A Graph-Segment-Based Unsupervised Classification for Multispectral Remote Sensing Images

Nana Liu[1], Jingwen Li[1], Ning Li[2]
[1]School of Electronics and Information Engineering
[2]State Key Lab of Software Development Environment
Beihang University
XueYuan Road No.37, HaiDian District, Beijing
CHINA
liunana@ee.buaa.edu.cn lijingwen@buaa.edu.cn lining@nlsde.buaa.edu.cn

Abstract: - With more applications of multispectral remote sensing images, how to effectively and correctly make automated classification of multispectral images is still a great challenge. Utilizing both spatial contextual information and spectral information can achieve better classification performance. In order to make better utilization of the spatial contextual information, we apply graph model to the multispectral image, and use graph-based segmentation to produce units of pixels for further classification. In this paper, we present an unsupervised approach for multispectral remote sensing image classification with graph-based segment and fuzzy c-means clustering. Our method mainly involves following steps: First, represent image as graph $H=(V,E)$ based on the feature vector of per pixel and the relationships among neighboring pixels, and segment the graph into groups of sub-regions as basic object units using the effective graph segmentation algorithm. Then according to those global feature vectors of sub-regions, the fuzzy c-means clustering is used to obtain the classification map based on these sub-regions. Experiments shows the results by different segmentation scales, and then turn out that the approach proposed in this paper can achieve better accuracy and efficiency.

Key-Words: - Multispectral image; Segment-based classification; Graph-based segmentation; Fuzzy c-means; Unsupervised classification; Remote sensing images classification

1 Introduction

Multispectral images are the main type of images acquired by remote sensing radiometers, and are used commonly in remote sensing applications, especially concerning the earth environment. With the increasing number and size of multispectral images, how to automatically extract useful information effectively and accurately from multispectral images is becoming a great challenge. Multispectral Image Classification, as an important issue in multispectral image processing, is to identify and partition the pixels to its class (e.g. forest, building, bare land, etc.) according to the feature of the noisy image of an area. There are some common methods for multispectral image classification, such as statistical methods [1-3], neural network [1][4], fuzzy set theory[1], etc. According to the learning scheme, those classification algorithms can be broadly divided into two categories, supervised and unsupervised. The objective of supervised multispectral image classification methods is to identify and partition pixels to their corresponding classes. It is usually required to estimate parameters in the model using a large number of training samples (ground truths) [5]. The supervised methods can achieve better accuracy, but require the availability of a suitable data set of ground truths for the classification. But in many applications there are not enough ground training sets because of the expensive cost of sampling, therefore unsupervised methods are developed to perform classification in such cases [6][7]. For both supervised and unsupervised methods, the spectral signature is the main considered aspect used to classify those pixels, and traditionally one pixel is considered as the basic information union.

Nowadays more high-resolution multispectral images are taken by sensors such as IKONOS and QuickBird. The traditional pixel-based classification only utilizes spectral information, while the image data acquired by remote sensing radiometers have not only spectral attributes with correlated bands but also spatial attributes. Information contained in a high spatial resolution image is usually not represented by single pixel but by meaningful image objects, which include the association of multiple pixels and their mutual relations. Utilization of this spatial contextual information, in additional to spectral information, can improve the classification performance. In order to take account for the relationship of neighboring pixels, many efforts have been made on the spatial contextual classification [8-17]. Contextual classification procedures can be roughly divided into contextual classifiers and contextual re-classifiers, depending upon whether they use raw or classified data [15]. Many of the contextual classifiers are either based on the use of local windows [11][12] or consider the pixels at a given distance and/or direction from the pixel to be classified. Wu [5] presented method using the discrete wavelet transform for contextual classification. Object-based approaches were proposed in [14-17], and segment-based classification algorithms were used to take groups of neighboring pixels as basic units. Despite the significant achievements obtained recently, the segment-based classification is still limited on some crucial issues. How to take both contextual information and global feature into account well, and achieve better accuracy and lower complexity, is still a key problem for multispectral image classification.

In this paper we present an unsupervised segment-based approach for multispectral image classification. Our contribution is the application of graph-based segmentation combined with fuzzy c-means clustering. Taking the whole multispectral image as a graph, and using the graph-based segment can represent the contextual relationship of neighboring pixels intuitively. Firstly, represent the image as a graph based on pixels; Then graph-based segmentation algorithm is used to divide image into sub-regions as elementary object units; Based on these sub-regions, we compute the feature of each object unit; Finally, we use fuzzy c-means classifier, a typical unsupervised method, to obtain the classification map. The result shows that the whole approach can achieve smoother appearance, better accuracy and efficiency in comparison with pixel-based fuzzy c-means.

The organization of the paper is as follows: In section 2, we give the preliminary of graph-based segmentation and fuzzy c-means classifier; In section 3, we present the whole process for the multispectral image classification, and then give some details of graph-based segmentation and fuzzy c-means classification; In section 4, we propose the experiment for multispectral classification, show the result of our approach, and make a further comparison and discussion on those results; A conclusion is given in section 5.

2 Preliminary

Graph construction, graph segmentation, and fuzzy c-means classification are three key steps in our approach. Graph can be used to represent the relationship intuitively, and the segmentation on the graph divides the image into many sub-regions. Fuzzy c-means classifier then clusters these sub-regions into different classes. In this section, we give the preliminary of both the graph-based segmentation and fuzzy c-means classification.

2.1 Graph-based segmentation

Segmentation refers to the process of partitioning an image into multiple regions (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to be analyzed [18]. The result of image segmentation is a set of regions, which cover the whole image without overlap.

Graph-based segmentation is an important type of segmentation method. Normalized cut method was first proposed by Shi and Malik [19]. In graph-based approaches image segmentation is treated as a graph partitioning problem. The image is used to define a weighted undirected graph $H=(V,E)$. Every vertex $v_i \in V$ in H corresponds to a pixel of image P_i, and vertices v_i and v_j are connected by edge $(v_i,v_j)\in E$. The image is partitioned into disjoint sets by removing those edges, which connect some segments. The weight function $w((v_i,v_j))$ is used to measure the similarity between vertices v_i and v_j and it is generally based on the two feature vectors used in segmentation corresponding to v_i and v_j, and the distance between them. There are different ways to view edges, such as max-flow model, min-cut model, shortest path model, random walk model, etc

Felzenszwalb and Huttenlocher [20] gave an effective graph-based segmentation algorithm with better performance and accuracy. The effective graph-based segmentation method both captures perceptually important non-local image characteristics and is computationally efficient - running in $O(n\log n)$ time for n image pixel. An important characteristic of the method is its ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions. For these advantages of the method, we apply it for multispectral image segmentation. The following part mainly focuses on the introduction of this method.

The algorithm using local variation offers an effective and accurate method. Define S to be a segmentation of V, including the set of vertices C and the set of edges G for the given segmentation. C_i denotes the vertices of one sub-region after segmentation. The internal variation of a component $C \subseteq V$ is defined with the maximum weight edge in any minimum spanning tree of that component, that is

$$Int(C) = \max_{e \in MST(C,E)} w(e) \quad . \tag{1}$$

where $MST(C,E)$ is a minimum spanning tree of C with respect to the set of edges E. The external variation between two components to be the lowest weight edge connecting them,

$$Ext\left(C_1, C_2\right) = \min_{v_i \in C_1, v_j \in C_2} w\left(\left(v_i, v_j\right)\right) \tag{2}$$

Two components are similar, and should be merged into a single component, when the external variation between the components is small relative to both their internal variations,

$$Ext\left(C_1, C_2\right) \le MInt\left(C_1, C_2\right) \tag{3}$$

Where the minimum internal variation $MInt$ of C_1 and C_2 is,

$$MInt(C_1, C_2) = \min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2)) \tag{4}$$

The threshold function τ controls the degree to which the variation between two components must be greater than their internal variation in order to there to be evidence of a boundary between them. For small components, $Int\left(C\right)$ is not a good estimate of the local characteristics of the data. Therefore the threshold function is defined based on the size of the component as follows,

$$\tau(C) = k / |C| \tag{5}$$

where $|C|$ denotes the size of C, and k is some constant parameter. That is, for small components we require stronger evidence for a boundary. In practice k sets a scale of observation, in that a larger k causes a preference for larger components. However, that k is not a minimum components size. Smaller components are allowed when there is a sufficiently large difference between neighboring components.

The algorithm is given as follow:

Step1: Sort E into $\Pi = (e_1, ..., e_{num})$ by non-decreasing edge weight, where num is the number of edges in E.

Step2: Start with a segmentation G^0, where each vertex v_i is in its own component.

Step3: Repeat step 4 for $q = 1, 2, ..., num$.

Step4: Construct G^q given G^{q-1} as follows. onsidering e_q connects vertices v_i and v_j, if $C_i^{q-1} \ne C_j^{q-1}$ and $w(e_q) \le MInt(C_i^{q-1}, C_j^{q-1})$ then G^q is obtained from G^{q-1} by merging C_i^{q-1} and C_j^{q-1}. Otherwise $G^q = G^{q-1}$.

This segmentation algorithm is illustrated that it can produce segmentations that satisfy global properties [20]. It has the following advantages: first, it can capture perceptually important grouping or regions, which often reflect global aspects of the image; second, it is highly efficient, running in time nearly linear in the number of image pixels and also fast in practice.

2.2 Fuzzy c-means

Classical “hard” classification techniques are designed for application on phenomena which can be considered to exist in discrete class. Each pixel is simply allocated to the class with which it displays the greatest level of similarity. However, geographical information, included that remotely sensed, is imprecise in nature. For the allowance of natural fuzziness of such an environment, a fuzzy set algorithm may be used.

Classes, in which each member belongs in some extent to every cluster or partition, are called continuous classes [21]. Continuous classes are a generalization of discontinuous classes where the indicator function of conventional sets theory, with values 0 or 1, is replaced by the membership function of fuzzy sets theory, with values in the range of 0 to 1.

Fuzzy c-means clustering (FCM) is a clustering algorithm that provides a fuzzy partition of the input points [22-24]. According to the fuzzy clustering framework, each point has a degree of belonging to the clusters, as in fuzzy logic, rather than completely belonging to just one cluster. FCM partitions arbitrary n input points' vectors into c fuzzy groups, and finds a cluster center for each group such that a cost function of similarity measure is maximized, or dissimilarity measure is minimized. The measurement, for how much the vector belongs to that particular cluster, can be numerically represented by a $c \times n$ membership matrix $U = \left(u_{ij}\right)$, supposing that we take the partition of an unlabeled data set $X = \{x_1, x_2, ..., x_n\} \subset R^d$ in c classes. The sum of those coefficients is defined to be 1, as shown in following conditions:

$$\begin{cases} \sum_{i=1}^{c} u_{ij} = 1 & j = 1,...,n \\ \sum_{j=1}^{n} u_{ij} > 0 & \mathrm{i} = 1,...,\mathrm{c} \\ u_{ij} \in [0,1] & i = 1,...,c;\ j = 1,...,n \end{cases} \tag{6}$$

The fuzzy c-means method minimizes the objective function $J_m(U,V)$, which represents the within-class sum of square errors between classes:

$$J_m(U,V) = \sum_{j=1}^{n}\sum_{i=1}^{c} (u_{ij})^m d^2(x_j, v_i) \tag{7}$$

where $m \in (1,\infty)$ is a weighting exponent on each fuzzy membership; $v_i \in R^d$ is the cluster center of class i; $d^2(x_j, v_i)$ is the distance between x_j and v_i.

Formula (7) could be minimized by Picard iteration of the following equations [23]:

$$u_{ij} = \frac{\left(1/d^2(x_j, v_i)\right)^{1/m-1}}{\sum_{k=1}^{c}\left(1/d^2(x_j, v_k)\right)^{1/m-1}} \quad i = 1,\ldots,c;\ j = 1,\ldots,n \tag{8}$$

$$v_i = \frac{\sum_{j=1}^{n}(u_{ij})^m x_j}{\sum_{j=1}^{n}(u_{ij})^m} \quad i = 1,\ldots c \tag{9}$$

With these results, well known FCM algorithms are listed below:

Step 1: Choose the number of classes c, with $1 < c < n$.

Step 2: Choose a value for the fuzziness exponent m, with $m > 1$.

Step 3: Choose a definition for distance in the variable space.

Step 4: Choose a value for the stopping criterion.

Step 5: Initialize membership Matrix $U = U_0$.

Step 6: Calculate V using equation (9) and re-calculate U using equation (8).

Step 7: Compute $\max_{ij}\left|u_{ij} - \hat{u}_{ij}\right|$. Stop if it is below the stopping value; otherwise return to step 6.

Fuzzy c-means clustering provides a good framework for image analysis, and it has been applied widely in medical images analysis [25], image segmentation, etc. The fuzzy c-means has several advantages: it is unsupervised method; it can be used with any number of features and any number of classes; it distributes the membership values in a normalized fashion. It also provides a way to represent and manipulate the fuzzy data contained in images.

3 Proposed method

In this section we describe the details of our approach for multispectral image classification. It applies the graph model into multispectral image, and constructs a graph based on the pixels and their spatial relationships. Sub regions are generated by the whole image as basic object units, and different sub regions may belong to the same classification. Fuzzy c-means method is used for clustering based on these sub-regions. Fig.1 shows the process of the proposed algorithm, which mainly contains four steps as follows:

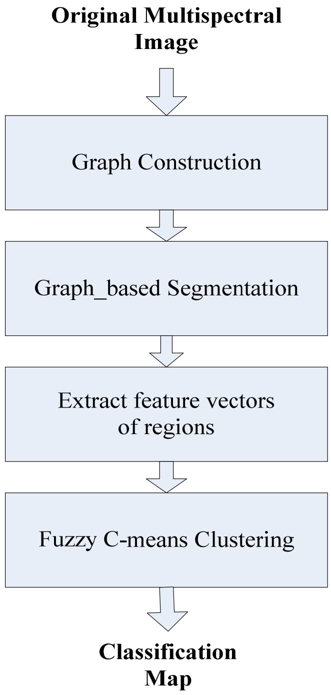

Fig.1: Process of proposed algorithm

1) Construct the weighted undirected graph $H = (V, E)$ based on original multispectral image. Each pixel is considered as a vertex of the graph, and the spatial relationship is presented by the edges. All the edges are added according to the given distance L between pixels during initialization.
2) Segment the image into different regions without overlap. Based on the Graph, an effective graph-cut based segmentation is used to obtain an over segmentation. Over

segmentation means that if two components belong to the same class but not connected spatially, these components will be considered as two different segments. Over segmentation is to get the neighboring pixels with similar features as a single unit.

3) Extract feature vector for each sub region generated by segmentation. We use the mean vector to represent each sub-region. Take these mean vectors as the input of fuzzy c-means clustering for unsupervised classification of sub-regions.
4) Classify the sub-regions based on mean vectors using fuzzy c-means clustering. Then label every sub-region with its mean vector's class label.

3.1 Graph-based segmentation

Segmentation is an important part of the whole approach, and it divides the whole multispectral image into different sub regions without overlap. The segmentation is based on graph model which is constructed to represent the multispectral image. The multispectral image is defined as a weighted undirected graph $H = (V, E)$. Every vertex $v_i \in V$ in H corresponds to a pixel of the image, and vertices v_i and v_j under the distance L are connected by edge $(v_i, v_j) \in E$.

Let $f(x, y) = (f_1(x, y), f_2(x, y), ..., f_d(x, y))^T$ be the feature vector corresponding to each pixel of multispectral image, where d is the number of spectral bands. As introduced in section 2.1, each vertex $v_i \in V$ corresponds to a pixel in the image, so the feature vector, relating to each vertex of graph, can be denoted by $f(v_i) = (f_1(v_i), f_2(v_i), ..., f_d(v_i))^T$. In order to describe the dissimilarity of neighboring pixels in multispectral image, we define the weight function. The weight function of edge is used to measure the dissimilarity degree between the vertices:

$$w((v_i)) = \begin{cases} \sqrt{\sum_{k=0}^{d} \left| f_k(v_i) - f_k(v_j) \right|^2} & (v_i, v_j) \in E \\ \infty & \text{otherwise} \end{cases} \quad (10)$$

where $E = \{(v_i, v_j) \mid \|P_i - P_j\| \le L\}$ is the set of edges for the distance between vertices which are under given distance L. In this paper, we use 8-connectd neighborhood, that means $L = 1$. Edges between two vertices in the same region should have relatively low weights, and edges between vertices in different regions should have higher weights.

Define $S = (C, G)$ to be a segmentation of image, including the set of vertices C and the set of edges G for the given segmentation. The algorithm is based on the method mentioned in section 2.1. Let G^q denote the subset of $G \subseteq E$ obtained with the first q edges in Π. Similarly S^q is used to denote the segmentation relevant to G^q. C_i^q is the set of all the vertices in one component of S^q. $MInt(C_1, C_2)$ is the minimum internal variation of C_1 and C_2. The segmentation process for the graph is as follows:

1) Construct E based on multispectral image pixels, and calculate the weight for each edge. Then sort E into $\Pi = (e_1, ..., e_{num})$ by non-decreasing edge weight, where num is the number of edges in E.
2) Initialize $G^0 = \{\}$ and $e_q = e_1$.
3) Compute G^q given G^{q-1} as follows. Considering e_q connects vertices v_i and v_j, If v_i and v_j are in disjoint components of G^{q-1} and $w(e_q)$ is small compared to the internal difference of both those components, then merge the two components otherwise do nothing. More formally, if $C_i^{q-1} \ne C_j^{q-1}$ and $w(e_q) \le MInt(C_i^{q-1}, C_j^{q-1})$ then $G^q = G^{q-1} \cup \{e_q\}$ $C_i^q = C_j^q = C_i^{q-1} \cup C_j^{q-1}$ else $G^q = G^{q-1}$.
4) Repeat step 3) for $e_q = e_2, e_3, ..., e_{num}$

The threshold function $\tau(C) = k / |C|$ controls the degree to which the variation between two regions must be greater than their internal variation. k is the only one runtime parameter for the algorithm, and is used to compute the threshold. k effectively sets a scale of observation, and a large k causes a preference for larger regions. The size of regions directly effects the result of classification, that means larger regions caused by large k may ignore some detail information of the image meanwhile filter some noisy. So the choice of k is very important in the approach. Experiments and further discussions are given in section 4.2.

The algorithm generates an over segmentation for the multispectral remote sensing image. Over