Multi-Scale Segmentation Method Based on Binary Merge Tree and Class Label Information

TENGFEI SU [ID] **AND SHENGWEI ZHANG**

College of Water Conservancy and Civil Engineering, Inner Mongolia Agricultural University, Hohhot 010018, China

Corresponding author: Shengwei Zhang (zsw@imau.edu.cn)

ABSTRACT This paper presents a newly developed segmentation algorithm for high-resolution remote sensing imagery. The new method is based on binary merge tree (BMT) and mainly comprises two parts, which are BMT construction and tree node selection. The first part is initialized by super-pixels, which are iteratively merged through using the well-known multi-resolution segmentation approach to complete BMT construction. The primary contribution of this paper resides in the second part, in which a new tree node selection criterion is developed. The new criterion is formulated by using spectral variance and class label cues in order to better describe intra-segment homogeneity. To validate the proposed strategy, four multi-spectral scenes captured by two Chinese satellites, ZiYuan-3 and GaoFen-2, were adopted for segmentation experiments. The results indicated that our method out-performed some state-of-the-art algorithms in terms of segmentation accuracy.

INDEX TERMS Binary merge tree, class label, image segmentation, region merging.

I. INTRODUCTION

The attention paid to the application of object-based image analysis (OBIA) has been growing during recent years [1], [2], due to the good feasibility of integrating region-based semantic features into the interpretation of remotely sensed images. Almost all types of remote sensing image data, including optical and radar scenes with various spatial and radiometric resolutions, have been utilized to test the potential of OBIA, such as agricultural landscape mapping [3]–[5], urban area classification [6], [7] and land-use/land-cover change detection [8], [9]. Most previous studies pointed out that the first and foremost step of OBIA is image segmentation [2], [3], [8]. Although it is often considered as an old and extensively studied problem in computer vision, efforts are still needed to better solve this issue when it comes to earth observation data, because the miscellaneous landscapes and geo-objects often seriously complicate the segmentation process.

The existent remote sensing image segmentation approaches can be coarsely categorized into two types, according to whether class label information is used. This kind of information is very important, since they are directly related to class-specific feature extraction and utilization, which can be considered as key elements for the problem of pattern recognition in the scenario of remote sensing. The methods incorporating labels in their models can provide detailed statistical information for different classes, so that geo-objects with confusing spectral appearances can be well singled out. However, considering that most image segmentation approaches are unsupervised, how to reliably assign class label to each pixel or super-pixel or segment in the initialization step becomes a key issue. In many cases, Markov random fields (MRF) and its variants are decent techniques to provide class labels in an unsupervised manner. The MRF-based segmentation algorithms are very popular and widely applied in remote sensing community [10]–[12], but these methods generally require many input parameters, and most of them are application-dependent and should be empirically or experimentally set by users. Some studies [13], [14] even indicated that prior knowledge is necessary for setting parameter(s) of some MRF models, which, in terms of operational applications, cannot be deemed as a simple task.

Many of the approaches not exploiting label cues are based on region-growing/merging strategies, in which geo-objects can be segmented out by aggregating or merging similar small segments. Practically, the aggregation of two segments

is controlled by a scale parameter which only allows the merging when the similarity measure is lower than the scale value. These methods are intrinsically hierarchical [15], since they can create a series of segmentation results by using different scales, and the segments in small scale result can be considered as components of a segment in large scale result, so that a hierarchical relationship can be found for the segments in results of different scales. A number of well-established algorithms, such as multi-resolution segmentation (MRS) [16], hybrid region-merging algorithm [17], boundary-constrained region-merging approach [18], hierarchical segmentation (HSeg) and refined HSeg [19], fall into this category. However, in the framework of these methods, how to efficiently and accurately determine the scale threshold used as the termination criterion of region-merging is often too difficult. Users have to run the algorithm for tens or even hundreds of rounds to find the ideal scale, which is often too time- and labor- consuming.

To solve the aforementioned problem, some researchers established unsupervised scale selection approaches [20]–[22], which help the user automatically estimate the optimal scale value. Although such methods can save a great deal of human work, they are still often inefficient since the segmentation method has to be executed for a large amount of time to produce a series of segmentation results, which are analyzed and compared to find the optimal scale.

Aside from the efforts on scale selection, there is an alternative, which is based on binary partition tree (BPT) or binary merge tree (BMT) [23], [24]. As can be inferred from the name, BMT reflects the binary nature of region-merging process, because for each merging, two segments are involved to form a larger one. Considering that BMT is a binary-tree-based data structure in which each tree node is associated with a segment, it is easy to understand that this model can be used to record the whole region-merging process. As exemplified by Fig. 1, the root node of BMT is a segment composed of the entire image, while each leaf node corresponds to an initial segment, which can be considered as either a single

pixel or a super-pixel, according to the initialization strategy adopted in implementation. In this framework, scale selection can be easily transformed into a tree node selection problem. For example, if the image shown in Fig. 1 contains two real geo-objects consisting of super-pixels a, b, c for the first and d, e for the second, the two intermediate nodes, symbolized as 'a, b, c' and 'd, e', should be chosen.

A recent study [25] attempted to solve the node selection problem by using supervised machine learning algorithms. This is an interesting idea and has been well-validated, but it is not an unsupervised strategy and a large number of training images are needed to achieve good performance. Also, the preparation of the training images is considered labor-costly, especially in the case of remote sensing images. A similar technique was proposed in an unsupervised manner and it was based on graph cut algorithm [26]. This kind of approach is also called pruning, in which it is implemented by firstly adding two nodes, including a source node connected to the root, and a sink node linked to all of the leaf nodes. Then, pruning is performed by cutting the edges between the selected nodes and their descendents, according to some graph-cut-based metrics. Although the aforementioned two types of node selection methods can help to improve the segmentation accuracy for some remotely sensed images, they all ignore label information, the significance of which has been previously described in this introduction.

This work presents an effort to further improve segmentation performance for remote sensing images by taking advantage of label cues and BMT-based techniques. The primary contribution resides in a new unsupervised node selection criterion, which not only considers spectral features of a tree node, but also a class-label-based entropy measure.

The structure of this article is as follows. The proposed framework is detailed in section 2. Section 3 describes the preparation of validation experiment, followed by the experimental results in section 4. Discussion and concluding remarks are given in section 5 and 6, respectively.

II. PRINCIPLE OF THE PROPOSED SEGMENTATION METHOD

A. OVERALL PROCESS

The whole processing chain of the proposed approach is illustrated in Fig. 2. As can be seen from this figure, there are four main steps, the first one of which is initial over-segmentation used to generate super-pixels. The reason for adopting super-pixel instead of pixel as the initial processing unit is that this strategy significantly reduces memory requirement and computational burden for the subsequent procedures with almost no cost on final segmentation performance.
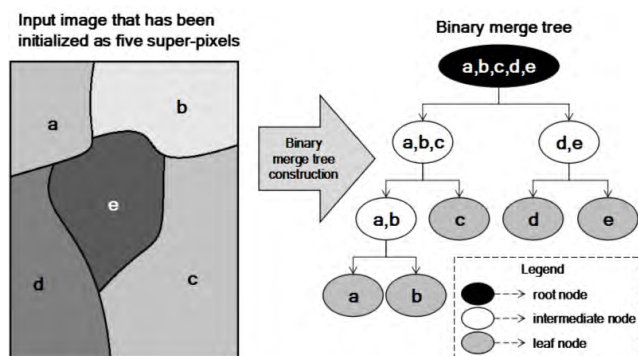
FIGURE 1. A simple illustration of binary merge tree (BMT). The left is a sample image that is initialized as 5 super-pixels (a, b, c, d, e). The right is a BMT corresponding to the left image. In BMT, each leaf node represents a super-pixel, as indicated by the labels (a - e). Note that the colors of leaf nodes do not reflect the grayscale values of their corresponding super-pixels.
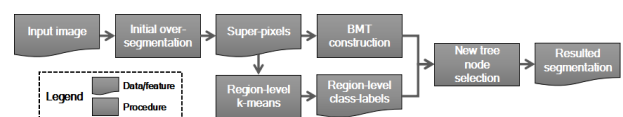
FIGURE 2. Flow chart of the proposed segmentation algorithm.
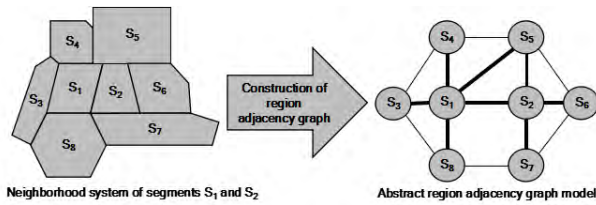
FIGURE 3. Illustration of a neighborhood system of two segments S1 and S2 and the corresponding abstract region adjacency graph model. The latter one can be used to delineate mutual-best-fitting (MBF) rule: only if the edge value of S1 and S2 is the lowest among all of the thickened edges, S1 and S2 are considered to fulfill MBF.

Since the errors in the initialized super-pixels will be unavoidably passed onto the resulted segmentation through intermediate steps, the selection of over-segmentation method should be very discreet. In this study, a seeded-region-growing-based algorithm [27] is chosen due to its decent performance of inter-geo-objects boundary preservation. The parameter setting for this method is detailed in section 3.

Based on super-pixels, a region-level k-means clustering is developed to extract class label information that is to be used for the proposed node selection approach. The super-pixels are also used to construct binary merge tree (BMT), in which the multi-resolution segmentation (MRS) [16] is employed to implement binary region-merging, and more details are provided in the following sub-section.
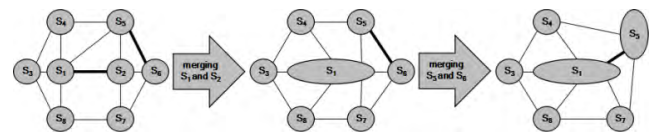
B. BMT CONSTRUCTION

Suppose that there are M_1 initialized super-pixels for the input image I, the MRS approach based on global mutual best fitting (GMBF) rule [16] is adopted and implemented to construct BMT. In the framework of GMBF, the word 'mutual' is a very important notion that deserves detailed description. To understand GMBF, it is necessary to first introduce mutual-best-fitting (MBF) rule [16]. Considering two segments S_1, S_2, and all of their neighboring segments, as illustrated in the left part of Fig. 3. This neighborhood system of S_1 and S_2 can be abstracted as a region adjacency graph with each node representing a segment and each edge indicating spatial bordering relationship between any two segments. Note that each edge can be given a value describing the similarity or fitness-of-merging for the two segments connected by the edge. In the right part of Fig. 3, the thickened edges originate from either S_1 or S_2, and among all of such edges, if the one linking S_1 and S_2 has the lowest similarity value (suppose lower similarity value indicates more suitable for merging), then these two segments can be deemed to meet MBF rule, and a pair can be formed by such two segments. Then, it is simple to delineate GMBF rule: for the M_1 segments/super-pixels, find all of the pairs fulfilling MBF; if there are M_2 pairs (M_2 can be no larger than M_1, because in a few cases, a segment cannot be paired with one of its neighbors according to MBF), order them by the similarity value (lowest first), and the first one is prioritized for merging.

TABLE 1. The detailed steps of Bmt construction.

Input: initialized super-pixels
Output: a binary tree structure used to represent BMT
1. For all of the M_1 initial super-pixels, create a tree node for each one, and store these tree nodes in a linear list *tree_list*; note that all of these tree nodes generated in this step are leaf nodes of BMT;
2. Based on the initialized segmentation state, find all of the segment pairs according to MBF rule;
3. Create a linear *list* by all of the segment pairs found in step 2, and then sort them in ascending order;
4. If *list* is not empty, go to step 5, otherwise go to step 10;
5. Take the first element of *list*, and merge the two segments (S_1 and S_2) of this element to yield a new segment S_{new}, then remove this element from *list*;
6. Find the two tree nodes corresponding to S_1 and S_2 mentioned in step 5 from *tree_list*; create a new tree node for S_{new}, and let the new tree node be the parent node for the two tree nodes of S_1 and S_2; add the new tree node to *tree_list*;
7. For S_{new} created in step 5, check: can it be paired according to MBF rule? if the answer is yes, go to step 8, otherwise go to step 9;
8. Insert the newly found segment pair generated in step 7 into *list*, and make sure the *list* elements are still in ascending order;
9. Go back to step 4;
10. Return *tree_list* as the resulted BMT.

FIGURE 4. An illustration of two merging steps. Different from those in Fig. 3, the thickened edges in this figure indicate a segment pair found by MBF.

In our GMBF-based BMT construction, a linear *list* with length of M_2 is used to store all of the segment pairs. Each element of *list* consists of three variables: the pointers of S_1 and S_2, and the similarity measure of the two segments. Thus, an element only takes up 12 bytes since pointer variable and floating point variable are both 4-byte long. At the beginning of BMT construction, the elements of *list* are sorted by similarity value, so that all of the elements are in ascending order.

The whole procedures of BMT construction is given in table 1. From these steps, it is worth mentioning that steps 7 and 8 are very necessary procedures that enable the completion of BMT construction. Because as the loop created by steps 4 and 9 proceeds, new segments are formed and they should be iteratively merged for the sake of BMT growing, until the whole image is merged as a single segment corresponding to the root node of BMT. To better understand this process, Fig. 4 shows a vivid demonstration of two merging steps. In this example, it is assumed that the similarity value of S_1 and S_2 is lower than that of S_5 and S_6, so the merging of the former pair is prior to the latter. After the second merging, two newly created segments (the S_1 and S_5 in the right-most part of Fig. 4) are found to fulfill MBF, so that they are to be merged subsequently.

In the above described approach, the selection of similarity measure is a key issue, because it can produce nontrivial effects on the structure of BMT, which strongly affects the

segmentation performance of the proposed method. In our implementation, the heterogeneity-based criterion used by MRS [16] is adopted, mainly due to that this criterion considers not only spectral but also geometric features, so that segments of visually pleasing polygonal shapes can be easily created.

According to the heterogeneity-based criterion, the similarity or suitability of merging for any two neighboring segments (S_1 and S_2) is quantified via estimating the heterogeneity change induced by their merging,

$$\Delta H = \omega_{shape} \cdot \Delta H_{shape} + (1 - \omega_{shape}) \cdot \Delta H_{spectra} \quad (1)$$

where ΔH, ΔH_{shape}, $\Delta H_{spectra}$ means total heterogeneity change, geometric heterogeneity change and spectral heterogeneity change, respectively; ω_{shape} represents the weighting factor for geometric heterogeneity change, and it ranges from 0 to 1. ΔH_{shape} and $\Delta H_{spectra}$ are defined as follows,

$$\Delta H_{shape} = \omega_{compactness} \cdot \Delta H_{compactness} + (1 - \omega_{compactness})$$
$$\cdot \Delta H_{smoothness} \quad (2)$$

$$\Delta H_{spectra} = \frac{1}{B} \sum_i^B (\sigma_{new,i} \cdot n_{new} - \sigma_{1,i} \cdot n_1 - \sigma_{2,i} \cdot n_2) \quad (3)$$

where $\Delta H_{compactness}$ and $\Delta H_{smoothness}$ are two components for the measurement of ΔH_{shape}; $\omega_{compactness}$ is a weighting factor balancing the relative contribution of $\Delta H_{compactness}$ and $\Delta H_{smoothness}$ to ΔH_{shape} and it also ranges from 0 to 1; B indicates the number of image channels; $\sigma_{new,i}$ represents spectral standard deviation of ith channel for the new segment (S_{new}) created by hypothetically merging S_1 and S_2; $\sigma_{1,i}$, $\sigma_{2,i}$ are spectral standard deviation of ith band for S_1, S_2, respectively; n_{new}, n_1, n_2 means the number of pixels included by S_{new}, S_1, S_2, respectively. The definitions of $\Delta H_{compactness}$ and $\Delta H_{smoothness}$ are given below,

$$\Delta H_{compactness} = \sqrt{n_{new}} \cdot p_{new} - \sqrt{n_1} \cdot p_1 - \sqrt{n_2} \cdot p_2 \quad (4)$$

$$\Delta H_{smoothness} = n_{new} \cdot \frac{p_{new}}{l_{new}} - n_1 \cdot \frac{p_1}{l_1} - n_2 \cdot \frac{p_2}{l_2} \quad (5)$$

where p_{new}, p_1, p_2 symbolize the perimeter length of S_{new}, S_1, S_2, respectively; l_{new}, l_1, l_2 indicate the perimeter length of the bounding box for S_{new}, S_1, S_2, respectively.

C. TREE NODE SELECTION SCHEME

With a fully established BMT, image segmentation can be achieved by selecting appropriate tree nodes. It should be noted that, for all of the selected tree nodes, their corresponding segments must not have overlapped portions and their union must equal the area of the entire input image. These two conditions can be mathematically formulated as: suppose the selected tree nodes compose a set $\mathbf{D} = \{E_1, E_2, \ldots\ldots, E_{M-1}, E_M\}$ with each element E_i representing a tree node, then

$$\forall E_i, E_j \in D, E_i \notin Descendant(E_j) \text{ and}$$
$$E_j \notin Descendant(E_i), \quad i \neq j \quad (6)$$

$$area(I) = \sum_i^M area(E_i) \quad (7)$$

TABLE 2. Leaf-starting tree node selection algorithm.

Input: the *tree_list* generated by the approach in table 1
Output: a set of selected tree nodes
1. Create a list called *result_list* to store the selected tree nodes;
2. Pick up all of the leaf nodes from *tree_list* and use them to form a new list named *leaf_list*;
3. If *leaf_list* is not empty, go to step 4, otherwise go to step 9;
4. Take and then remove the first element from *leaf_list*, and denote this tree node as *current_node*;
5. If *current_node* is a descendant of any tree node in *result_list*, go back to step 3, otherwise go to step 6;
6. If *current_node* is root node, go back to step 3, otherwise go to step 7;
7. Check whether *current_node* meets the tree node selection criterion; if the answer is yes, insert it into *result_list* and go back to step 3, otherwise go to step 8;
8. Let *current_node*'s parent node be the new *current_node*, and go back to step 6;
9. Return *result_list*.

TABLE 3. Root-starting tree node selection algorithm.

Input: the *tree_list* generated by the approach in table 1
Output: a set of selected tree nodes
1. Create a list called *result_list* to store the selected tree nodes;
2. Create two lists to store two neighboring levels of tree nodes, and name one of them as *current_level* and the other as *next_level*; initialize the *current_level* by inserting the root node into it;
3. If *current_level* is empty, go to step 6, otherwise go to step 4;
4. Process all of the elements of *current_level* in a one-by-one order: for a tree node, check whether it meets the tree node selection criterion, if the answer is yes, insert it into *result_list*, otherwise add its two children nodes into *next_level*;
5. Switch *current_level* and *next_level*, and go back to step 3;
6. Return *result_list*.

where $Descendant(E_i)$ indicates a set of tree nodes made up of all the descendant nodes of E_i; $area(\cdot)$ symbolizes the areal value of a segment, and it is measured by the number of pixels belonging to the segment.

It is difficult to implement a node selection approach simultaneously fulfilling equations (6) and (7). However, it is still straightforward to finish this task by using two strategies: the first one starts from leaf nodes while the second begins from the root. The leaf-starting and root-starting schemes are demonstrated by the pseudo code in table 2 and table 3, respectively. Note that the adopted tree node selection criterion mentioned in tables 2 and 3 is detailed in sub-section II.D, since this sub-section mainly focuses on the primary steps of tree node selection.

After a careful analysis of the contents listed in table 2 and 3, it is clear that the leaf-starting strategy tends to consume less memory than its counterpart, because the root-starting method requires the construction of two linear lists (*curretn_level* and *next_level*), while the leaf-starting one only establishes one linear list (*leaf_list*), given that the length of the three lists should all be initialized as the number of leaf nodes. However, the leaf-starting strategy is apparently more complicated to implement, since there are 9 steps in table 2 as compared to 6 ones in table 3. Another drawback of the leaf-starting approach is that it may lead to incorrectness of the selected tree nodes, which is illustrated in Fig. 5. This case shows that a tree node is firstly selected in step t, but
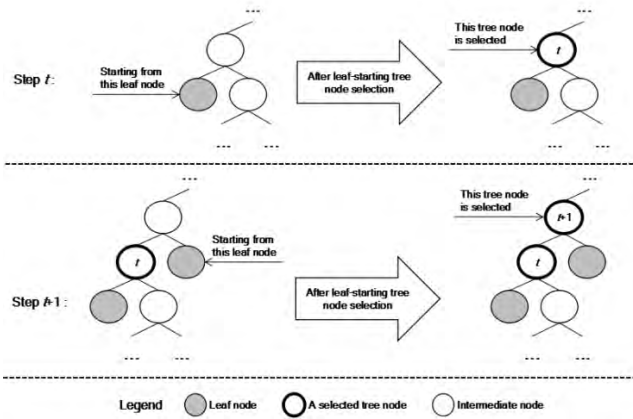
FIGURE 5. An illustration of a type of error that may occur in leaf-starting tree node selection. This example demonstrates that the selecting process of step t is incorrect, since the selected tree node t is a child node of the selected tree node in step t+1, causing breach of equation (6).

in step $t + 1$, its parent node is selected, which not only wastes computational time, but also breaches equation (6). On the contrary, the root-starting scheme can fully avoid this kind of situation, because this strategy will firstly select tree node symbolized as $t + 1$ in Fig. 5, and since its two children will not be included in *next_level*, they will not be considered in the next round of loop, not to mention selecting them.

By considering the aforementioned pros and cons of the two schemes, the root-starting strategy is chosen to implement the proposed segmentation algorithm, due to its superiority in being less complicated to implement as well as being more efficient in execution.

After the tree node selection procedures, for any selected tree node, all of the super-pixels corresponding to its descendant leaf nodes are merged as one segment. In this way, the final segmentation result is achieved.

D. TREE NODE SELECTION CRITERION

The proposed tree node selection criterion integrates both spectral as well as label information to effectively delineate intra-segment homogeneity. Before the introduction of this new criterion, it is necessary to first provide the definition of segment energy used to quantitatively measure the homogeneity of a segment,

$$E = \frac{1}{B} \sum_b^B \sigma_b \cdot (1 + Q) \qquad (8)$$

where E indicates the homogeneity energy of a segment; B means the number of image bands; σ_b represents a segment's spectral standard deviation of bth band; Q is class label entropy, and is defined as,

$$Q = - \sum_c^L p_c \cdot \log_2 (p_c) \qquad (9)$$

where L is the number of classes; $\log_2(\cdot)$ is a logarithm operator with 2 being its base; p_c symbolizes the probability of a segment belonging to class c, and it is estimated by

$$p_c = \frac{n_c}{n} \qquad (10)$$

where n_c is the number of a segment's pixels with class label c; n is the total number of pixels of a segment, thus $n_c \leq n$ and $p_c \leq 1$.

From equation (8), it can be easily observed that E is expected to be low valued for a homogeneous segment, because such a segment's spectral variance is small and the pixels within it tend to have the same class label. Moreover, when $L = 1$, E is purely dependent on spectral information, and in this situation the proposed tree node selection criterion degenerates to a traditional strategy relying solely on spectral feature.

Based on the model of (8) and the approach detailed in table 3, the proposed tree node selection method can be completed through adopting the following test:

$$T = \begin{cases} 1 & E < Scale \cdot E_{\text{root}} \\ 0 & \text{otherwise} \end{cases} \qquad (11)$$

where E indicates the homogeneity energy of the segment being tested and it is calculated by using equation (8); E_{root} is the energy value of the root node; $Scale$ is a scale parameter with range of (0,1). According to equation (11), a tree node is selected if its test result $T = 1$.

With the criterion devised by equation (11), it is simple to understand that large $Scale$ tends to result in under-segmentation, since high-valued $Scale$ will make the resulted segments have relatively low degree of homogeneity (and thus high-valued E). By analogy, low-valued $Scale$ is more likely to produce over-segmentation. Consequently, the selection of $Scale$ has a serious influence on the performance of the proposed segmentation, and it is thus analyzed in depth in the section of experimental result.

III. EXPERIMENT SETUP

A. DATASET

In order to validate the proposed segmentation method, four scenes of high resolution multi-spectral remote sensing images are adopted for experiment, as can be seen in Fig. 6. The first two images are acquired by the Chinese ZiYuan-3 satellite launched on Jan 9 2012. The other two images are captured by a newer satellite, GaoFen-2, which was launched on Aug 19 2014. Both mission aimed at providing high quality and high resolution images for civil applications, such as surveying natural resources, mapping urban and agricultural landscapes, and monitoring forested areas. The four scenes are termed as T1, T2, T3, T4 in the following texts for convenience. Their basic information can be seen in Table 4. As for the location information of the two images, T1 is situated at the southern part of Zhengzhou city, the capital of Henan province, China; T2 primarily covers the northern outskirt of Shenzhen city, a very important economic center in Guangdong province, China; T3 is located at the north-western part of Beijing, the political center of China; T4 displays a part of eastern Guangzhou, an important economic center in northern China. The four scenes have very different landscape patterns: T1 exhibits an urban district
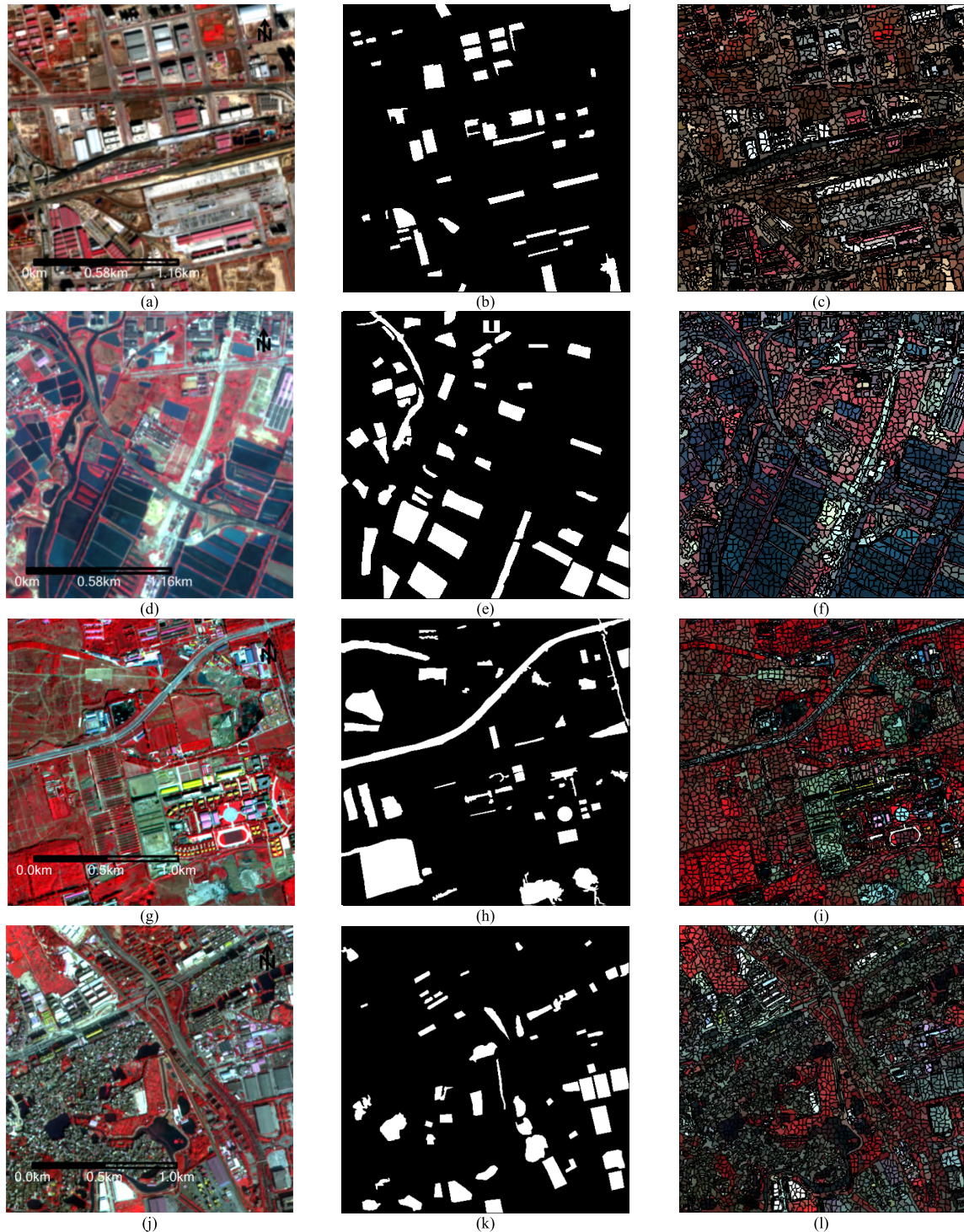
FIGURE 6. The dataset used to validate the proposed segmentation approach. (a), (d), (g), (j) are the original images of T1, T2, T3, T4, respectively, their color combination is: R: Near infrared, G: Red, B: Green; (b), (e), (h), (k) are the ground truth segments of T1, T2, T3, T4, respectively; (c), (f), (i), (l) are the over-segmentation results of T1, T2, T3, T4, respectively.

full of buildings and roads, while T2 is a rural area mainly composed of aquaculture pools. T3 and T4 are more complicated, since both urban and rural geo-objects can be found in the two images. The large differences in geo-object types

of the dataset provide different perspectives on the evaluation experiment.

Fig. 6 also provides the ground truth geo-objects used to quantitatively score the performance of segmentation

TABLE 4. Basic information of the dataset.

	Size (pixels)	Band information	Acquisition date	Resolution (m)	Acquisition platform
T1	400×400	Near infrared, Red, Green, Blue	Dec 31 2014	5.8	ZiYuan-3
T2	400×400	Near infrared, Red, Green, Blue	Dec 23 2013	5.8	ZiYuan-3
T3	500×500	Near infrared, Red, Green, Blue	Sep 27 2014	4.0	GaoFen-2
T4	500×500	Near infrared, Red, Green, Blue	Jan 23 2015	4.0	GaoFen-2

TABLE 5. Class information of the two test images.

	Number of primary classes	Names of the primary classes
T1	8	Light color building, grey building, bright building, shadow, water body, bare soil, vegetation, road
T2	7	Light color aquaculture pool, dark color aquaculture pool, vegetation, building, bright road, dark road, bare soil (river is spectrally similar to dark aquaculture pool, so they are considered as one)
T3	8	Light color building, grey building, bright building, bare soil, vegetation, agricultural land, road, shadow
T4	9	Light color building, grey building, bright building, vegetation, bare soil, shadow, road, water pond, impervious ground

results. There are totally 48, 49, 52, 54 reference polygons extracted from T1, T2, T3, T4, respectively. These reference geo-objects were all manually digitized by three expert-level remote sensing image interpreters, who were asked to finish the digitization work for those scenes independently. Then, the three operators discussed the quality for each reference geo-object, and the best one was chosen.

Despite the reference polygons needed for quantitative evaluation, class label information is also necessary with respect to analyzing the proposed segmentation method. Table 5 lists the detailed information on the types of geo-objects captured by the test images. It can be inferred from these information that the four images are rich in the category of geo-objects, which brings about difficulties for their successful segmentation.

B. EVALUATION METHOD

With the reference geo-objects shown in Fig. 6, it is straight-forward to utilize supervised strategy for quantitative evaluation. Yi *et al.*'s [28] method is adopted for this purpose, since it is widely applied in studies related to remote sensing image segmentation algorithm and has been reported to effectively reflect segmentation performance [29], [30]. This approach provides for each segmentation result four scores including precision (p), recall (r), f_1 and m_2. Readers are referred to [28] for detailed definition of these four variables, so they are not described in this article to save space. Note that p, r, f_1 and m_2 are all numerically ranged within (0,1) and the first two can be used to determine the type of the dominant segmentation error. For any segmentation result, apparent over-segmentation leads to high p and low r, while under-segmentation has the opposite pattern. f_1 and m_2 are overall indicators, so they are both low-valued when either over- or under-segmentation error becomes prevalent. On the contrary, a good segmentation results in relatively high-scored f_1 and m_2.

C. OVER-SEGMENTATION AND REGION-LEVEL K-MEANS

Before introducing the experimental results, it is necessary to provide descriptions on the two important initialization procedures, which are over-segmentation and region-level k-means. As already mentioned in sub-section 2.1, the over-segmentation approach based on seeded region growing [27] is used throughout this work. There are two parameters to be tuned for this method, including the seed spacing variable l_{seed}, and the seed growing threshold T_{spec}. l_{seed} mainly affects the average size of the resulted super-pixels, thus it is often set as the length of the minimal geo-object of interest. l_{seed} was tuned to be 10 for the four images, because this value can effectively avoid the over-merging of some small geo-objects, as well as help create initial segments with sufficiently large size. T_{sepc} controls the relative contribution of spectral and geometric information to the evolution of seeded segments. High T_{spec} will result in regularly shaped super-pixels, while their boundary may not well conform to the real edge of geo-objects. On the other hand, low T_{spec} underlines that the seeded segments should be merged with spectrally similar neighboring pixels, thus the shapes of the segments may vary wildly, producing bad visual effects. After a trial-and-error process, 30, 25, 30 and 30 were chosen for the T_{spec} of T1, T2, T3, T4, respectively. The resulted over-segmentation for the test images can be seen in the third column of Fig. 6, from which it can be observed that the super-pixels are quite consistent with the real boundary of geo-objects, and most of them are evenly shaped and averagely sized.

Based on the results of the aforementioned over-segmentation, a region-level k-means technique can be initiated to provide label for each initial segment. In the process of k-means clustering, every super-pixel is represented by a spectral vector, in which each entry is a mean spectral value of one band, thus the vector length equals the number of bands. The termination condition of the iterative clustering is similar to the traditional pixel-level k-means, so that this

procedure will end when the total change of clustering centers as compared to the ones generated in the last iteration is less than a very small positive value, namely 0.00001. The only parameter of this step is the number of classes, k, and some prior knowledge is required to properly set it. In fact, k has a serious impact on the performance of the proposed segmentation method, and a detailed analysis has been conducted to fully reflect it, which can be seen in the next section. Note that when $k = 1$, the region-level k-means is not performed, since the label information will not be used under this circumstance.

IV. RESULTS
This section is composed of three parts: the first describes the parameter analysis; the second exhibits the comparison of the proposed method to some other state-of-the-art approaches; the first two parts use the two ZiYuan-3 images. The third part further tests the proposed approach by using the GaoFen-2 dataset.

A. PARAMETERS ANALYSIS
1) RESULTS OF T1
According to the introduction of section 2 and 3, there are two primary parameters for the proposed method, including 1) the number of classes, k, and 2) the *Scale* in equation (11). Although ω_{shape} and $\omega_{compactness}$ in equation (1) and (2) are also two parameters of the proposed approach, there have been quite ample studies on them, such as [31]. Accordingly, a widely applied setting of these two parameters ($\omega_{shape} = 0.1$ and $\omega_{compactness} = 0.5$) was chosen throughout this work.

Fig. 7 shows the variation of the four performance scores (p, r, f_1, m_2) of T1 segmentation results obtained by our approach parameterized by different combinations of k and *Scale*. Note that from Fig. 7(a) to (j), each sub-figure corresponds to a k value, which can only be set as a positive integer and ranges from 1 to 10 with incremental step equal to 1. On the contrary, the incremental step of *Scale* is 0.01, being much smaller than that of k, because the numerical scope of *Scale* is very narrow (being (0,1)), and in this experiment, *Scale* ranges from 0.01 to 0.99), and the performance scores are very sensitive to this parameter according to the supervised evaluation.

By firstly observing the first ten sub-figures of Fig. 7, it is quite consistent that the best overall scores of different ks all occur at a relatively small *Scale* (generally less than 0.30). Besides, for all ks, with the rise of *Scale*, p gradually increases while r is approximately on the constant decline, indicating that high *Scale* tends to result in under-segmentation, but when it is very low, over-segmentation becomes dominant.

Fig. 7(k) demonstrates that only when $k = 3, 8, 9, 10$, f_1 is higher than the situation of $k = 1$, while in other cases, the values of f_1 are all inferior to the score marked by the red line. This is an interesting phenomenon because it indicates that the use of class label information may not guarantee the improvement of segmentation quality, and thus the selection

of k should be very cautious in order to achieve more accurate segmentation performance.

An interesting fact can be recognized by observing Fig. 7(l), in which the *Scale* corresponding to the maximal f_1 is obviously higher in the case of $k = 1$ than those of other k values. This is attributed to the introduction of Q in equation (8), because when $k > 1$, the numerical range of E is larger than that of $k = 1$ ($Q = 0$ in this case), so that for the selection of one tree node, relatively lower *Scale* is required for the situation of $k > 1$ as compared to $k = 1$. However, when $k > 1$, there is no apparent pattern for the variation of peak-f_1-*Scale*.

2) RESULTS OF T2
Fig. 8 illustrates the parameter analysis of T2, and the first ten sub-figures show the variation of the four performance scores with the increase of *Scale* for different k values. By comparing them to the counterparts of T1, similar conclusions can be made, primarily indicating that high-valued *Scale*s contribute to under-segmentation error while low-valued ones lead to over-segmentation incorrectness.

However, when it comes to the kth sub-figure of Fig. 8, different things can be noticed. Only two values of k (8, 9) can result in better f_1 than the result derived by employing our method parameterized by $k = 1$. Additionally, the advantage symbolized by the red line is not as obvious as that in Fig. 7. In cases of $k = 5, 6, 7$, the peak-f_1 scores are even much worse than the red-line number.

As for the last sub-figure of Fig. 8, it is evident that the peak-f_1-*Scale* has a decreasing trend with the increase of k, which is also different from the counterpart of T1's result. By contrast, for both test images, the values of peak-f_1-*Scale* corresponding to $k = 1$ are consistently much higher than those of $k > 1$.

B. COMPARISON TO OTHER COMPETITIVE METHODS
In order to validate the proposed segmentation algorithm, three competitive approaches are selected for comparison purposes, including a newly formulated priority-guided region-merging (PGRM) method [27], a hybrid region-merging (HRM) strategy [17], and a well-known algorithm named hierarchical segmentation (HSeg) [19]. The three segmentation schemes are all based on region-merging and are specifically designed for the segmentation of high-resolution multi-spectral remote sensing imagery. It is worth mentioning that the merging criteria of PGRM and HRM are both founded on the principle introduced by Baatz and Schäpe [16], which is also adopted in the BMT construction of our method, thus it is effective to exploit these two state-of-the-art algorithms to reveal whether the proposed approach can further improve segmentation performance as compared to the existent techniques. Note that in order to test the primary idea of this work (the use of class label information), the proposed method excluding class label cues is also adopted for comparison, so there are totally five segmentation strategies in this part of experiment.
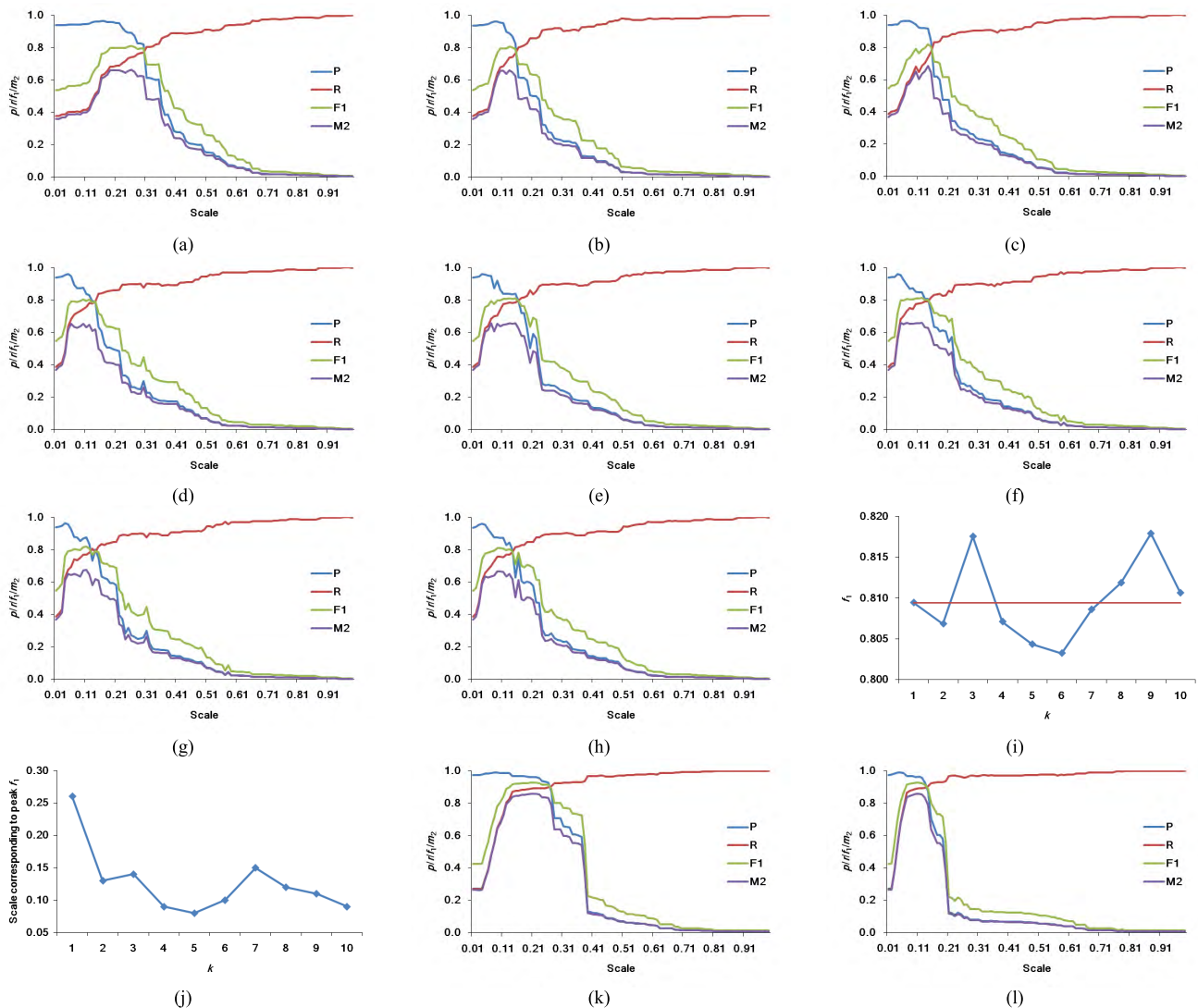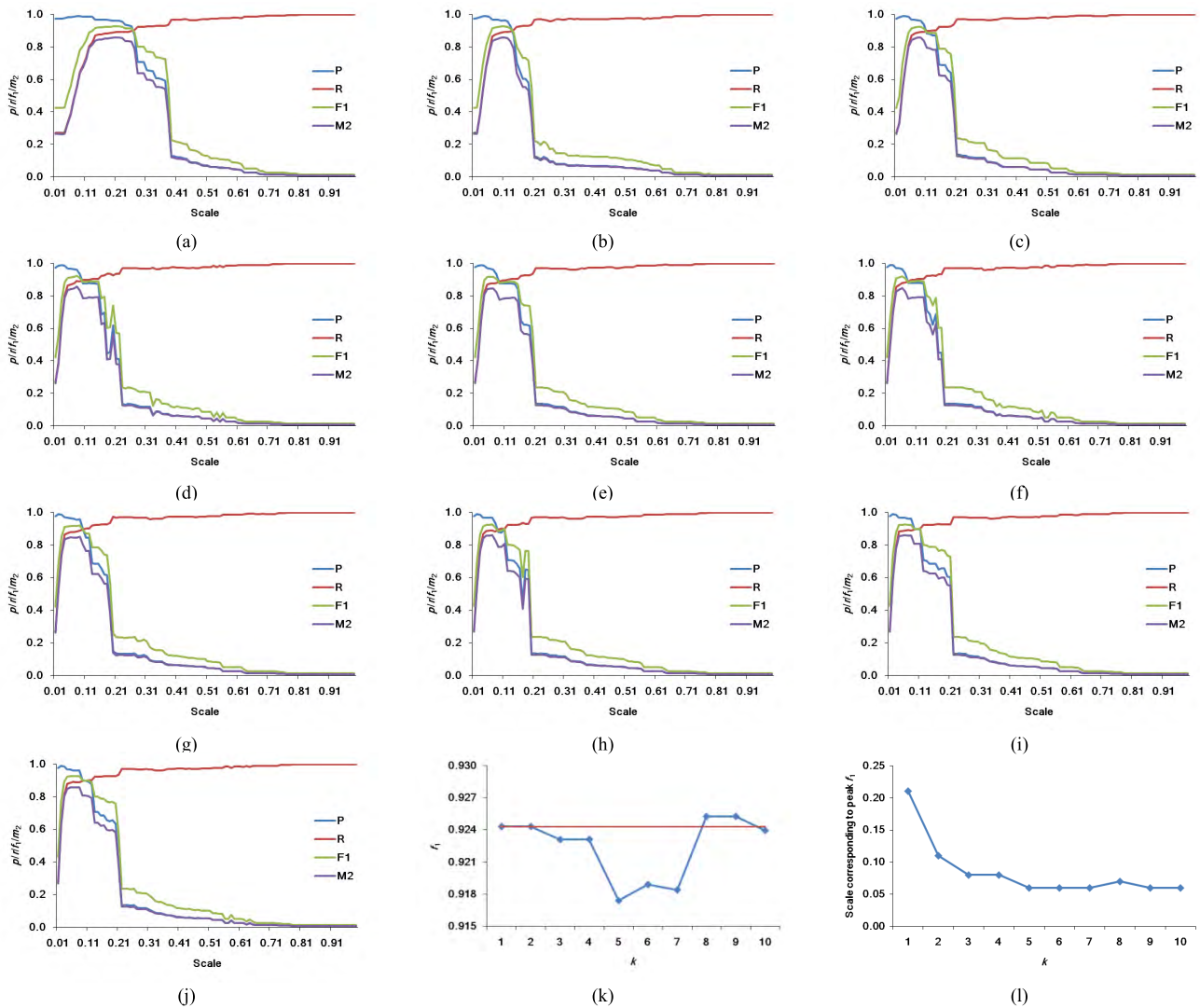
FIGURE 7. Parameter analysis based on the segmentation results of T1. (a) to (j) represent the variation of p, r, f_1, m_2 with the increase of Scale, and they respectively correspond to different k values ranging from 1 to 10. (k) illustrates the maximal f_1 measure of different k values; the red line indicates the f_1 value derived by the proposed segmentation method without using label information (when $k = 1$), and this line simply shows which k can lead to better performance than the situation of $k =$. The last sub-figure (l) exhibits the Scale value corresponding to the peak f_1 measure of different ks.

1) RESULTS OF T1

Fig. 9 shows T1's segmentation results produced by using the five algorithms. For fair comparison, the result with the highest f_1 and m_2 score for each method was chosen. The parameter setting for Fig. 9(a) and (b) was chosen according to the analysis shown in sub-section 4.1.1. Since the most influential parameter for the three other strategies (PGRM, HRM, HSeg) is the scale threshold, the four performance scores were plotted against this parameter to choose the optimal one. Fig. 10 provides these information. Consistent with the most multi-scale segmentation frameworks, the three methods all had the best overall performance at a relatively intermediate scale value, since over- or under-segmentation error took place when scale was too low or too high. By comparing the segmentation results in Fig. 9 to

the reference geo-objects displayed in Fig. 6(b), it can be apparently observed that most buildings were successfully segmented out, since this type of geo-object generally has distinct boundary and spectral features. However, for bare soil and vegetation, all of the five approaches yielded over-segmentation errors of different extent. For example, a piece of vegetated land (a red-colored spot situated at the upper-right part of T1) was obviously over-segmented by PGRM, HRM and HSeg, while it was more completely extracted by the two BMT-based methods.

As for the comparison between the two proposed algorithms, it can be seen that the two results were quite similar. However, some differences can be still observed. The one without using class label cues produced slightly more over-segmentation errors. For instance, an irregularly shaped white

FIGURE 8. Parameter analysis based on the segmentation results of T2. (a) to (j) represent the variation of p, r, f_1, m_2 with the increase of Scale, and they respectively correspond to different k values ranging from 1 to 10. (k) illustrates the maximal f_1 measure of different k values; the red line indicates the f_1 value derived by the proposed segmentation method without using label information (when $k = 1$), and this line simply shows which k can lead to better performance than the situation of $k = 1$. The last sub-figure (l) exhibits the Scale value corresponding to the peak f_1 measure of different ks.

TABLE 6. T1's optimal scores achieved by using the five segmentation approaches. The best score for each metric is highlighted in bold.

| Method (Parameter setting)	p	r	f_1	m_2
The proposed (k=9, $Scale$=0.11)	0.8757	**0.7672**	**0.8179**	**0.6760**
The proposed (k=1, $Scale$=0.26)	0.8898	0.7424	0.8094	0.6640
PGRM ($Scale$=20)	**0.9008**	0.6895	0.7811	0.6239
HRM ($Scale$=30)	0.8177	0.7346	0.7739	0.5893
HSeg ($Scale$=60)	0.8147	0.7413	0.7763	0.6150

building in the middle-left part of T1 was evidently over-segmented in Fig. 9(b), but it was well singled out in Fig. 9(a). This was in line with the quantitative evaluation results tabulated by Table 6, since the p and r of the method exploiting class label information were respectively higher and lower than the algorithm not considering label cues.

The numbers in Table 6 also indicate that the BMT-based methods were both superior to the other three approaches, and

the use of class label information further improved the overall segmentation performance.

2) RESULTS OF T2

Similar to the experiment carried out for T1, the best segmentation results produced by each method were generated and illustrated in Fig. 11. Fig. 12 demonstrates the
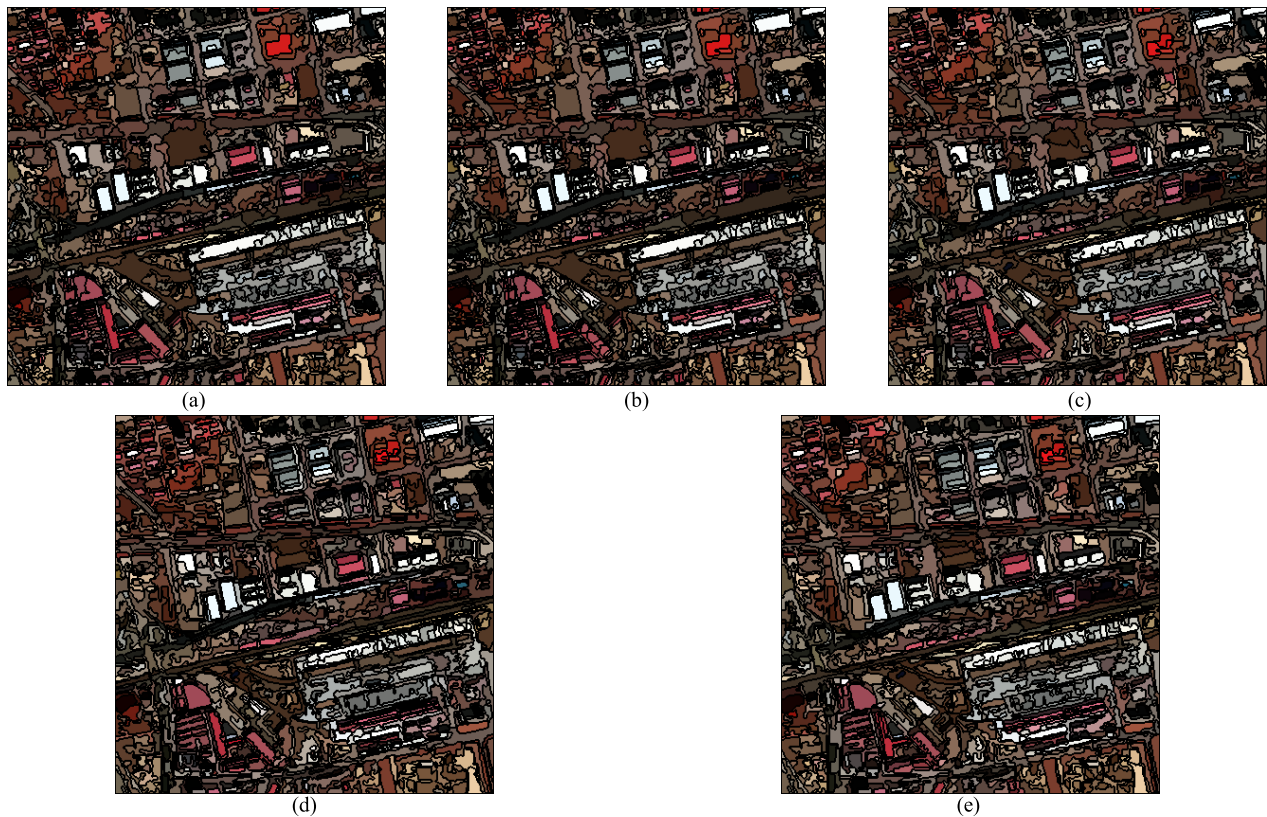
FIGURE 9. T1's segmentation results produced by using different segmentation approaches. (a) The proposed method ($k = 9$). (b) The proposed method without using class label information ($k = 1$). (c) PGRM. (d) HRM. (e) HSeg.
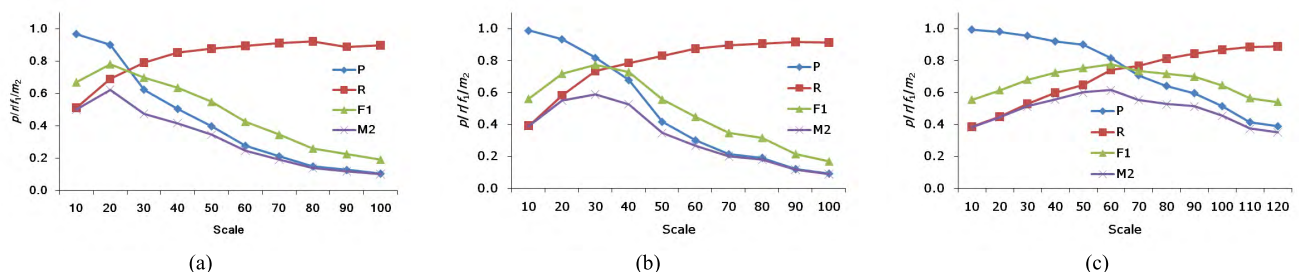
FIGURE 10. Variation of the four performance scores (p, r, f_1, m_2) for T1 with the increase of scale parameter for the approaches adopted for comparison. (a) PGRM. (b) HRM. (c) HSeg. These sub-figures can be used to identify the optimal scale parameter for the three segmentation methods.
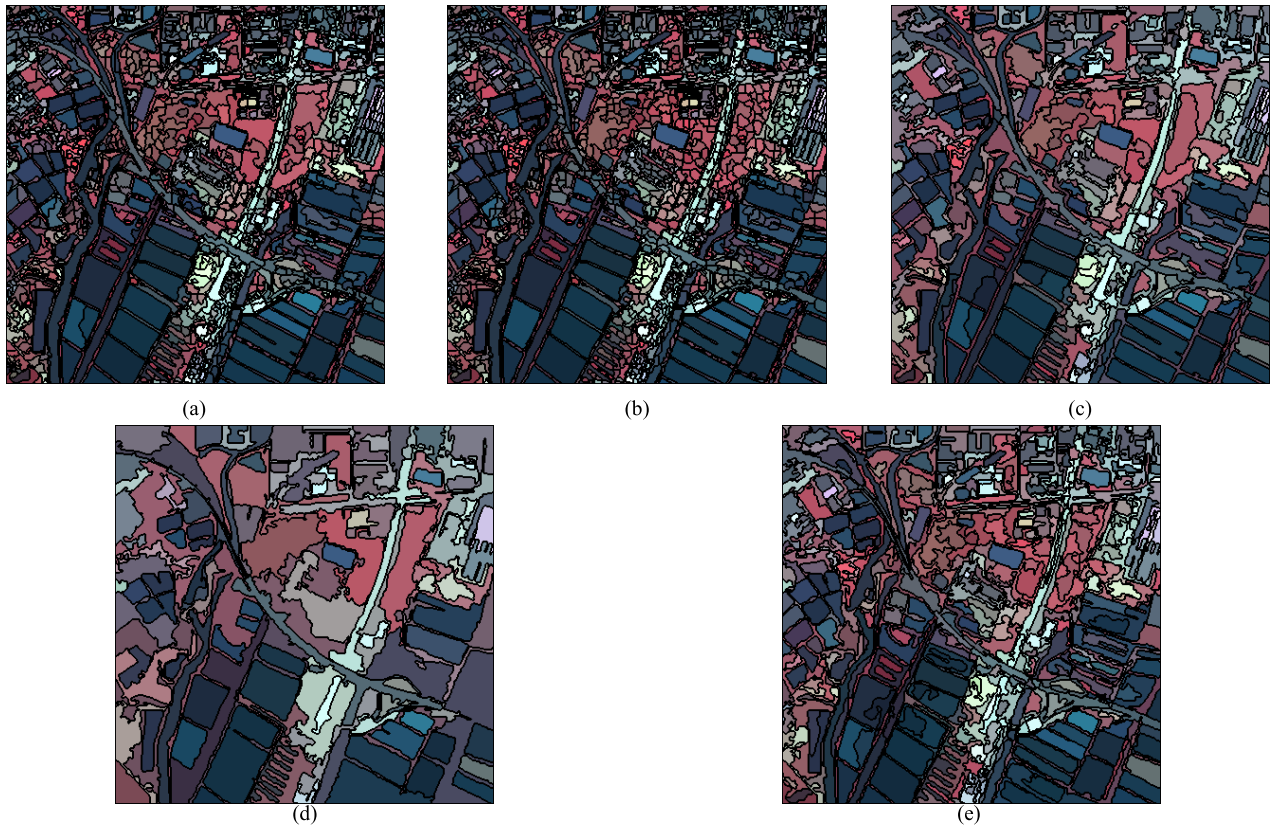
variation of the four performance cores with the increase of scale parameter for the three competitive methods (PGRM, HRM, HSeg), which was used to identify their optimal scale thresholds.

By carefully comparing the five segmentation results in Fig. 11, it is clear that most aquaculture pools were well segmented out, despite the fact that HRM produced obvious under-segmentation error. PGRM and HSeg produced relatively similar results, but they both over-segmented a few aquaculture pools as compared to the results obtained by using the two BMT-based techniques. The most obvious example is the dark-colored aquaculture pool located at the lower-left part of T2.

It is interesting to note that the two BMT-based approaches had quite similar performance. Although the method without using label information apparently over-segmented the bare soil area in the middle left part of T2, most reference geo-objects were quite similarly extracted by both algorithms. This is also embodied by the quantitative evaluation results listed in Table 7, in which it can be observed that the advantage of using class label cues is not as evident as that reflected by T1's experiment.

A closer inspection of Table 7 exhibits that the overall performance scores (f_1 and m_2) derived by the two BMT-based segmentation methods are all obviously superior to the counterparts of the other three algorithms.

FIGURE 11. T2's segmentation results produced by using different segmentation approaches. (a) The proposed method ($k = 9$). (b) The proposed method without using class label information ($k = 1$). (c) PGRM. (d) HRM. (e) HSeg.
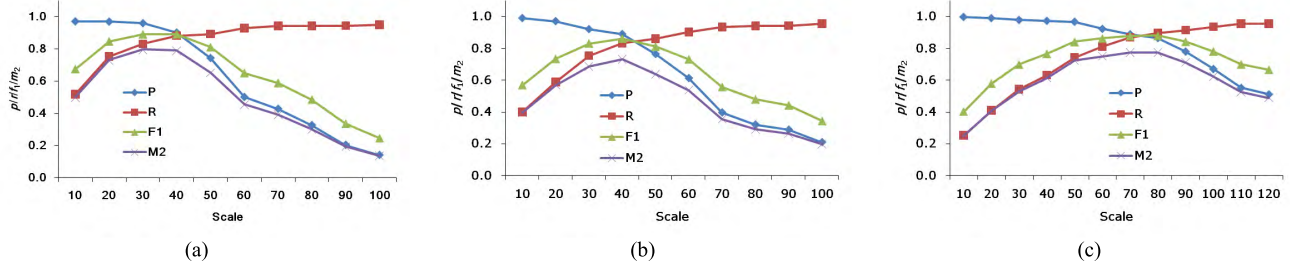
FIGURE 12. Variation of the four performance scores (p, r, f_1, m_2) for T2 with the increase of scale parameter for the approaches adopted for comparison. (a) PGRM. (b) HRM. (c) HSeg. These sub-figures can be used to identify the optimal scale parameter for the three segmentation methods.

TABLE 7. T2's optimal scores achieved by using the five segmentation approaches. The best score for each metric is highlighted in bold.

Method (Parameter setting)	p	r	f_1	m_2
The proposed (k=8, $Scale$=0.07)	**0.9650**	0.8886	**0.9252**	**0.8587**
The proposed (k=1, $Scale$=0.21)	0.9618	**0.8896**	0.9243	0.8577
PGRM ($Scale$=40)	0.9010	0.8816	0.8912	0.7898
HRM ($Scale$=40)	0.8912	0.8313	0.8602	0.7333
HSeg ($Scale$=80)	0.8647	0.8960	0.8801	0.7733

C. VALIDATION USING GAOFEN-2 DATASET

This sub-section demonstrates a further validation of the proposed approach by adopting the two GaoFen-2 images. Two competitive methods are used for comparison, including the multi-resolution segmentation (MRS) [16] which is very well-known in the community of object-based image analysis, and a locally thresholded region-merging technique (LTR) [32], which is recently proposed
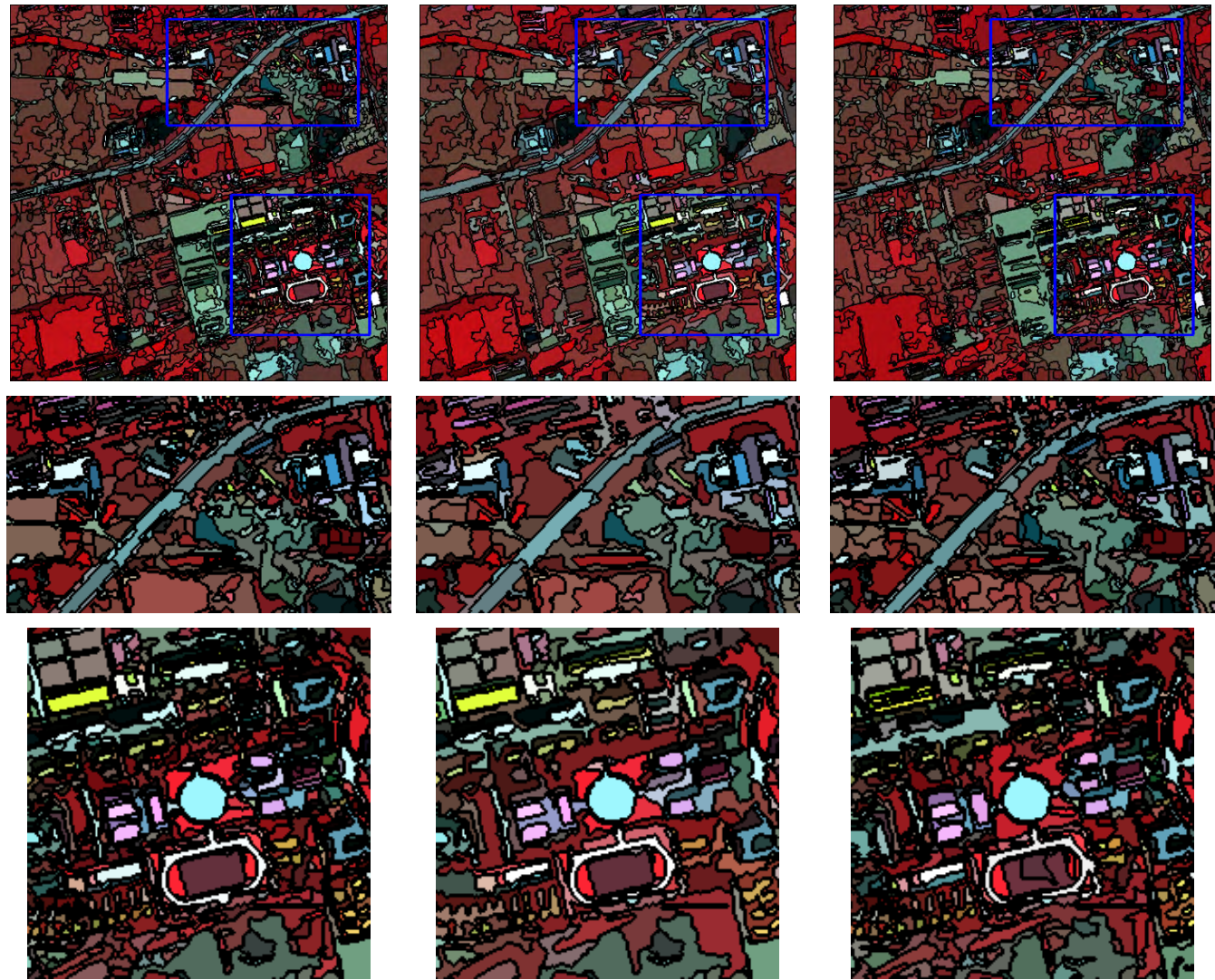
FIGURE 13. T3's segmentation results. From left to right, the first, second and third column correspond to the proposed method, MRS, LTR, respectively. The middle and the lowest row show the zoomed in segmentation results of the two blue rectangles marked in the first row.

TABLE 8. Quantitative evaluation results for T3.

Method (Parameter setting)	p	r	f_1	m_2
The proposed (k=9, *Scale*=0.19)	**0.7014**	0.6165	**0.6562**	**0.4548**
MRS (*Scale*=40)	0.6451	0.6210	0.6328	0.4053
LHT (*Scale*=1.8)	0.6127	**0.6935**	0.6506	0.4258

and reported to perform well for segmenting complex large images.

1) RESULTS OF T3

Fig. 13 illustrates the segmentation results of T3. For fair comparison, this figure exhibits the optimal result for each approach. The corresponding parameter setting for the three algorithms can be seen in table 8, where quantitative evaluation scores are also provided. According to visual observation, the three approaches produced decent segmentation

results, while a few spatial details can be found, as indicated in the two zoomed in regions shown in the lower two rows of Fig. 13. LTR yielded the most over-segmentation errors, since many geo-objects are obviously fragmented, such as the playground in the lower right rectangle of T3. MRS and the proposed technique performed approximately equally for most urban geo-objects, while for many rural ones, such as agricultural land, the latter leads to more complete segmentation.

According to table 8, the quantitative evaluation scores are in line with the visual analysis of Fig. 13. Our algorithm
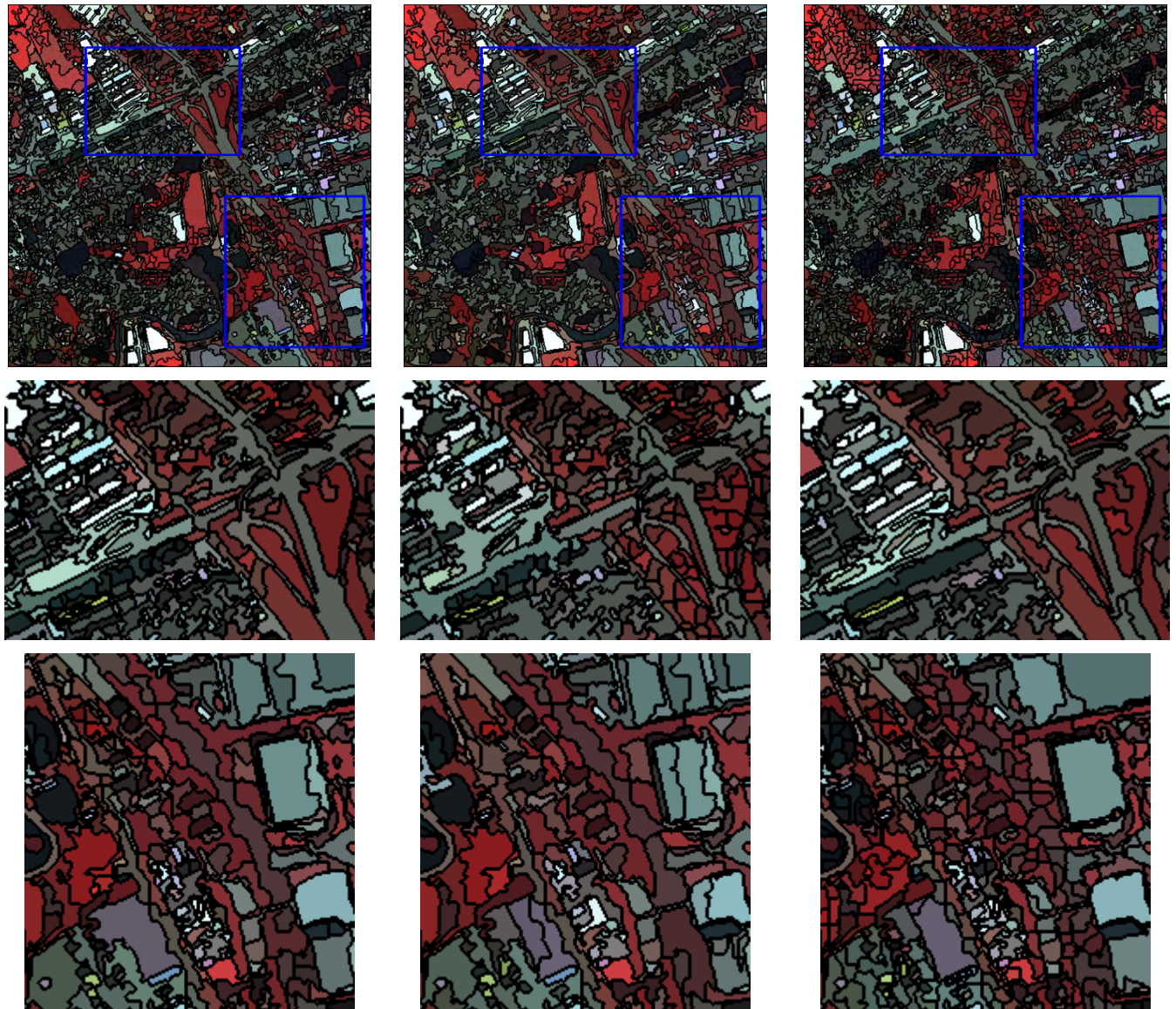
FIGURE 14. T4's segmentation results. From left to right, the first, second and third column correspond to the proposed method, MRS, LTR, respectively. The middle and the lowest row show the zoomed in segmentation results of the two blue rectangles marked in the first row.

TABLE 9. Quantitative evaluation results for T4.

Method (Parameter setting)	p	r	f_1	m_2
The proposed (k=10, *Scale*=0.15)	**0.8408**	**0.7961**	**0.8178**	**0.6910**
MRS (*Scale*=30)	0.7632	0.7602	0.7617	0.5538
LHT (*Scale*=0.5)	0.7220	0.7156	0.7188	0.5000

had the highest overall scores (f_1 and m_2), supporting the contribution of this work.

2) RESULTS OF T4

The optimal segmentation results for the three methods are illustrated in Fig. 14, and the corresponding parameter settings are given in table 9. Through visual analysis for the results of T4, it can be observed that although the new BMT method produced good results, some segmentation errors can

be evidently noticed, such as the light purple building being under-segmented as shown in the last row of Fig. 14, which probably due to the complexity of this test image. Unlike the results of T3, for which MRS outperformed LTR, T4's results tend to have an opposite pattern. For the two zoomed in regions of Fig. 14, LTR had better descriptions for many artificial geo-objects than those of MRS.

Consistently, the scores listed in table 9 agree well with the analytical results mentioned in the last paragraph. It is worth

noting that the proposed approach had the best scores for all of the four evaluation metrics for T4, which further prove that the new BMT segmentation algorithm is very competitive.

V. DISCUSSION

According to the experimental results described in section 4, it is safe to conclude that the proposed BMT-based technique out-performed the many state-of-the-art methods (PGRM, HRM, HSeg, MRS, LTR) in segmenting the test images. According to sub-section A of the experimental results, when comparing the two BMT algorithms, an interesting phenomenon can be observed. For T1 and T2, the best overall performance scores were achieved by using $k = 9$ and 8, respectively. These two values of k were both 1-integer larger than the counterparts listed in Table 5. Then, one may ask why k being slightly larger than the real number of primary geo-object types resulted in optimal scores, while when k was identically equal to the real number, the performance score was not that good? One possible explanation is provided as follows.

In the proposed technique, class label information is initialized by k-means clustering. This technique is actually an unsupervised classification strategy, which categorizes a number of entities into k ($k > 1$) classes. Each entity is represented by a feature vector, and it is assigned to the class label whose class center has the closest distance to this entity's vector. After the iterative process of k-means clustering, the variances of each class tend to be relatively low, so that the entities of each resulted class are as homogeneous as possible. However, in many cases different classes may have different extents of spectral homogeneity, indicating that the variances of different geo-object types may vary wildly. Under such circumstances, a class of large variance can be divided into two or more categories, in this way all of the resulted geo-object types have comparatively low level of spectral variance. In this sense, if the image to be classified has k classes and one of them has large variance, k-means algorithm parameterized by k may confuse a few entities of the large-variance class with other type(s), leading to sub-optimal classification result. Accordingly, if k-means is parameterized by $k + 1$, the resulted geo-object types are prone to be more homogeneous and more accurate, being more beneficial to the segmentation performance of the proposed segmentation algorithm. In this work, the class 'bare soil' of T1 (the dominant yellow colored parts in T1) and the class 'vegetation' of T2 (most reddish pixels in T2) are remarkably more heterogeneous than the other geo-object types, which can be deemed as the main cause of this phenomenon.

On the other hand, when k is set too high, the segmentation evaluation scores are also not optimal, as illustrated by Fig. 7(k) and Fig. 8(k). This is because the initial labels of the super-pixels belonging to one geo-object may be very different when k is set too large, producing negative effect on the calculation of equation (9), which in turn degrades the segmentation accuracy.

From the aforementioned discussion, it is clear to suggest that the setting of parameter k has an obvious influence of the performance of the proposed segmentation technique, and it should be set 1-integer larger than the real number of geo-object types to achieve optimal segmentation performance when the image to be segmented has a class with distinctively large variance.

VI. CONCLUSION

This article proposes a binary merge tree (BMT)-based segmentation algorithm for high resolution remote sensing imagery. The new method consists of two steps, including 1) BMT construction and 2) tree node selection. In the first step, the region-merging technique developed by Baatz and Schäpe [16] is adopted to build BMT. In order to increase the running efficiency of the proposed approach, super-pixels other than pixels are used as the processing units, and the seeded-region-merging algorithm based on the method developed by Su [27] is employed to generate super-pixels. The second step finishes the whole segmentation by selecting the appropriate tree nodes formed in step one. Through a detailed analysis on two schemes of tree node selection, a root-starting strategy is chosen in our implementation. Most importantly, by combining spectral as well as class label cues, a new tree node selection criterion is proposed. The validation experiments indicate that the proposed image segmentation method can produce superior performance as compared to some state-of-the-art approaches.

In the future, the proposed method will be combined with some OBIA classification or geo-object detection techniques to further test the potential of the contribution of this work. For example, the publicly available SZTAKI-INRIA Building Detection dataset [33], [34] may be used to see if the new BMT segmentation approach can be applied to building detection.

REFERENCES

[1] T. Blaschke, "Object based image analysis for remote sensing," *ISPRS J. Photogramm. Remote Sens.*, vol. 65, no. 1, pp. 2–16, 2010.

[2] T. Laschke *et al.*, "Geographic object-based image analysis—Towards a new paradigm," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 180–191, Jan. 2014.

[3] T. Su, "Efficient paddy field mapping using Landsat-8 imagery and object-based image analysis based on advanced fractel net evolution approach," *GISci. Remote Sens.*, vol. 54, no. 3, pp. 354–380, Jun. 2017.

[4] F. Löw, P. Knöfel, and C. Conrad, "Analysis of uncertainty in multi-temporal object-based classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 105, pp. 91–106, Jul. 2015.

[5] J. Peña, P. Gutiérrez, C. Hervás-Martínez, J. Six, R. Plant, and F. López-Granados, "Object-based image classification of summer crops with machine learning methods," *Remote Sens.*, vol. 6, no. 6, pp. 5019–5041, 2014.

[6] C. Berger, M. Voltersen, S. Hese, I. Walde, and C. Schmullius, "Robust extraction of urban land cover information from HSR multi-spectral and LiDAR data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 6, pp. 1–16, Oct. 2013.

[7] M. Voltersen, C. Berger, S. Hese, and C. Schmullius, "Object-based land cover mapping and comprehensive feature calculation for an automated derivation of urban structure types at block level," *Remote Sens. Environ.*, vol. 154, pp. 192–201, Nov. 2014.

[8] L. Ma *et al.*, "Object-based change detection in urban areas: The effects of segmentation strategy, scale, and feature space on unsupervised methods," *Remote Sens.*, vol. 8, no. 9, p. 761, 2016.

[9] W. Yu, W. Zhou, Y. Qian, and J. Yan, "A new approach for land cover classification and change analysis: Integrating backdating and an object-based method," *Remote Sens. Environ.*, vol. 177, pp. 37–47, May 2016.

[10] M. Golipour, H. Ghassemian, and F. Mirzapour, "Integrating hierarchical segmentation maps with MRF prior for classification of hyperspectral images in a Bayesian framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 54, no. 2, pp. 805–816, Feb. 2016.

[11] A. K. Qin and D. A. Clausi, "Multivariate image segmentation using semantic region growing with adaptive edge penalty," *IEEE Trans. Image Process.*, vol. 19, no. 8, pp. 2157–2170, Aug. 2010.

[12] Q. Yu and D. A. Clausi, "IRGS: Image segmentation using edge penalties and region growing," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 30, no. 12, pp. 2126–2139, Dec. 2008.

[13] O. Yousif and Y. Ban, "Improving SAR-based urban change detection by combining MAP-MRF classifier and nonlocal means similarity weights," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 10, pp. 4288–4300, Oct. 2014.

[14] A. A. Farag, R. M. Mohamed, and A. El-Baz, "A unified framework for MAP estimation in remote sensing image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 7, pp. 1617–1634, Jul. 2005.

[15] J.-M. Beaulieu and M. Goldberg, "Hierarchy in picture segmentation: A stepwise optimization approach," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 11, no. 2, pp. 150–163, Feb. 1989.

[16] M. Baatz and M. Schäpe, "Multiresolution segmentation—An optimization approach for high quality multi-scale image segmentation," in *Angewandte Geographische InformationsVerarbeitung XII*, J. Strobl and T. Blaschke, Eds. Heidelberg, Germany: Wichmann, 2000, pp. 12–23.

[17] X. Zhang, P. Xiao, X. Feng, J. Wang, and Z. Wang, "Hybrid region merging method for segmentation of high-resolution remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 98, pp. 19–28, Dec. 2014.

[18] X. Zhang, P. Xiao, X. Song, and J. She, "Boundary-constrained multi-scale segmentation method for remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 85, pp. 15–25, Apr. 2013.

[19] J. C. Tilton, Y. Tarabalka, P. M. Montesano, and E. Gofman, "Best merge region-growing segmentation with integrated nonadjacent region object aggregation," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4454–4467, Nov. 2012.

[20] J. Yang, P. Li, and Y. He, "A multi-band approach to unsupervised scale parameter selection for multi-scale image segmentation," *ISPRS J. Photogramm. Remote Sens.*, vol. 94, pp. 13–24, Aug. 2014.

[21] J. Yang, Y. He, and Q. Weng, "An automated method to parameterize segmentation scale by enhancing intrasegment homogeneity and intersegment heterogeneity," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 6, pp. 1282–1286, Jun. 2015.

[22] B. Johnson and Z. Xie, "Unsupervised image segmentation evaluation and refinement using a multi-scale approach," *ISPRS J. Photogram. Remote Sens.*, vol. 66, no. 4, pp. 473–483, Jul. 2011.

[23] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 561–576, Apr. 2000.

[24] H. G. Akçay and S. Aksoy, "Automatic detection of geospatial objects using multiple hierarchical segmentations," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 7, pp. 2097–2111, Jul. 2008.

[25] T. Liu, M. Seyedhosseini, and T. Tasdizen, "Image segmentation using hierarchical merge tree," *IEEE Trans. Image Process.*, vol. 25, no. 10, pp. 4596–4607, Oct. 2016.

[26] P. Salembier and S. Foucher, "Optimum graph cuts for pruning binary partition trees of polarimetric sar images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5493–5502, Sep. 2016.

[27] T. Su, "A novel region-merging approach guided by priority for high resolution image segmentation," *Remote Sens. Lett.*, vol. 8, pp. 771–780, Aug. 2017.

[28] L. Yi, G. Zhang, and Z. Wu, "A scale-synthesis method for high spatial resolution remote sensing image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 4062–4070, Oct. 2012.

[29] M. Wang, J. Huang, and D. Ming, "Region-line association constraints for high-resolution image segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 2, pp. 628–637, Feb. 2017.

[30] T. Su and S. Zhang, "Local and global evaluation for remote sensing image segmentation," *ISPRS J. Photogramm. Remote Sens.*, vol. 130, pp. 256–276, Aug. 2017.

[31] B. Schultz, M. Immitzer, A. R. Formaggio, I. D. A. Sanches, A. J. B. Luiz, and C. Atzberger, "Self-guided segmentation and classification of multi-temporal Landsat 8 images for crop type mapping in southeastern Brazil," *Remote Sens.*, vol. 7, no. 11, pp. 14482–14508, 2015.

[32] J. Yang, Y. He, and J. Caspersen, "Region merging using local spectral angle thresholds: A more accurate method for hybrid segmentation of remote sensing images," *Remote Sens. Environ.*, vol. 190, pp. 137–148, Mar. 2017.

[33] I. Grinias, C. Panagiotakis, and G. Tziritas, "MRF-based segmentation and unsupervised classification for building and road detection in peri-urban areas of high-resolution satellite images," *ISPRS J. Photogram. Remote Sens.*, vol. 122, pp. 145–166, Dec. 2016.

[34] C. Benedek, X. Descombes, and J. Zerubia, "Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 33–50, Jan. 2012.

TENGFEI SU was born in Hohhot, Inner Mongolia, China, in 1987. He received the B.S. degree in marine technology from the Tianjin University of Science and Technology in 2010 and the M.S. degree in physical oceanography from the First Institute of Oceanography, State Oceanic Administration of China, in 2013.

Since 2013, he has been a Lab Master with the College of Water Conservancy and Civil Engineering, Inner Mongolia Agricultural University. His research interests include OBIA-related remote sensing image processing and interpretation.

SHENGWEI ZHANG was born in Harbin, Heilongjiang, China, 1979. He received the B.S. degree in geography from Harbin Normal University in 2002, the M.S. degree in ecology from the Centre for Agricultural Resources Research, Institute of Genetic and Developmental Biology, in 2007, and the Ph.D. degree in eco-hydrology from the Centre for Agricultural Resources Research, Institute of Genetic and Developmental Biology, in 2010.

From 2010 to 2016, he was an Associate Professor with the College of Water Conservancy and Civil Engineering, Inner Mongolia Agricultural University. Since 2016, he has been a Professor with the College of Water Conservancy and Civil Engineering, Inner Mongolia Agricultural University. His research interests include quantitative remote sensing and hydrological process. He was a recipient of the two honorary titles, Chinese Academy of Science Light of Western China and Yong Scholar of Grassland Elite in 2017.

● ● ●