

Postprocessing framework for land cover classification optimization based on iterative self-adaptive superpixel segmentation

Boce Chu,^{a,b,*} Jie Chen,^a Qi Guo,^b Feng Gao,^b Jinyong Chen,^b
and Shicheng Wang^b

^aBeihang University, School of Electronics and Information Engineering, Beijing, China

^bKey Laboratory of Aerospace Information Applications of CETC, Shijiazhuang, China

Abstract. An increasing number of applications require land cover information from remote sensing images, thereby resulting in an urgent demand for automatic land use and land cover classification. Therefore, effectively improving the accuracy of land cover classification is a main objective in remote sensing image processing. We propose a land cover classification post-processing framework based on iterative self-adaptive superpixel segmentation (LCPP-ISSS) for remote sensing image data. This framework can further optimize the land cover classification results obtained by neural networks without changing the network structure. First, we propose the iterative self-adaptive superpixel segmentation algorithm for high-resolution remote sensing images to extract the boundary information of different land cover classes. Then, we propose a land cover classification result optimization method based on patch complexity to optimize the classification result by combining the boundary information with the semantic information. In an experiment, we compare the classification accuracy before and after using LCPP-ISSS and with other common methods. The results show that LCPP-ISSS outperforms the dense conditional random field and provides a 4% increase in the mean intersection over union and a 10% increase in overall accuracy. © The Authors. Published by SPIE under a Creative Commons Attribution 4.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.JRS.14.036514](https://doi.org/10.1117/1.JRS.14.036514)]

Keywords: remote sensing; high resolution; land cover classification; postprocessing; iterative; self-adaptive.

Paper 200268 received Apr. 9, 2020; accepted for publication Aug. 14, 2020; published online Aug. 28, 2020.

1 Introduction

With continuous improvements in data resolution in recent years, the details and boundaries of various land cover in images have become clearer. A high-resolution image contains abundant detailed information but also generally has considerable noise and redundant information, which interfere with the automatic and high-precision classification of land cover objects. The demand for an automatic land cover/land use classification method that can better handle the detailed information of high-resolution remote sensing images has become increasingly urgent.

Recently, many researchers have performed land cover classification based on traditional machine learning^{1,2} and deep learning methods.^{3,4} The full convolutional network (FCN)⁵ and its improved network have been applied for land cover classification tasks in high-resolution remote sensing images, which have achieved a certain effect.^{6,7} However, the distortion caused by the convolution structure of upsampling and downsampling will inevitably lead to errors, such as edge blurring and holes. Therefore, FCNs often confuse categories and provide unclear boundaries,⁸ thereby leading to poor performance in extracting land cover from dense and variable areas. In subsequent studies, many researchers hoped to resolve this problem by transforming the neural network structure. However, although changing the network structure can improve classification accuracy to a certain extent, neural networks have poor interpretability and operability due to the “black box” effect, and some specific problems may appear in actual

*Address all correspondence to Boce Chu, E-mail: Boce012628077@163.com

application.⁹ For example, when only parts of an image are poorly classified, as occurs frequently, the neural network is difficult to adjust, and retraining is very troublesome. Therefore, we think that it is meaningful if we can develop a simple efficient and human-controlled post-processing method to optimize interior and boundary areas of classification results obtained by neural networks.

In this paper, optimizing and improving the land cover classification postprocessing framework for remote sensing image data using an iterative self-adaptive superpixel segmentation algorithm (LCP-SS) is proposed. This framework can optimize the land cover classification results obtained by current algorithms, such as DeepLab^{10,11} and Unet,¹² instead of changing the network structure. In LCP-SS, we first propose an iterative self-adaptive superpixel segmentation (ISSS) algorithm to extract the boundary information of different land covers using the boundary attachment characteristics of superpixels. In addition, a land cover classification optimization method based on patch complexity (LCP-PC) is proposed to optimize the classification results by combining the boundary information and semantic information obtained by the neural network. In the experiments, we compared the results before and after using the proposed method, and the proposed method was also compared with other commonly used postprocessing methods, such as the dense conditional random field (DenseCRF).¹³ The remainder of this article is organized as follows: Sec. 2 presents the related work in this research field; Sec. 3 introduces the proposed method in detail; Sec. 4 discusses the experimental results and performance analysis; and Sec. 5 presents the conclusions of the paper.

2 Related Work

2.1 Superpixel Segmentation

The superpixel is the basic component of many land cover classification tasks, and it is generally used for land cover segmentation before classification. The concept of superpixels was first proposed by Ren and Malik¹⁴ in 2003. Superpixels are formed through object-based methods for homogeneous pixel merging. Pixel blocks with regular shapes are generated, and good boundary attachment characteristics are obtained. Many superpixel-based segmentation methods have been proposed, including the graph-based method proposed by Felzenszwalb and Huttenlocher,¹⁵ the Ncut (normalized cuts) method proposed by Shi and Malik,^{16,17} the superpixel lattice method proposed by Moore et al.¹⁸ and the turbopixel method proposed by Levinshtein et al.¹⁹ The simple linear iterative clustering (SLIC) method proposed by Achanta et al.^{20–22} in 2010 displayed good segmentation performance in images taken by conventional cameras and has been a main superpixel segmentation method.

However, remote sensing images are very large, and the density distributions of different land covers can vary greatly in the same image. Conventional segmentation methods, such as SLIC, need to set stable segmentation thresholds, such as the maximum number of seeds; consequently, these methods cannot obtain a variety of segmentation scales according to the density distribution of each land cover. Therefore, these methods cannot be effectively applied for segmentation tasks involving remote sensing images.

2.2 Land Cover Classification

In land cover classification, as a first step, scholars usually use superpixels to segment images into objects. Therefore, the unit to be processed is converted from pixels to superpixels. The segmentation results are then sent to various machine learning classification models to obtain classification results. For example, Liu et al.²³ used a multiscale superpixel-guided filter approach and classified land cover using high-resolution remote sensing images and a support vector machine (SVM). Subsequent studies improved this method in two aspects: the superpixel segmentation method and the classification model. Zhang et al.²⁴ optimized the superpixel segmentation method and designed an improved SLIC method to improve segmentation accuracy. Gu et al.²⁵ and Martins et al.²⁶ replaced the SVM with a deep convolutional neural network to model the superpixel content, thereby effectively improving classification ability.

Since the FCN, which frees the classification task from the presegmentation step, was proposed by Long et al.⁵ in 2015, an increasing number of scholars have applied FCNs in various fields. Kumar et al.²⁷ used an FCN to achieve the end-to-end segmentation of medical images. Mountelos et al.²⁸ used FCNs to perform wiper segmentation for unmanned vehicle images. Maggiori et al.²⁹ first applied FCNs to land cover classification in 2016 and obtained a significant improvement in accuracy compared to the accuracy of traditional methods. Subsequent scholars have improved FCNs in various ways. McGlinchy et al.³⁰ used Unet to replace FCNs for land cover classification, and Niu et al.³¹ used a deep network to classify hyperspectral images. Many scholars have focused on improving the structure of neural networks, such as SegNet³² and RefineNet,³³ to improve classification accuracy. However, due to the boundary blur and hole problems caused by the convolution step, it is generally difficult to obtain a breakthrough by continuously modifying the structure of the network.

2.3 Postprocessing Methods for Image Classification

Some scholars have also performed relevant research work on the postprocessing of image classification results and proposed some relevant postprocessing methods. In conventional approaches, simple morphological methods, such as expansion and decomposition, are generally used for optimization, and Wang and Zhang³⁴ used this approach to perform hole filling in airport segmentation tasks. However, morphological methods require the operator to understand the overall image distribution and accurately set the morphological parameters. Due to the large differences between the parameters of different images or even different parts of one image, it is difficult to achieve an ideal optimization result. In 2016, Zhou et al.³⁵ proposed a method for optimizing FCN classification results using a CRF, and the classification results for the PASCAL VOC2012 dataset were obviously improved compared to those based on other methods. Subsequently, Shen and Zhang³⁶ and Biao et al.³⁷ applied this method to various applications, such as brain segmentation in medicine and image segmentation for unmanned vehicles, and achieved certain improvements. In the past 2 years, Chu et al.³⁸ and Du and Du³⁹ applied CRF methods to optimize the results of land cover classification and achieved good optimization results in water classification and building classification, respectively. However, CRFs also need preset parameters, and the whole image is optimized by the same set of parameters, thus resulting in the fixed optimization strength; in addition, CRFs are unable to fit the remote sensing images with different land cover distributions.

3 Proposed Method

In this paper, LCPP-ISSS, which combines ISSS and an LCOM-PC, is proposed to optimize the preliminary land cover classification result. In this paper, the preliminary land cover classification result is obtained using Unet,⁴⁰ which has an outstanding performance compared to most other methods. Figure 1 shows a flowchart of LCPP-ISSS.

As shown in Fig. 1, Unet is first used to obtain the preliminary land cover classification results. Then, ISSS is used to obtain the optimal superpixel segmentation results, which are called leaf-superpixels in this paper. Finally, LCOM-PC is used to optimize the land cover classification result in the area of the leaf-superpixel. After all the leaf-superpixel segmentation results are optimized one by one, the final optimization result can be obtained by merging them.

3.1 Iterative Self-Adaptive Superpixel Segmentation Algorithm

Due to the differences in the density and complexity of land covers in different remote sensing images, current methods may lead to insufficient or excessive segmentation. For example, the land cover distribution of villages and towns is often relatively simple and should not be densely segmented, although cities that have complex land cover distributions should be segmented with high densities to obtain better segmentation results.

To solve the problems above, we propose ISSS, which can be used to adaptively select the number of iterations according to the density distribution of various land covers in remote

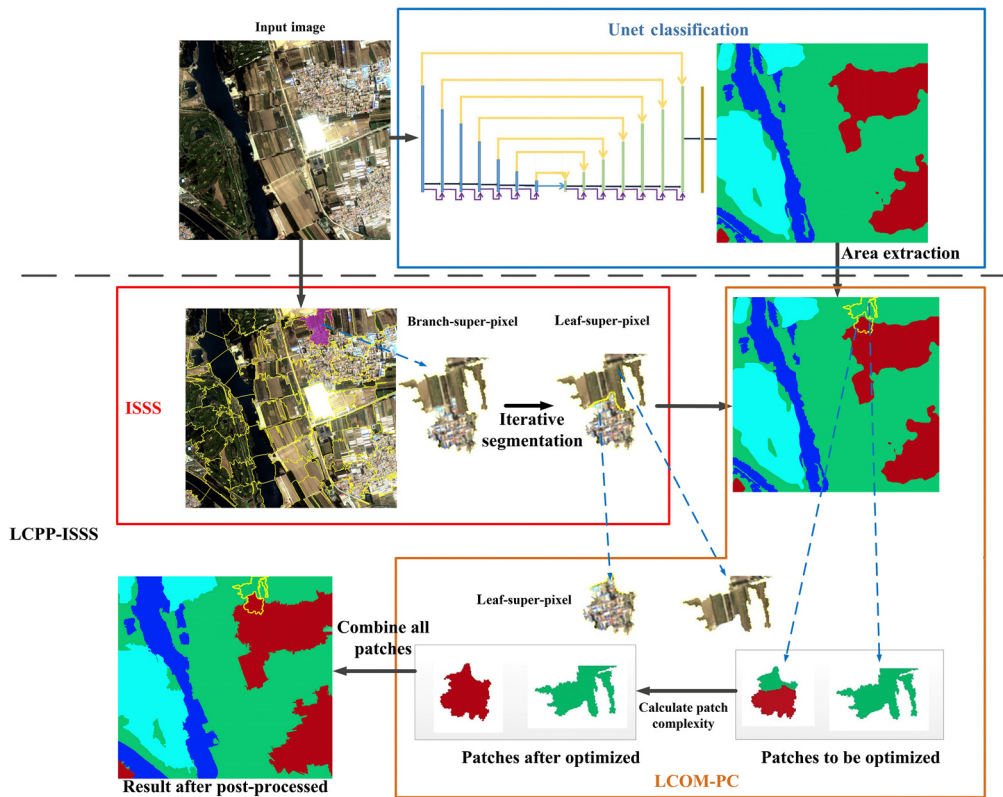


Fig. 1 Process of LCPP-ISSS.

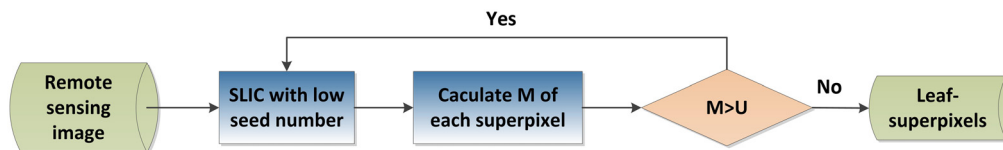


Fig. 2 Workflow of ISSS.

sensing images, and superpixels of different sizes and different intensities can be segmented according to the complexity of the land cover. Figure 2 shows the workflow details of the ISSS algorithm.

In Fig. 2, the ISSS algorithm uses SLIC with a low seed number to segment the image in the first step, and the rough superpixels are then obtained. The segmentation result of rough superpixels is shown in Fig. 3. The main reason for setting a small number of seeds is to avoid oversegmentation at the beginning. Then, a parameter M is defined to calculate the image complexity of pixel values and the spatial distribution inside the superpixels. The larger the value of M , the more complex the land cover classes inside the superpixel are. In addition, a fixed maximum superpixel complexity threshold U is preset by experience to determine whether the segmentation of superpixels is sufficient or not by comparing U with the M of the current superpixel. If the M of the current superpixel is less than U , then the current superpixel has been fully segmented. Otherwise, it is necessary to use SLIC to segment the superpixel two or more times again until the M is less than the U . We define a superpixel whose M is less than U as a leaf-superpixel; other superpixels are branch-superpixels. The ISSS is completed when all the superpixels are segmented into leaf-superpixels. M can be expressed by the following equations:

$$M = P * Q, \tag{1}$$

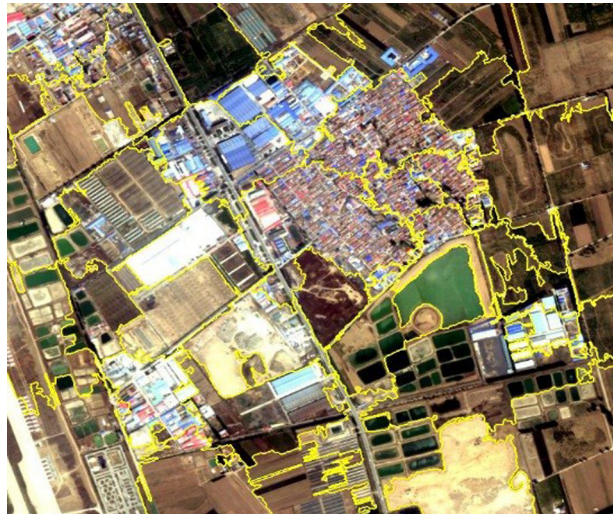


Fig. 3 Results after the first SLIC superpixel segmentation.

$$P = \sum_{i=1}^{255} m_i/N * \log(m_i/N), \tag{2}$$

$$Q = \frac{1}{N} \sum_{i=1}^h \sum_{j=1}^{w_i} [p(i, j) - \text{aver}(i, j, k)]^2, \tag{3}$$

$$\text{aver}(i, j, k) = \sum_{w=1}^k \sum_{v=1}^k [p(i + w, j + v)], \tag{4}$$

where P represents the difference in pixel values for the leaf-superpixels, Q represents the variety of spatial distributions, m_i is the number of pixels in a superpixel with pixel value I , N is the number of pixels in a superpixel, h is the height of a superpixel, w_i is the width of a superpixel with height I , $p(i, j)$ is the value of a pixel with coordinates (i, j) , and $\text{aver}(i, j, k)$ is the average of the pixel values of the matrix with (i, j) as the center and k as the width.

In Fig. 4, to display the difference between leaf-superpixels and branch-superpixels and the effect of ISSS more intuitively, we show the segmenting of superpixels from branch-superpixels into leaf-superpixels using ISSS. The parts in orange are branch-superpixels, and the others are leaf-superpixels. The land cover classes in the leaf-superpixels are very simple. In contrast, the land cover classes in branch-superpixels are complicated due to insufficient segmentation. In addition, the number of branch-superpixels decreases as the number of iterations increases.

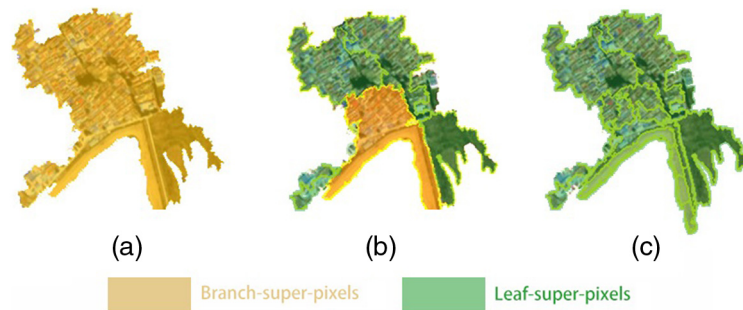


Fig. 4 Sample of iterative segmentation. (a) Sample superpixel after the first segmentation step. (b) Second segmentation result. (c) All leaf-superpixels after multiple iterations, which can be used as the final output of ISSS.

Finally, the non-orange superpixels in Fig. 4(b) were not segmented in Fig. 4(c), thus indicating that ISSS does not produce excessive segmentation.

3.2 LCOM-PC: Land Cover Classification Optimization Method Based on Patch Complexity

The distribution of land cover inside the leaf-superpixels after running ISSS should be relatively simple. However, due to the fuzzy boundary issue and other problems, the classification results of Unet in the leaf-superpixels' region are not simple. In this paper, LCOM-PC is proposed to use the boundary information of leaf-superpixels to optimize the classification result of Unet by adjusting the land cover classes in the leaf-superpixel region.

We defined the classification results of Unet in the leaf-superpixel region as a leaf-patch. A parameter S was defined to represent the distribution complexity of land cover in each leaf-patch. The formula of S will be derived later. As shown in Fig. 5, the S of each leaf-patch is calculated in the first step of LCOM-PC. Then, a fixed maximum leaf-patch complexity threshold L is preset. LCOM-PC can determine whether a leaf-patch is appropriate by comparing S with L . If S is less than L , then the classification results inside the leaf-patch are simple enough; therefore, Unet has performed well in the leaf-superpixel region, and the leaf-patch remains unchanged. If S is larger than L , then Unet has performed poorly in the leaf-superpixel region because the classification results inside the leaf-patch are too complicated. In this case, all the land cover types in a leaf-patch are optimized to the class that occupies the largest proportion of the leaf-patch. After all the leaf-patches are optimized, the final output of LCPP-ISSS can be obtained by merging them. To display the effect of LCOM-PC more intuitively, we show the optimizing of the leaf-patch in Fig. 6.

The parameter S is calculated by the following equations:

$$\text{region}_{\text{leaf-patch}} = \{\text{region}_1, \text{region}_2, \dots, \text{region}_k\}, \tag{5}$$

$$S = \sum_{\text{region}=\text{region}_1}^{\text{region}_k} \frac{n_k}{n_{\text{leaf-patch}}} \log\left(\frac{n_k}{n_{\text{leaf-patch}}}\right), \tag{6}$$

where $\text{region}_{\text{leaf-patch}}$ is the region of the whole leaf-patch and is composed of several disconnected regions in different classes, n_k is the number of pixels in region_k , and $n_{\text{leaf-patch}}$ is the number of pixels in the leaf-patch.

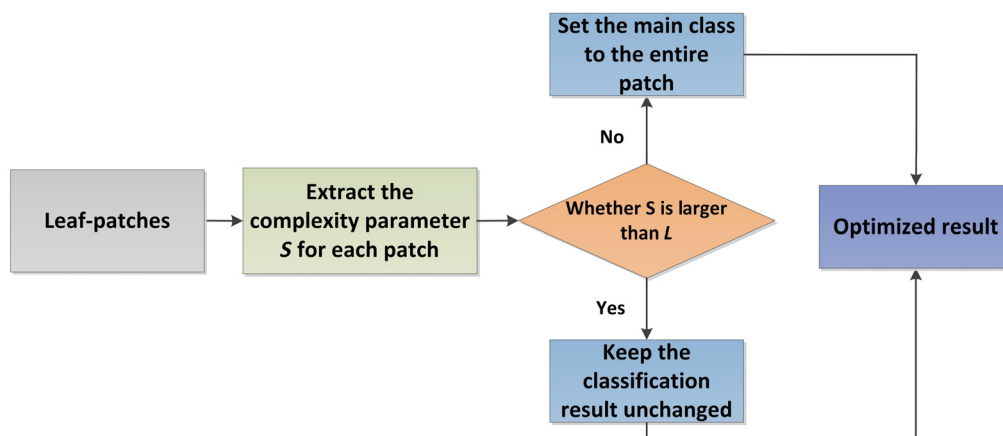


Fig. 5 Workflow of LCOM-PC.

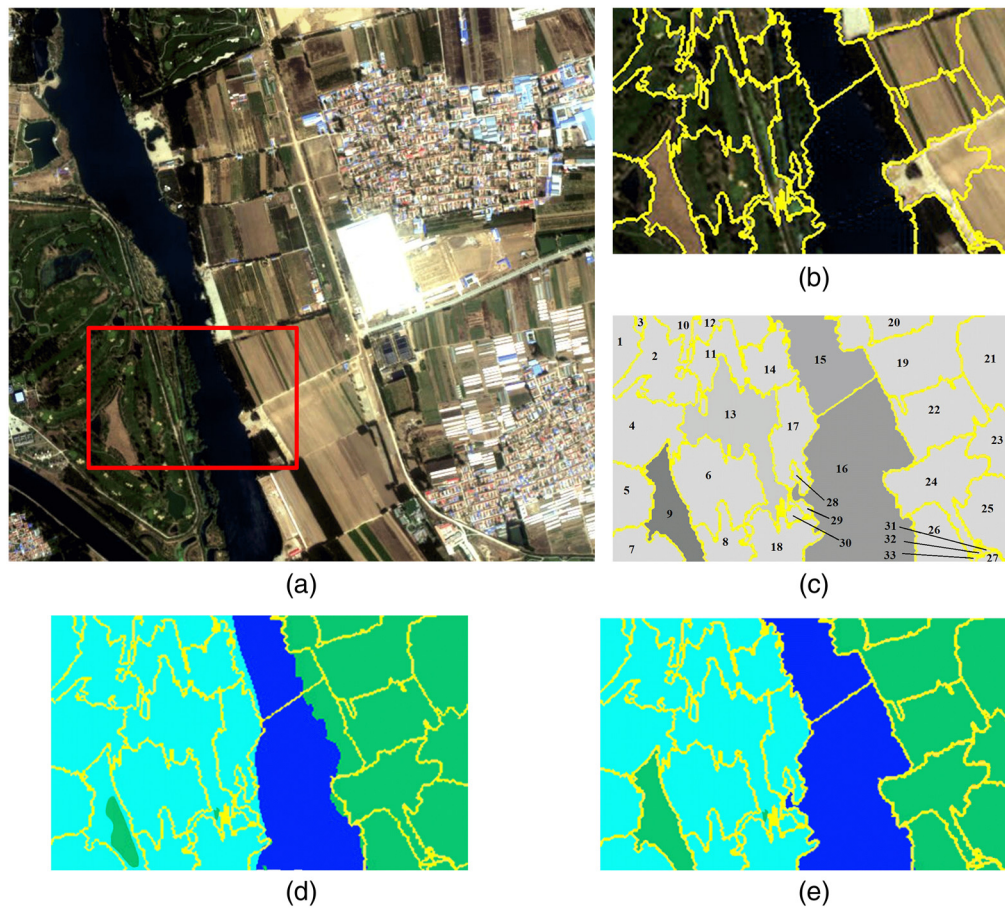


Fig. 6 Process demonstration through example. (b) ISSS result of area in (a), which is in a red rectangle; (c) code of each leaf-superpixel in (b), (d) the classification result of Unet, and (e) optimized result of LCOM-PC. S of {9,15,16}, which is larger than L ; therefore, the classification result is adjusted to the main class. S of other regions is less than L . Classification results remain unchanged.

4 Implementation and Performance Analysis

4.1 Data and Study Areas

The images used in this experiment come from the Gaofen-2 satellite,⁴¹ which is equipped with two panchromatic and multispectral charge-coupled device camera sensors with respective resolutions of 1 and 4 m. This setup can provide a 45-km combined mapping band, which yields multispectral images of 6908×7300 pixels. Considering the high resolution, high quality, and abundant details of Gaofen-2 satellite images, such images provide an ideal data source for land cover classification and are highly suitable for verifying the effectiveness of postprocessing methods.

To verify that the proposed method has good robustness in different regions, we used 50 Gaofen-2 satellite images as the verification dataset, which contains images of most of the main cities in China, such as Wuhan, Beijing, and Shanghai. A sample image is shown in Fig. 7. The area in Fig. 7(a) is situated in Wuhan, Hubei Province, China, with central geographical coordinates of $30^{\circ}6' N$ and $114^{\circ}2' E$; the image data were taken on February 12, 2015. We cut each image into several patches in 720×680 with 50-pixel overlapping as input data to reduce computing memory usage.

4.2 Experimental Results and Performance Analysis

All programs were executed in TensorFlow and Python 3.5. Four different land cover classes (including water, buildings, vegetation, and others) were selected for the experiment.

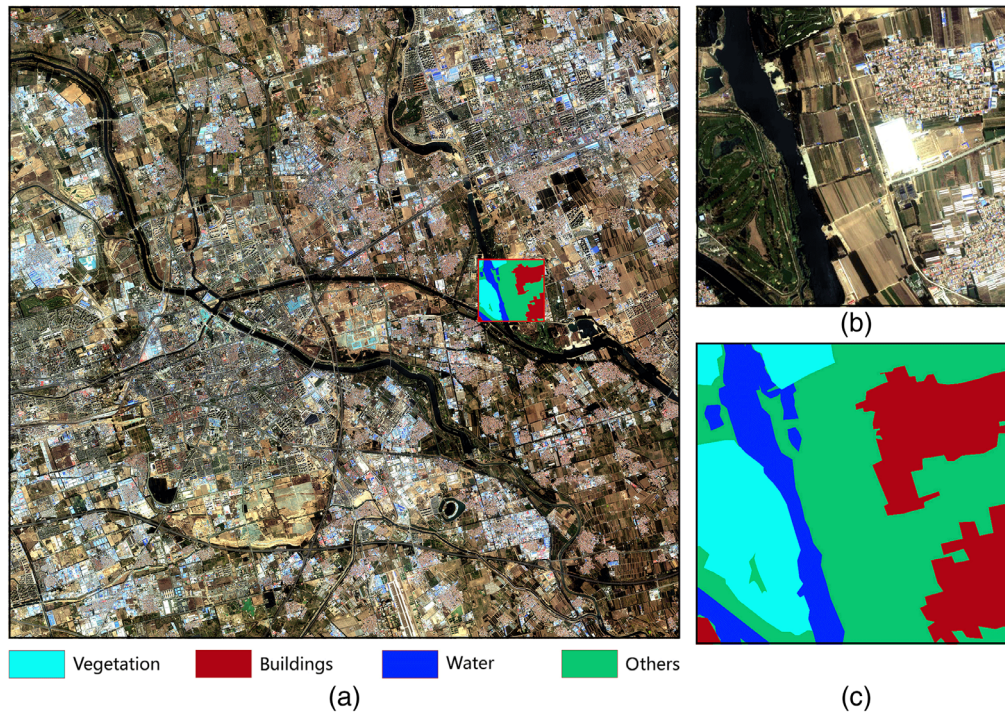


Fig. 7 GaoFen-2 image and example data. (a) GaoFen-2 multispectral imagery, (b) example of a sampled image patch, and (c) ground truth of the image patch.

LCPP-ISSS includes mainly two parts: segmentation and optimization, which are completed by ISSS and LCOM-PC, respectively. Therefore, we conducted experiments separately on the segmentation and optimization results. First, we verified the effectiveness and superior performance of LCPP-ISSS in segmentation by comparing ISSS with SLIC. Then, we compared LCPP-ISSS with the current mainstream method, DenseCRF, to verify the superiority of the proposed method in optimization.

4.2.1 Segmentation performance comparison between ISSS and SLIC

We experimentally compared the ISSS and SLIC methods and the segmentation result of two typical images with a size of 720×680 , as shown in Fig. 8. As is shown in Fig. 8(a), when the seed number of SLIC is small, the areas of the white factory building and small lake (which are marked by a red circle) are insufficiently segmented and these areas are mixed in the same superpixel with other land covers. In Fig. 8(b), when we increase the seed number of SLIC, the white factory building and small lake are divided into multiple parts; i.e., there is oversegmentation. In Fig. 8(c), in ISSS, the factory building and the small lake are independently divided into a single superpixel. By comparison, ISSS has a better segmentation effect.

Figure 8 shows that the result of traditional segmentation methods (such as SLIC) is severely limited by some parameters (such as seed number), and it is difficult to find a stable seed to apply to all situations very well. On the contrary, ISSS is an adaptive iterative segmentation method and is unaffected by the parameters, so ISSS can achieve better segmentation results in most cases. To further compare the segmentation performance of SLIC and ISSS, we calculated the relationship between the number of superpixels and the average complexity of superpixels for the two methods, and this comparison is shown in Fig. 9.

With the increasing number of superpixels, ISSS can obtain a lower average complexity with a smaller number of superpixels faster than SLIC. Although SLIC can eventually achieve a similar average complexity, SLIC needs to segment the image into more superpixels, and the convergence is relatively slow. As we mentioned above, LCOM-PC optimized classification by adjusting the land cover classes using superpixels as units. Therefore, the segmentation method

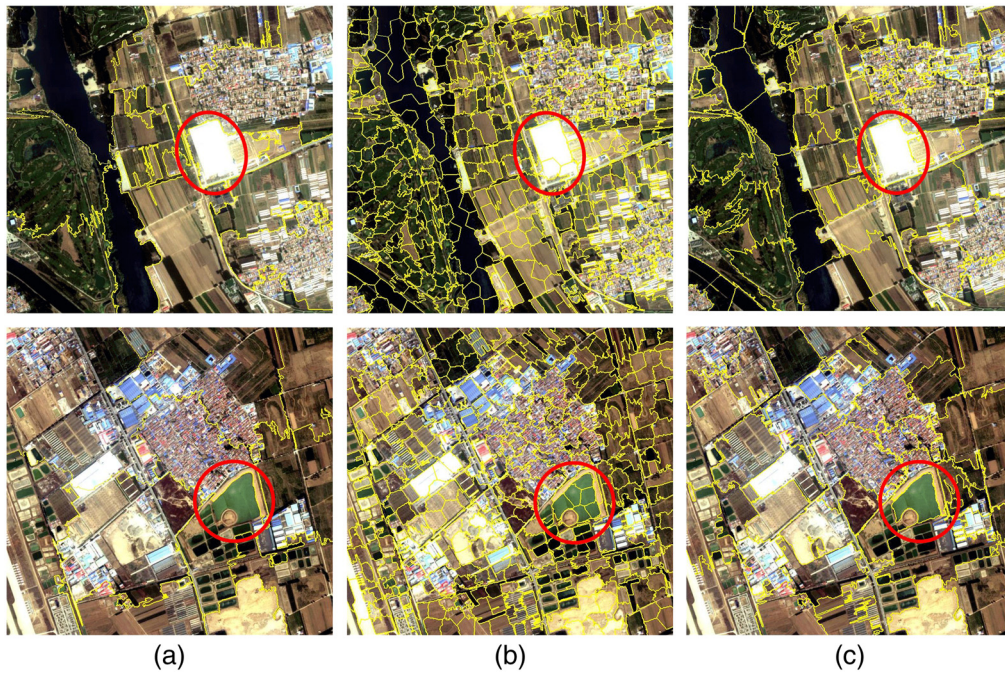


Fig. 8 Comparison of ISSS and SLIC segmentation performance. (a) SLIC with a small number of seeds, (b) SLIC with a large number of seeds, and (c) segmentation result of ISSS.

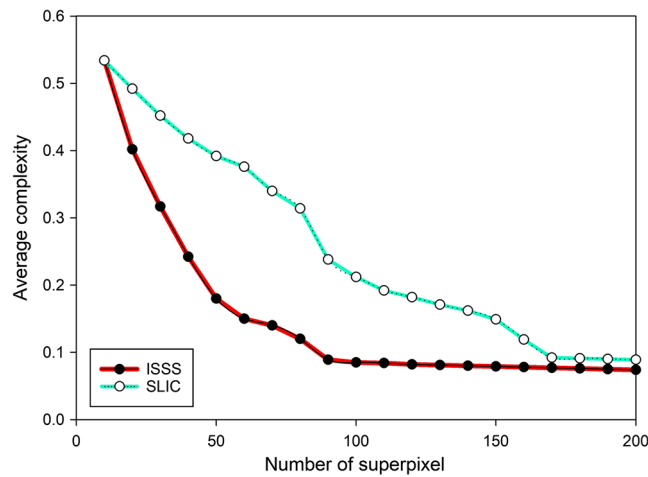


Fig. 9 Comparison of the number of superpixels and the average complexity of superpixels based on ISSS and SLIC.

that can achieve the same segmentation effect with fewer superpixels is better. Obviously, ISSS is more suitable than SLIC for postprocessing tasks.

4.2.2 Optimization performance comparison between LCPP-ISSS and DenseCRF

DenseCRF is currently widely used in neural network classification postprocessing. Therefore, we compared the proposed method with DenseCRF to validate the performance of the proposed method in terms of optimization. We performed several comparative experiments, and the results are shown in Fig. 10. To better reflect the difference in the classification results of the different methods, we draw the circles around areas where the methods obtained significantly different classification results. The region where the results produced by LCPP-ISSS are closer to the

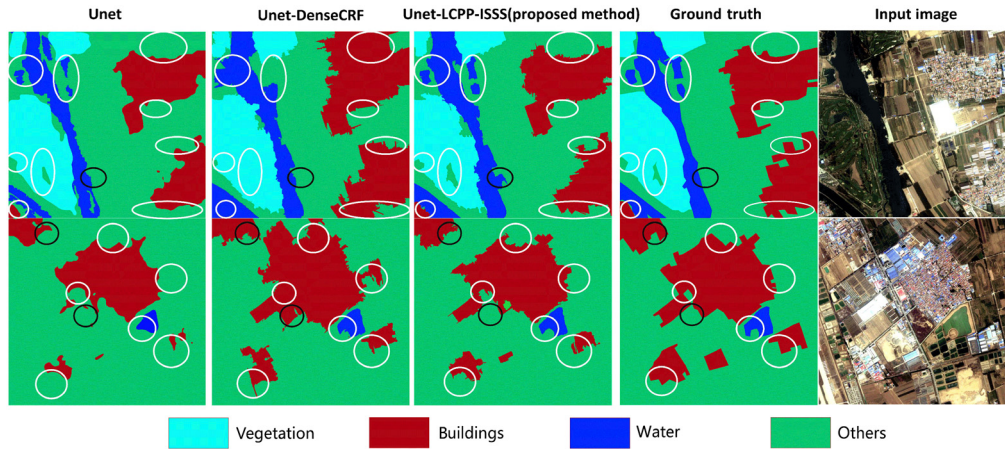


Fig. 10 Comparison of LCPP-ISSS and DenseCRF. (a) Original classification result, (b) classification result of DenseCRF, (c) classification result of LCPP-ISSS, (d) ground truth, and (e) input image.

ground truth than those produced by DenseCRF is marked with a white circle. The region where the results produced by DenseCRF are closer to the ground truth than those produced by LCPP-ISSS is marked with a black circle.

As shown in Fig. 10, the number of white circles is much larger than the number of black circles. Therefore, compared with DenseCRF, LCPP-ISSS displays outstanding performance. In addition, the overall results show that the optimization result of LCPP-ISSS is closer to the ground truth than the optimization result of a simple Unet and DenseCRF. To comprehensively compare LCPP-ISSS with other methods and quantitatively assess the performance of these methods, the overall accuracy (OA), intersection over union (IOU), and mean intersection over union (mean-IOU) were employed. OA is the percentage of correctly predicted pixels among all pixels. The IOU is a statistical value that reflects the consistency between predicted labels and ground truth. Let $Q_{i,j}$ denote the number of pixels of class i predicted as class j and N represent the number of all pixels. The values of OA, IOU, and mean-IOU range between 0 and 1, and a higher value indicates higher accuracy and better performance. OA, IOU, and mean-IOU can be expressed as

$$OA = \frac{\sum_i Q_{ii}}{\sum_i \sum_j Q_{i,j}}, \tag{7}$$

$$IOU = \frac{\sum_i Q_{ii}}{\sum_j Q_{ji}}, \tag{8}$$

$$\text{Mean-IOU} = \frac{1}{N} \sum_i \frac{\sum_i Q_{ii}}{\sum_j Q_{ji}}, \tag{9}$$

$$N = \sum_i \sum_j Q_{i,j}. \tag{10}$$

Table 1 shows that compared with DenseCRF, the proposed method provides a 4% increase in mean-IOU and a 10% increase in accuracy. Table 2 shows that the proposed method outperforms DenseCRF in terms of the mean-IOU of the different land cover classes, especially in terms of the mean-IOU of vegetation, which improves from 92.313% to 96.36% under the proposed method. In addition, LCPP-ISSS is also superior to DenseCRF in operability. DenseCRF requires many parameters to be set manually, such as kernel functions, which differ greatly for different kinds of images, and adjusting parameters is cumbersome. In comparison, LCOM-PC requires fewer parameters to be set in advance; consequently, this method can avoid many limitations related to manual intervention.

Table 1 Comparison of different indexes.

Method	Mean-IOU (%)	OA (%)
Unet	77.41	82.56
Unet-DenseCRF	86.53	85.25
Unet-LCPP-ISSS (proposed method)	90.94	95.52

Table 2 IOU of different land cover classes.

Classes	Unet	Unet-DenseCRF	Unet-LCPP-ISSS (proposed method)
Water	82.5	85.42	91.81
Buildings	68.36	75.33	80.85
Vegetation	86.06	92.313	96.36
Other	82.73	83.069	94.74

5 Conclusions

In this paper, a postprocessing framework named LCPP-ISSS is proposed to optimize the results of land cover classification. In LCPP-ISSS, ISSS is proposed to introduce the concept of iteration to achieve superpixel segmentation, and different kinds of images are appropriately segmented by considering the complexity of superpixels. ISSS effectively solves the problems of oversegmentation and insufficient segmentation caused by traditional superpixel methods for segmenting remote sensing images. Then, an optimization algorithm called LCOM-PC is proposed to optimize the classification results in combination with the segmentation results. In LCOM-PC, we defined and calculated the patch complexity of the classification results obtained by Unet. Some strategies were used to optimize the results according to the patch complexity, and these strategies proved to be effective in experiments.

It can be concluded from the experiments that the proposed ISSS algorithm performs better than SLIC in superpixel segmentation for remote sensing images, and ISSS can accurately segment regions with different land cover densities. Moreover, compared to other current methods, the optimization method we propose can achieve higher values for different indexes. In the future, we will improve the patch complexity calculation method and design more reasonable optimization strategies according to different results to further improve the optimization effect.

Acknowledgments

We are grateful to the anonymous reviewers and academic editor, whose constructive suggestions have improved the quality of this paper. In addition, in the current severe situation created by the COVID-19 epidemic, we would like to express our sincere gratitude to all frontline medical staff. This research was funded by the China Electronics Technology Group (CETC) Key Laboratory of Aerospace Information Applications, Grant No. SXX19629X060. The authors declare no conflicts of interest.

References

1. T. John et al., "Global land cover classification by remote sensing: present capabilities and future possibilities," *Remote Sens. Environ.* **35**(2–3), 243–255 (1991).
2. D. Jiang et al., "A simple semi-automatic approach for land cover classification from multi-spectral remote sensing imagery," *PLoS One* **7**(9), e45889 (2012).

3. K. Konstantinos et al., "Deep learning for multi-label land cover classification," *Proc. SPIE* **9643**, 96430Q (2015).
4. P. Zhang et al., "Urban land use and land cover classification using novel deep learning models based on high spatial resolution satellite imagery," *Sensors* **18**(11), 3717 (2018).
5. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, Boston, Massachusetts, pp. 3431–3440 (2015).
6. M. Pashaie and M. J. Starek, "Fully convolutional neural network for land cover mapping in a coastal wetland with hyperspatial UAS imagery," in *IEEE Int. Geosci. and Remote Sens. Symp.*, Yokohama, Japan, pp. 6106–6109 (2019).
7. Y. Liu et al., "Automatic building extraction on high-resolution remote sensing imagery using deep convolutional encoder–decoder with spatial pyramid pooling," *IEEE Access* **7**, 128774–128786 (2019).
8. L. Mou and X. X. Zhu, "RiFCN: recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images," arXiv:1805.02091 (2018).
9. O. Loyola-González, "Black-box vs. white-box: understanding their advantages and weaknesses from a practical point of view," *IEEE Access* **7**, 154096–154113 (2019).
10. C. Chen et al., "Encoder–decoder with atrous separable convolution for semantic image segmentation," *Lect. Notes Comput. Sci.* 833–851 (2018).
11. L. Yuan, J. S. Yuan, and D. Z. Zhang, "Remote sensing image classification based on DeepLab-v3+," *Laser Optoelectron. Prog.* **56**(15), 152801 (2019).
12. R. Li et al., "DeepUNet: a deep fully convolutional network for pixel-level sea-land segmentation," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **11**(11), 3954–3962 (2018).
13. N. Reddy, P. Singhal, and K. Krishna, "Semantic motion segmentation using dense CRF formulation," in *Proc. Indian Conf. Comput. Vision Graphics and Image Process.* (2014).
14. X. F. Ren and J. Malik, "Learning a classification model for segmentation," in *Proc. 9th IEEE Int. Conf. Comput. Vision*, pp. 10–17 (2003).
15. P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vision* **59**, 167–181 (2004).
16. J. B. Shi and J. Malik, "Normalized cuts and image segmentation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognit.*, San Juan, Puerto Rico, pp. 731–737 (1997).
17. J. B. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(8), 888–905 (2000).
18. A. P. Moore et al., "Superpixel lattices," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, Anchorage, Alaska, pp. 1–8 (2008).
19. A. Levinstein et al., "TurboPixels: fast superpixels using geometric flows," *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(12), 2290–2297 (2009).
20. R. Achanta et al., "SLIC superpixels," EPFL (2010).
21. R. Achanta et al., "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(11), 2274–2282 (2012).
22. Y. Zhang et al., "Semisupervised classification based on SLIC segmentation for hyperspectral image," *IEEE Geosci. Remote Sens. Lett.* **17**(8), 1440–1444 (2020).
23. S. Liu et al., "A multiscale superpixel-guided filter approach for VHR remote sensing image classification," in *IEEE Int. Geosci. and Remote Sens. Symp.*, Yokohama, Japan, pp. 1017–1020 (2019).
24. L. Zhang, C. Han, and Y. Cheng, "Improved SLIC superpixel generation algorithm and its application in polarimetric SAR images classification," in *IEEE Int. Geosci. and Remote Sens. Symp.*, Fort Worth, Texas, pp. 4578–4581 (2017).
25. F. Gu, H. Zhang, and C. Wang, "A classification method for polsar images using SLIC superpixel segmentation and deep convolution neural network," in *IEEE Int. Geosci. and Remote Sens. Symp.*, Valencia, pp. 6671–6674 (2018).
26. J. Martins et al., "Image segmentation and classification with SLIC superpixel and convolutional neural network in forest context," in *IEEE Int. Geosci. and Remote Sens. Symp.*, Yokohama, Japan, pp. 6543–6546 (2019).

27. S. Kumar et al., “A deep learning for brain tumor MRI images semantic segmentation using FCN,” in *4th Int. Conf. Comput. Commun. and Autom.*, Greater Noida, India, pp. 1–4 (2018).
28. A. Mountelos et al., “Vehicle windshield detection by fast and compact encoder–decoder FCN architecture,” in *8th Int. Conf. Modern Circuits and Systems Technol.*, Thessaloniki, Greece, pp. 1–4 (2019).
29. E. Maggiori et al., “Fully convolutional neural networks for remote sensing image classification,” in *IEEE Int. Geosci. and Remote Sens. Symp.*, Beijing, pp. 5071–5074 (2016).
30. J. McGlinchy et al., “Application of UNet fully convolutional neural network to impervious surface segmentation in urban environment from high resolution satellite imagery,” in *IEEE Int. Geosci. and Remote Sens. Symp.*, Yokohama, Japan, pp. 3915–3918 (2019).
31. Z. Niu et al., “DeepLab-based spatial feature extraction for hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.* **16**(2), 251–255 (2019).
32. V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: a deep convolutional encoder–decoder architecture for image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(12), 2481–2495 (2017).
33. G. Lin et al., “RefineNet: multi-path refinement networks for high-resolution semantic segmentation,” in *IEEE Conf. Comput. Vision and Pattern Recognit.*, Honolulu, Hawaii, pp. 5168–5177 (2017).
34. S. Wang and L. Zhang, “Airport detection based on superpixel segmentation and saliency analysis for remote sensing images,” in *IEEE Int. Geosci. and Remote Sens. Symp.*, Valencia, pp. 2511–2514 (2018).
35. H. Zhou et al., “Image semantic segmentation based on FCN-CRF model,” in *Int. Conf. Image, Vision and Comput.*, Portsmouth, pp. 9–14 (2016).
36. H. Shen and J. Zhang, “Fully connected CRF with data-driven prior for multi-class brain tumor segmentation,” in *IEEE Int. Conf. Image Process.*, Beijing, pp. 1727–1731 (2017).
37. W. Biao et al., “Research on image semantic segmentation algorithm based on fully convolutional HED-CRF,” in *Chin. Autom. Cong.*, Xi’an, China, pp. 3055–3058 (2018).
38. Z. Chu et al., “Sea-land segmentation with Res-UNet and fully connected CRF,” in *IEEE Int. Geosci. and Remote Sens. Symp.*, Yokohama, Japan, pp. 3840–3843 (2019).
39. S. Du and S. Du, “Land cover classification using remote sensing images and lidar data,” in *IEEE Int. Geosci. and Remote Sens. Symp.*, Yokohama, Japan, pp. 2479–2482 (2019).
40. O. Ronneberger, P. Fischer, and T. Brox, “U-Net: convolutional networks for biomedical image segmentation,” in *Med. Image Comput. and Comput.-Assisted Intervention – MICCAI 2015*, pp. 234–241 (2015).
41. X. Tong et al., “Large-scale land cover classification in Gaofen-2 satellite imagery,” in *IEEE Int. Geosci. and Remote Sens. Symp.*, Valencia, pp. 3599–3602 (2018).

Boce Chu received his master’s degree in information and communication engineering from Beijing University of Posts and Telecommunications, China, in 2016. He is currently pursuing his PhD in information and signal processing at Beihang University. His present research interests focus on algorithms for remote sensing image processing based on artificial intelligence.

Jie Chen received his BS and PhD degrees in information and communication engineering from Beihang University, Beijing, China, in 1996 and 2002, respectively. Since 2004, he has been an associate professor in the School of Electronics and Information Engineering at Beihang University. From 2009 to 2010, he was a visiting researcher in the School of Mathematics and Statistics at University of Sheffield, Sheffield, United Kingdom, where he was involved in ionospheric effects on low-frequency space radars that measure forest biomass and ionospheric electron densities. Since 2011, he has been a professor in the School of Electronics and Information Engineering at Beihang University.

Biographies of the other authors are not available.