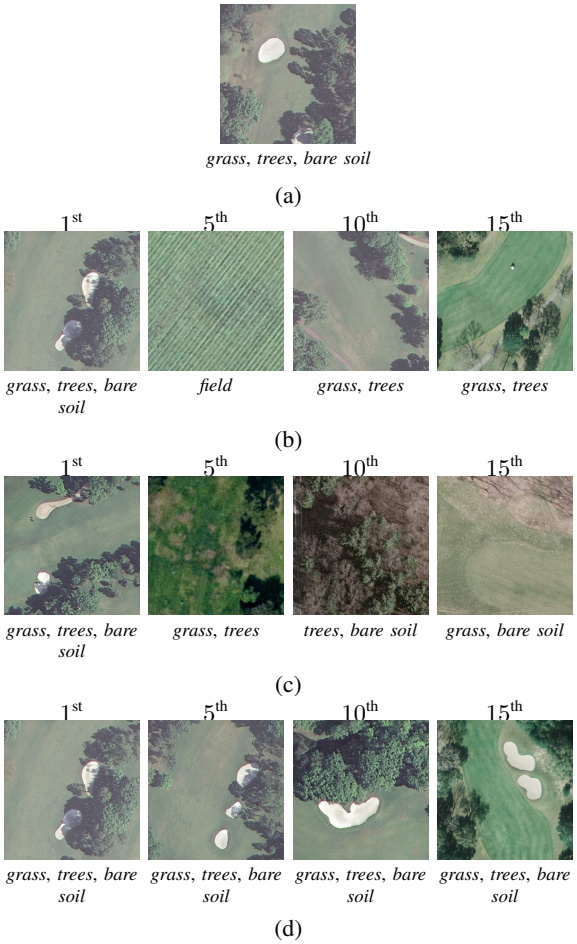


Fig. 8: Golf course image retrieval: (a) query image, (b) images retrieved by the ARGMM, (c) images retrieved by the MC-SVM and (d) images retrieved by the proposed MLIRM (multi-labels of each image are reported below the related image).

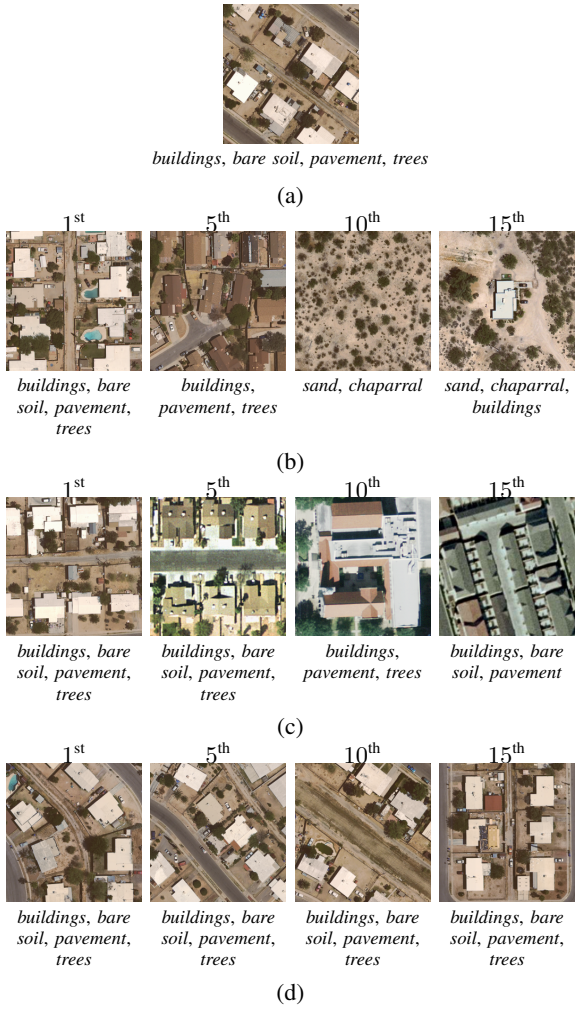


Fig. 9: Medium residential image retrieval: (a) query image, (b) images retrieved by the ARGMM, (c) images retrieved by the MC-SVM and (d) images retrieved by the proposed MLIRM (multi-labels of each image are reported below the related image).

of the proposed MLIRM aims to associate multiple class labels to each image in the archive based on the information obtained from the training images. After the second step, the query image $\mathbf{X}_q$ is also associated with multi-labels in the same way. It is to be noted that relevant images could be searched in $\mathbf{X}$ by merely matching the class labels of $\mathbf{X}_q$ with those of $\mathbf{X}_i \in \mathbf{X}$. The system would then retrieve the images having the same class labels as that of $\mathbf{X}_q$. However, in this case, the system would not be able to rank the retrieved images in the order of their similarity with $\mathbf{X}_q$. Fig. 10 shows an example of the top 3 retrieved images for a query image from the beach category, both by performing only the first two steps and by performing all the four steps. It can be observed that $\mathbf{X}_q$ as well as all the retrieved images in Fig. 10(b) and Fig. 10(c) contain the two class labels - *sand* and *sea*. Hence the values of the performance metrics used to evaluate the performance of the MLIRM are not affected much. However, in terms of visual similarity, the retrieved images in Fig. 10(c) are more similar to $\mathbf{X}_q$ in Fig. 10(a) than the retrieved images in Fig. 10(b). In the case when only the first and second steps are performed, the system detects the primitive classes *sand* and *sea* in $\mathbf{X}_q$ and accordingly retrieves all images in $\mathbf{X}$ containing the labels

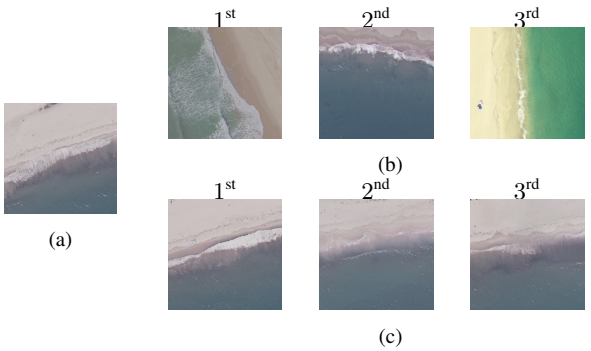


Fig. 10: Top 3 images retrieved by the MLIRM for a query image from the beach category. (a) query image, (b) images retrieved if only the 1st and 2nd steps are performed, (c) images retrieved if all the 4 steps are performed.

of *sand* and *sea*. But since the second step has no provision of ordering the images based on their similarity to $\mathbf{X}_q$, the images are retrieved in a random order. On the other hand, the steps of region labeling and graph matching in MLIRM computes the similarity of $\mathbf{X}_i \in \mathbf{X}^{\text{sub}}$ with $\mathbf{X}_q$ and retrieves the images in the order of similarity values. Thus, it is necessary to perform

all the four steps of MLIRM for achieving satisfactory retrieval performance.

2) Retrieval performance when only the first, third and fourth steps are executed: In the proposed method, the retrieval system initially detects the class labels present in the given query image and filters out the irrelevant images from the search space before proceeding to the region labeling and graph matching steps. This strategy reduces the computational time considerably, since it is no longer required to construct graphs and perform matching for all the images in the archive. If the multi-label categorization step is not executed, the system performs region labeling for all the images $\mathbf{X}_i \in \mathbf{X}$ instead of $\mathbf{X}_i \in \mathbf{X}^{\text{sub}}$ as proposed in the MLIRM. For labeling the regions of unlabeled images in that case, the system will compute $d(\cdot,\cdot)$ between the feature vector of a region and each of the characteristic feature vectors of all the class labels $c \in \mathcal{L}$. This may result in a region being wrongly labeled as a class which is not present in the image. In the proposed MLIRM, the multiple primitive classes associated with the images in the first step prevents this kind of wrong labeling, since each region is only assigned to one of those primitive classes which are present in the image. Hence, avoiding the second step will not only increase the computational time but also reduce the retrieval accuracy. Table V lists the values of the performance metrics obtained by performing the experiment with and without the multi-label categorization step. These values are the average of the values obtained by considering each image in the archive as the query image and by retrieving the 20 most similar images. As it is evident from the results, the retrieval performance deteriorates if multi-label categorization is not performed.

TABLE V: Values of the performance metrics obtained by executing the MLIRM with and without the 2nd step (multi-label categorization).

Proposed MLIRM	**Accuracy**(%)	**Precision**(%)	**Recall**(%)
With 2nd step	74.29	85.68	80.25
Without 2nd step	71.23	82.97	78.59

D. Retrieval results for a query image from outside the archive

We have also analyzed the efficiency of the MLIRM by giving as input to the system a query image that (i) does not belong to the considered archive, (ii) has a size different from those in the archive and (iii) is acquired by a different sensor having the same spectral characteristics of the archive images. For this experiment, we selected a patch from Google Earth (see Fig. 11(a)) as the query image such that the considered patch has a different size with respect to those of the archive images. We observed that the results obtained by the proposed retrieval system extract from the archive the images that are very similar to query image (see Fig. 11(b)).

From Fig. 11(b), one can observe that each of the retrieved images contains all or a subset of the primitive classes present in the query image in Fig. 11(a). The retrieved images are also ranked in the order of similarity between the images as determined by the subgraph matching algorithm. These results show that the proposed method works well as long as the spatial resolution and the spectral characteristics of the sensors used to acquire the images remain similar.

Fig. 11: Google map image retrieval: (a) query image and (b) top 10 images retrieved by the proposed MLIRM (multi-labels of each image are reported below the related image).

E. Analysis of the computational efficiency

The computational efficiency of the proposed MLIRM, the MC-SVM, the ML-SVM, the ARGMM and the KNN can be analyzed by considering the computational time required for each step of the methods. All the experiments are implemented via MATLAB® on a standard PC with Intel® Core™ 2.93 GHz i7 processor and 8 GB RAM. In our experiments, the MLIRM takes an average of 10 seconds to segment one image and about 7 seconds to extract features from the obtained regions. The average time required for the multi-label categorization step (step 2) is about 13 seconds (which is made up of 5.21 seconds required for neighborhood graph construction, 7.82 seconds for label propagation and 0.07 seconds for label score binarization). The computational time for these two steps depends on the size of the image archive, the size of the images and the number of training images used. It is worth noting that although segmentation and multi-label categorization of the images take considerable amount of time, they are performed offline prior to querying the retrieval system. The region labeling step (step 3) requires an average of 1.49 seconds for one image. Finally, an average of 7.78 seconds is taken to match the query graph with the graphs of all $\mathbf{X}_i \in \mathbf{X}^{\text{sub}}$ in step 4. The computational time for the fourth step depends on the size of the graphs and the size of the search space. The computational time taken by the KNN, the ARGMM, the ML-SVM, the MC-SVM and the MLIRM are given in Table VI. The time reported in Table VI for the first step of the KNN consists of extracting LPQ features from one

TABLE VI: Comparison of the computational time (in seconds) required by the KNN, the ARGMM, the ML-SVM, the MC-SVM and the proposed MLIRM.

KNN [4]			**ARGMM** [6]			**ML-SVM** [9]			**MC-SVM** [10]			**Proposed MLIRM**				
Step 1	Step 2	Total	Step 1	Step 2	Total	Step 1	Step 2	Total	Step 1	Step 2	Total	Step 1	Step 2	Step 3	Step 4	Total
2.65	0.46	**3.11**	30.00	81.75	**111.75**	42.35	1.25	**43.60**	75.70	2.20	**77.90**	17.00	13.00	1.49	7.78	**39.27**

image and that of the second step is due to the matching of the feature vector of $\mathbf{X}_q$ with the feature vectors of all $\mathbf{X}_i \in \mathbf{X}$. For the ARGMM, the computational time for step 1 accounts for the time required to model an image as an attributed relational graph. The computational time for step 2 is due to the time required to match G_q with all G_i in the archive. For both ML-SVM and MC-SVM, the first step includes the training of the independent SVM classifiers along with the cross-validation and the second step includes the prediction of the label vector of the query image and the retrieval of the similar images. Both these methods require comparatively larger amount of time because of training 17 classifiers. Note that the total time mentioned for each method in Table VI is merely the sum of the time required by the individual steps of the related method. The results show that in spite of the requirement to train the retrieval system, the proposed MLIRM is faster than the ARGMM. This is mainly due to the reduction of search space after the 2$^{\text{nd}}$ step in our proposed method. We would like to point out that although the KNN is the fastest of these five methods, it has the poorest retrieval performance compared to the other two (see Table IV).

V. CONCLUSION AND DISCUSSION

In this paper, we have introduced a semi-supervised graph-theoretic method in the framework of multi-label remote sensing image retrieval, which requires only a small number of training images characterized by multi-labels (associated to different land-cover classes). The proposed method consists of four main steps. The first step includes image segmentation and feature extraction from the segmented regions. The second step exploits the underlying label correlations of different land-cover classes and uses a semi-supervised graph-based algorithm to associate each image in the archive with multi-labels by propagating the label information from the training images to the unlabeled images. In the third step, few region labels of the training images are used to associate each region of the training images (and subsequently of the unlabeled images) with a particular class label. In the fourth step, these labels are used to create a region adjacency graph for each image, which are then used in the graph matching algorithm to compute image similarity.

In order to evaluate the proposed method, we re-defined a benchmark archive in RS CBIR problems by associating each image with a set of labels (instead of only a single label). Experimental results on this multi-labeled image archive demonstrate that our proposed method leads to significant performance improvement over the RS CBIR methods used in the comparisons. The main reasons for the efficiency of the proposed method are: 1) The second step exploits the information inherently present in RS images due to both the detailed level of semantic content associated to land-cover classes and their correlation at semantic level, which significantly improves the retrieval accuracy; 2) The strategy of pre-filtering before the third step and then performing finer search in a smaller search space considerably reduces the computational time required for the method; and 3) The final step of matching labeled graphs allows one to avoid spurious region matching based on features as was done in our previous unsupervised method. As a result, the images retrieved by the proposed method are found to be much more similar to the query image compared to the images retrieved by the state-of-the-art CBIR methods. We would like to point out that while we have used RGB aerial orthoimagery images in our experimental analysis, the proposed method can be used with any kind of remote sensing images (e.g. multispectral, hyperspectral images, etc.) by performing feature extraction for each spectral channel. The main drawback of the proposed method is that its accuracy is sensitive to the choice of the segmentation algorithm and the region features. Thus, these steps should be defined carefully by taking into account the properties of the images in the considered archive. Moreover, we also plan to apply fast scalable parallel algorithms in the context of multi-label categorization and subgraph matching in order to reduce the computational complexity.

As a final remark, we would like to point out that our method cannot identify a class in an image during multi-label categorization if it is not associated with any of the training images considered by the system. In our experiments, we have ensured that the system is trained on a sufficient number of training images for each possible class label by manually labeling the images in the archive. However, in case of partially labeled archive, it is critical to detect whether a few class labels are missing from the training set. In order to overcome this, as a future development of this work we plan to perform an active learning method through relevance feedback [8], where a classifier is trained on the currently labeled images and the images which are not confidently classified are returned as images to be labeled, thereby updating the label set associated with the archive.

APPENDIX A

See Table VII.

REFERENCES

[1] D. Peijun, C. Yunhao, T. Hong, and F. Tao, "Study on content-based remote sensing image retrieval," *IEEE International Geoscience and Remote Sensing Symposium*, vol. 2, July 2005.

[2] Y. Yang and S. Newsam, "Geographic image retrieval using local invariant features," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 2, pp. 818–832, February 2013.

[3] E. Aptoula, "Remote sensing image retrieval with global morphological texture descriptors," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 5, pp. 3023–3034, May 2014.

[4] M. Musci, R. Q. Feitosa, G. A. O. P. Costa, and M. L. F. Velloso, "Assessment of binary coding techniques for texture characterization in remote sensing imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 6, pp. 1607–1611, November 2013.

TABLE VII: TABLE OF SYMBOLS

Symbol	Description	Symbol	Description
$\mathbf{X}$	the archive of I images	$\mathbf{X}_i$	i^{th} image in the archive $\mathbf{X}$
$\mathbf{T}$	set of training images from the entire $\mathbf{X}$	$\mathbf{X}_t$	t^{th} image in $\mathbf{T}$
$\mathbf{X}^{\text{sub}}$	set of images selected after 1^{st} step	$\mathbf{X}_q$	query image
$\mathbf{X}_{t'}$	$t'-$ th unlabeled image in $\mathbf{X} \setminus \mathbf{T}$	$\mathbf{x}_t^p$	p^{th} labeled pixel of $\mathbf{X}_t$
$\mathbf{X}^{\text{final}}$	set of finally retrieved images	$\mathbf{X}_r$	r^{th} retrieved image in $\mathbf{X}^{\text{final}}$
$\mathcal{L}$	set of class labels for the archive	c	an element of $\mathcal{L}$
$\mathcal{L}_q$	set of class labels associated with $\mathbf{X}_q$	$\mathcal{L}_r$	set of class labels associated with $\mathbf{X}_r$
$\mathbf{L}_i$	label vector of image $\mathbf{X}_i$	l_i^c	c^{th} element of $\mathbf{L}_i$
$\tilde{\mathbf{L}}_i$	label score vector of image $\mathbf{X}_i$	$\tilde{l}_i^c$	c^{th} element of $\tilde{\mathbf{L}}_i$
r_i^k	k^{th} region of image $\mathbf{X}_i$	$\mathbf{f}_i^k$	feature vector modeling region r_i^k
$\mathcal{G}$	Neighborhood graph defined on the entire archive	$\mathbf{W}$	weight matrix of $\mathcal{G}$
K	number of neighbors of $\mathbf{X}_i$ in $\mathcal{G}$	n_i	number of regions in $\mathbf{X}_i$
$d(\cdot,\cdot)$	distance between two feature vectors	$\mathbf{N}_i$	set of images in the neighborhood of $\mathbf{X}_i$
$\mathbf{Y}$	matrix of label vectors	$\tilde{\mathbf{Y}}$	matrix of label score vectors
α_1 and α_2	weights used in computing edge weights $\forall\, G_i$	β	weight used for correlated label propagation
G_i	RAG for $\mathbf{X}_i$ defined as $G_i = (V_i, E_i, \mathbf{A}_i)$	$\mathbf{A}_i$	$n_i \times n_i$ weighted adjacency matrix of G_i
$c_{r_i^s}$	centroid of the pixel co-ordinates in region r_i^s	$\theta_{r_i^s}$	orientation angle of region r_i^s
$\mathbf{M}$	correspondence matrix used in SM algorithm	$\mathbf{y}$	indicator vector used in SM algorithm

[5] G. J. Scott, M. N. Klaric, C. H. Davis, and C. R. Shyu, "Entropy-balanced bitmap tree for shape-based object retrieval from large-scale satellite imagery databases," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 5, pp. 1603 – 1616, May 2011.

[6] B. Chaudhuri, B. Demir, L. Bruzzone, and S. Chaudhuri, "Region-based retrieval of remote sensing images using an unsupervised graph-theoretic approach," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 7, pp. 987–991, July 2016.

[7] S. Aksoy, "Modeling of remote sensing image content using attributed relational graphs," *IAPR International Workshop on Structural and Syntactic Pattern Recognition*, pp. 475–483, August 2006.

[8] B. Demir and L. Bruzzone, "A novel active learning method in relevance feedback for content-based remote sensing image retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 5, pp. 2323–2334, May 2015.

[9] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757 – 1771, 2004.

[10] R. Li, Y. Zhang, Z. Lu, J. Lu, and Y. Tian, "Technique of image retrieval based on multi-label image annotation," *IEEE Second International Conference on Multimedia and Information Technology (MMIT)*, vol. 2, 2010.

[11] M. L. Zhang and Z. H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognition*, vol. 40, no. 7, pp. 2038 – 2048, 2007.

[12] V. Ranjan, N. Rasiwasia, and C. V. Jawahar, "Multi-label cross-modal retrieval," *IEEE International Conference on Computer Vision*, December 2015.

[13] M. George and C. Floerkemeier, *Recognizing Products: A Per-exemplar Multi-label Image Classification Approach*. Springer International Publishing, 2014, pp. 440–455.

[14] G. Nasierding and A. Z. Kouzani, "Empirical study of multi-label classification methods for image annotation and retrieval," in *2010 International Conference on Digital Image Computing: Techniques and Applications*, Dec 2010, pp. 617–622.

[15] M. Wang and T. Song, "Remote sensing image retrieval by scene semantic matching," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 5, pp. 2874–2886, May 2013.

[16] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," in *IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, Oct 2005, pp. 1482–1489.

[17] M. Ben Salah, A. Mitiche, and I. B. Ayed, "Multiregion image segmentation by parametric kernel graph cuts," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 545–557, February 2011.

[18] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, pp. 55–67, January 2008.

[19] X. Li, X. Zhao, Z. Zhang, F. Wu, Y. Zhuang, J. Wang, and X. Li, "Joint multilabel classification with community-aware label graph learning," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 484–493, January 2016.

[20] M. Cho, J. Lee, and K. M. Lee, "Reweighted random walks for graph matching," *European Conference on Computer Vision*, 2010.

[21] T. Cour, P. Srinivasan, and J. Shi, "Balanced graph matching," *Neural Information Processing Systems*, vol. 20, no. 2, pp. 545–557, February 2006.

[22] M. Leordeanu, M. Hebert, and R. Sukthankar, "An integer projected fixed point method for graph matching and map inference," in *Advances in Neural Information Processing Systems 22*. Curran Associates, Inc., 2009, pp. 1114–1122.

[23] R. Zass and A. Shashua, "Probabilistic graph and hypergraph matching," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008.

[24] F. Long, H. Zhang, and D. Feng, "Fundamentals of content-based image retrieval," in *Multimedia Information Retrieval and Management*, ser. Signals and Communication Technology. Springer Berlin Heidelberg, 2003, pp. 1–26.

[25] X. Liu and D. Wang, "Texture classification using spectral histograms," *IEEE Transactions on Image Processing*, vol. 12, no. 6, pp. 661–670, June 2003.

[26] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[27] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[28] F. Li, Q. Dai, W. Xu, and G. Er, "Multilabel neighborhood propagation for region-based image retrieval," *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1592–1604, December 2008.

[29] S. Ayache, G. Quénot, and J. Gensel, "Classifier fusion for svm-based multimedia semantic indexing," in *Proceedings of the 29th European Conference on IR Research*. Springer-Verlag, 2007, pp. 494–504.

[30] F. Omruuzun, B. Demir, L. Bruzzone, and Y. Y. Cetin, "Content based hyperspectral image retrieval using bag of endmembers image descriptors," in *8th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Los Angeles, US*, 2016.