Int. J. Appl. Math. Comput. Sci., 2020, Vol. 30, No. 3, 399–413
DOI: 10.34768/amcs-2020-0030

CLASSIFICATION OF HIGH RESOLUTION SATELLITE IMAGES USING IMPROVED U–NET

YONG WANG [a], DONGFANG ZHANG [a], GUANGMING DAI [a,*]

[a] School of Computer Science
China University of Geosciences
Wuhan, Hubei 430074, China
e-mail: {yongwang,zhangdongfang}@cug.edu.cn,
li2002t@gmail.com

Satellite image classification is essential for many socio-economic and environmental applications of geographic information systems, including urban and regional planning, conservation and management of natural resources, etc. In this paper, we propose a deep learning architecture to perform the pixel-level understanding of high spatial resolution satellite images and apply it to image classification tasks. Specifically, we augment the spatial pyramid pooling module with image-level features encoding the global context, and integrate it into the U-Net structure. The proposed model solves the problem consisting in the fact that U-Net tends to lose object boundaries after multiple pooling operations. In our experiments, two public datasets are used to assess the performance of the proposed model. Comparison with the results from the published algorithms demonstrates the effectiveness of our approach.

Keywords: satellite image classification, deep learning, U-Net, spatial pyramid pooling.

1. Introduction

Satellite image classification is essential for many socio-economic and environmental applications of geographic information systems, including urban and regional planning, conservation and management of natural resources, etc. (Miao *et al.*, 2014). Thanks to the rapid progresses in remote sensing technology, and the reduction of acquisition costs, a bulk of images of the Earth are readily available nowadays. These images are taken from satellites or airplanes and differ in imaging modalities, spatial and spectral resolutions, or dynamic ranges (Castelluccio *et al.*, 2015). Satellite imagery, covering a large geographic area with a high temporal frequency, offers a unique opportunity for deriving land use and land cover information through the process of image interpretation and classification.

During the last decades, great efforts have been made in developing approaches to infer land usage from satellite images (Gong *et al.*, 2015). However, it is still one of the most challenging problems for automatic labeling high spatial resolution satellite images at the pixel level due to the high intraclass and low interclass variabilities presented in the images (Yang *et al.*, 2019). In fact, traditional pixel-based or object-oriented algorithms, which mostly rely on hand-crafted features, are not fit for the complexity of objects shown in high spatial resolution scenes (Cleve *et al.*, 2008). Owing to the development of deep learning frameworks, especially deep convolutional neural networks (CNNs), the state-of-the-art performances of various computer vision tasks are obtained based on some methods, such as semantic segmentation (Chen *et al.*, 2018) and object detection (Ren *et al.*, 2017). Meanwhile, semantic segmentation of satellite images has benefited greatly from deep learning approaches (Bei *et al.*, 2017; Zhang *et al.*, 2018a; Scott *et al.*, 2017; Zhao and Du, 2016; Gang *et al.*, 2018).

Deep CNNs have become dominant approaches in remote sensing since they are able to automatically learn powerful representations from the input images (Razavian *et al.*, 2014). A deep CNN comprises multiple connected layers, mainly convolutional layers and pooling layers. It can efficiently extract multi-level features from the spectral and spatial information of satellite images. In

*Corresponding author

images, local combinations of edges form motifs, motifs assemble into parts, and parts form objects. The pooling allows representations to vary very little when elements in the previous layer vary in position and appearance. Deep CNNs exploit the property that many natural signals are compositional hierarchies. In such hierarchies, higher-level features are obtained by composing lower-level ones. Therefore, more semantic information is shown in the representation as the feature level ascends.

In terms of semantic segmentation of satellite images, a variety of CNN-based approaches have been presented. These approaches are usually grouped into patch-based and pixel-based methods. Patch-based networks train models on small image tiles and predict one label for each small tile. Such networks achieve pixel-level prediction for the entire image by adopting a sliding window approach (Sharma *et al*., 2017). With respect to pixel-based methods, they take arbitrary-sized inputs and predict correspondingly-sized labels. Generally, these end-to-end frameworks apply architectures based on fully convolutional networks (FCNs) (Maggiori *et al*., 2016). An FCN employs the convolutional layer instead of the fully-connected layer to achieve pixel-level prediction, and then uses the surrounding label information more efficiently than the patch-based network does (Long *et al*., 2015; Tao *et al*., 2018). Afterwards, almost all advanced methods in semantic segmentation are based on this model.

It is noteworthy that an encoder-decoder architecture becomes increasingly popular in semantic segmentation due to its high flexibility and superiority (Peng *et al*., 2019). The mostly used encoder-decoder example is SegNet, a widely used FCN, where unpooling operation is included for better up-sampling (Badrinarayanan *et al*., 2017). However, in this way, skip connections between the encoder and decoder layers are ignored. Hence, SegNet has a poor spatial accuracy. U-Net, an extension of SegNet by adding skip connections, has better spatial accuracy and achieves great success in semantic segmentation on both medical images and RS images (Ronneberger *et al*., 2015; Kim *et al*., 2018). Such an architecture can capture sharper object boundaries by gradually recovering the spatial information. The encoder path in U-Net follows the typical architecture of a convolutional network; therefore, rich semantic information is encoded in the last feature map. Meanwhile, the decoder path gradually recovers sharp object boundaries.

For multi-class labeling tasks such as land use and land cover classification, since the features of different land cover types or ground objects are usually presented at various scales, a local and global balance needs to be traded off for multiple spatial domain information. Besides, although the context information of the image helps to remove the ambiguity of the local features, this information is not incorporated in the original form of the U-Net model (Mi and Hu, 2017). In addition, in the encoding process, local information such as boundaries of objects may be lost when employing consecutive pooling layers, which allows reducing the parameters and extracts long-range information. In order to learn the contextual information at multiple scales, the spatial pyramid pooling module is widely used by probing the incoming features with filters or pooling operations at multiple rates and multiple effective fields-of-view. Feature maps in different subregions generated by the spatial pyramid module significantly enhance the segmentation of various classes. The multi-scale feature capturing capability of the spatial pyramid module is especially useful in land cover and land use classification tasks in the complex urban area characterized by a high spatial heterogeneity.

In this paper, attempting to combine the advantages of U-Net and spatial pyramid pooling, we follow the work of Zhang *et al.* (2018b) and use spatial pyramid pooling to enrich the encoder module in U-Net for semantic segmentation of high spatial resolution satellite images. Specifically, the spatial pyramid pooling module is utilized in the bottom layer of U-Net to extract multi-scale global context features. In addition, the spatial pyramid pooling module with image-level features encoding the global context is integrated into the U-Net structure. Based on two public datasets from the Evlab-SS benchmark (Mi and Hu, 2017), we train and test our model and related works—not only ASPP-Unet (Zhang *et al.*, 2018b), but also FCN-8s (Long *et al*., 2015), SegNet (Badrinarayanan *et al*., 2017) and U-Net (Ronneberger *et al*., 2015). Experimental results show that the model by Zhang *et al.* (2018b) produces better classification results than U-Net and our model further boosts performance. According to the visualization of the results on the testing patches with different methods, our model solves the problem that U-Net will lose object boundaries after multiple pooling operations.

2. Related work

Before the arrival of deep networks, the best performing methods mostly relied on hand engineered features for classifying pixels independently. Typically, a patch is fed into a classifier, such as boosting (Shotton *et al*., 2009; Zhuowen and Xiang, 2010), random forests (Shotton *et al*., 2008), or support vector machines (SVM) (Fulkerson *et al*., 2009), to predict the class probabilities of the center pixel. Although substantial improvements have been achieved by incorporation of richer information from the context (Carreira *et al*., 2012) and structured prediction techniques (Carreira and Sminchisescu, 2011), the performance of these systems

has always been compromised by the limited expressive power of the features. In recent years, a lot of works have proven that deep learning is an effective way for semantic segmentation of satellite images. FCNs marked an important milestone in the development of semantic segmentation. The deconvolutional procedure of the original FCN is simple, but leads to the loss of details of object structures. Models based on FCNs have demonstrated significant improvements on several segmentation benchmarks (Cordts *et al*., 2016; Zhou *et al*., 2017; Caesar *et al*., 2018). There are several model variants of FCNs proposed to exploit global features or contextual information for pixel-level classification tasks, including those that employ multi-scale inputs (i.e., image pyramid) or those that adopt probabilistic graphical models (Vemulapalli *et al*., 2016; Chandra and Kokkinos, 2016; Chandra *et al*., 2017).

Encoder-decoder networks have been successfully applied to many computer vision tasks, especially pixel-level image classification (Lin *et al*., 2017; Pohlen *et al*., 2017; Chao *et al*., 2017; Fu *et al*., 2019). In the encoder path, the spatial dimension of feature maps is gradually reduced. Thus, longer range information can be more easily captured in the deeper encoder output. Then, in the decoder path, object details and spatial dimensions are gradually recovered.

In the literature, the following networks have demonstrated the effectiveness of the encoder-decoder structure. The classification algorithm based on a deep deconvolution network proposed by Noh *et al*. (2015), learns the network on top of the convolutional layers adopted from the VGG 16-layer net. The deconvolution network by Noh *et al*. (2015) is composed of deconvolution and unpooling layers, which identify pixel-wise class labels and predict segmentation masks. This algorithm based on instance-wise prediction is advantageous to handle object scale variations by eliminating the limitation of the fixed-size receptive field in the FCN. Badrinarayanan *et al*. (2017) design SegNet, a deep fully convolutional neural network architecture, for pixel-wise image classification. The key component of SegNet is the decoder network consisting of a hierarchy of decoders one corresponding to each encoder. In detail, the appropriate decoders are based on the max-pooling indices received from the corresponding encoder to perform non-linear upsampling of their input feature maps. Ronneberger *et al*. (2015) present U-Net, a network and a training strategy that relies on the strong use of data augmentation to employ the available annotated samples more efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. U-Net is an extension of SegNet by adding skip connections between the encoder and decoder layers such that it works with very few training images and yields more precise classifications.

Spatial pyramid pooling, an extension of the Bag-of-Words (BoW) model (Sivic and Zisserman, 2003), is one of the most successful methods in computer vision (Grauman and Darrell, 2005; Lazebnik *et al*., 2006; He *et al*., 2015). Such a method partitions the image into divisions in which local features are aggregated from finer to coarser levels. The advantages of spatial pyramid pooling are orthogonal to the specific CNN designs. Liu *et al*. (2015) propose ParseNet, an end-to-end simple and effective convolutional neural network, for pixel-level image classification. Exploiting the FCN architecture, ParsetNet can directly use global average pooling from feature maps, resulting in the feature of the whole image as context. Now that ParseNet can capture the context of the image, it can improve local patch prediction results. Zhao *et al*. (2017) propose the pyramid scene parsing network (PSPNet) to incorporate suitable global features. In addition to traditional dilated FCNs for pixel prediction, the pixel-level feature is extended to the specially designed global pyramid pooling one in PSPNet. The local and global clues together make the final prediction more reliable.

In order to capture the contextual information at multiple scales, Chen *et al*. (2017a) propose the atrous spatial pyramid pooling (ASPP) technique, which employs multiple parallel atrous convolution filters with different rates. It is inspired by the success of spatial pyramid pooling (He *et al*., 2015), which showed that it is effective to resample features at different scales for accurately and efficiently classifying regions of an arbitrary scale. The ASPP is capable of collecting multi-level global information. Its structure is illustrated in Fig. 1. Given the specific layer of the convolutional feature map, multiple pyramid scales are applied to generate multi-scale features. The features extracted for each sampling rate are further processed in separate branches and fused as the global feature. Further, Chen *et al*. (2017b) propose to incorporate image-level features encoding global context, which significantly improves performance on a semantic image segmentation benchmark. After that, Chen *et al*. (2018) propose to combine the advantages from the spatial pyramid pooling module and the encode-decoder structure. In particular, the proposed model extends the previous network by adding a simple yet effective decoder module to recover the object boundaries and attains a new state-of-art performance on two public datasets.

Zhang *et al.* (2018b) develop ASPP-Unet that takes advantage of strengths from both the encoder-decoder structure and spatial pyramid pooling for urban land cover classification from high spatial resolution satellite images. In particular, multi-parallel 3×3 atrous convolutions are implemented in a parallel way to the input feature maps of the bridge part in U-Net, and then fused together by sum operation. After that, standard convolution

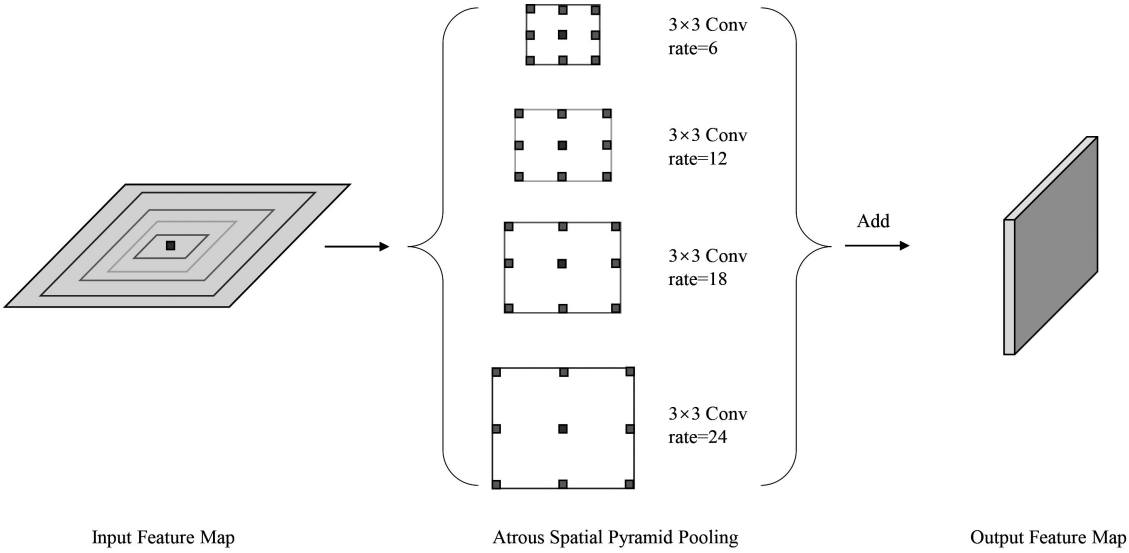


Fig. 1. ASPP exploits multi-scale features by employing multiple parallel filters with different rates.

and ReLU are performed. According to Zhang *et al.* (2018b), ASPP-Unet outperforms the state-of-the-art models, e.g., U-Net, CNN and SVM models over two satellite images, and yields robust and efficient urban land cover classification results. The ASPP-UNet architecture possesses the advantages of capturing fine-grained details, thereby generating better segmentation results than UNet. Therefore, it is promising to exploit the potential of the encoder-decoder structure and the pyramid pooling module to efficiently represent local and global context features for the high spatial resolution satellite images.

3. Methods

There are several algorithms based on FCNs that are applied to object segmentation (Volpi and Tuia, 2017; Marmanis *et al.*, 2016; Marcos *et al.*, 2018). Because the performance of deep learning algorithms depends on their structures, it should be optimized to improve the performance by adjusting and fine-tuning. U-Net, a specific type of FCN, has received a lot of interest for the segmentation of biomedical images using a reduced dataset, but has proven to be also very efficient for the pixel-wise classification of satellite images. To compensate for the shortcomings of U-Net in extracting multi-scale features, we integrate the augmented ASPP into U-Net to obtain a higher segmentation accuracy in high resolution satellite images. In our ASPP module, the output of each pyramid level is combined and upsampled to the same resolution as the input via transposed convolution. After that, the per-pixel prediction is presented. In this section, the details of our network architecture and the network training stage are described.

3.1. Multiscale feature extraction using ASPP.

Atrous convolution can be used to decrease blurring in semantic segmentation maps, and can at least in part extract long range information without the need for pooling (Chen *et al.*, 2017a). Atrous convolution generalizes standard convolution operation and expands the window size without increasing the number of weights or the amount of computations by inserting zero-values into convolution kernels. It thus offers an efficient mechanism to control the field-of-view and finds the best trade-off between accurate localization (small field-of-view) and context assimilation (large field-of-view). In the case of two-dimensional signals, for each location i on output feature map y and convolution filter w, atrous convolution is applied over the input feature map x as follows:

$$y[i] = \sum_k x[i + r \times k]w[k], \tag{1}$$

where the atrous rate r determines the stride with which we sample the input signal. Note that standard convolution is a special case in which rate $r = 1$. The filter's field-of-view is adaptively modified by changing the rate value.

ASPP with different atrous rates effectively reduces context information loss among multiple sub-regions. However, as the sampling rate becomes larger, the number of valid filter weights becomes smaller. In the extreme case where the rate value is close to the feature map size,

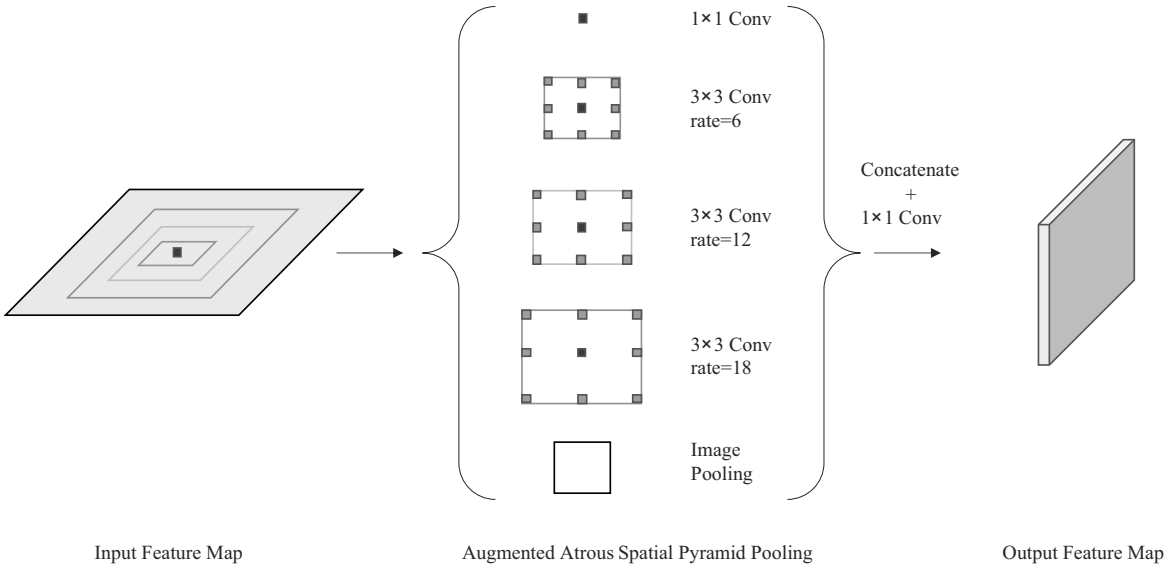


Fig. 2. Augmented ASPP with image-level features.

the 3×3 filter, instead of capturing the whole image context, degenerates to a simple 1×1 filter since only the center filter weight is effective. To overcome this problem and incorporate global context information to our model, similar to the methods used by Liu *et al*. (2015) and Zhao *et al*. (2017), DeepLabv3 (Chen *et al*., 2017b) adopts image-level features. Specifically, global average pooling is applied on the specific feature map of the model. After that, the resulting image-level features are fed to a 1×1 convolution with 256 filters, and then bilinearly upsample the feature to the desired spatial dimension. In the end, the augmented ASPP consists of one 1×1 convolution and three 3×3 atrous convolutions with different rates, besides the image-level features shown in Fig. 2. The resulting features from all the branches are then concatenated as the global feature.

3.2. Proposed deep learning architecture. Because semantic segmentation of satellite images can be treated as a problem of pixel-level classification, it is natural to introduce advanced semantic segmentation architectures to solve such tasks. Instead of developing a model from scratch, we decided to improve an existing model of the CNN for image segmentation. Namely, we turn to U-Net, originally developed for biomedical image segmentation (Ronneberger *et al*., 2015). Basically, U-net builds upon the FCN. A contracting path extracts features of different levels through a sequence of convolutions, rectified linear unit (ReLU) activations and max poolings, allowing to capture the context of each pixel. A symmetric expanding path then upsamples the result to increase the resolution of the detected features. In the U-Net architecture, skip-connections are added between the contracting path and the expanding path, allowing precise localization as well as context. The expanding path therefore consists of a sequence of up-convolutions and concatenations with the corresponding feature map from the contracting path, followed by ReLU activations. The number of features is doubled at each level of downsampling. In this letter, we show that the performance of U-Net can be further improved by substituting the specific layer of the convolutional feature map with augmented ASPP.

To perform multiclass object segmentation, we, inspired by the work of Zhang *et al.* (2018b), develop an encoder-decoder architecture for high spatial satellite images. Our proposed method is illustrated in Fig. 3. The network comprises three parts: the encoder, the bridge and the decoder. The first part encodes the input image into compact representations. The last part recovers the representations to a pixel-wise categorization, that is, semantic segmentation. The middle part serves as a bridge connecting the encoder and decoder paths. The encoder and decoder parts are built with plain neural units which contain two 3×3 convolution blocks. Each convolution block includes a convolutional layer and an ReLU activation layer. The structures of the encoder and decoder parts are symmetrical with skip connections between them, which proves to be effective to produce fine-grained segmentation results.

An input image goes through the encoder part first to generate down-sampled feature maps. Encoder path

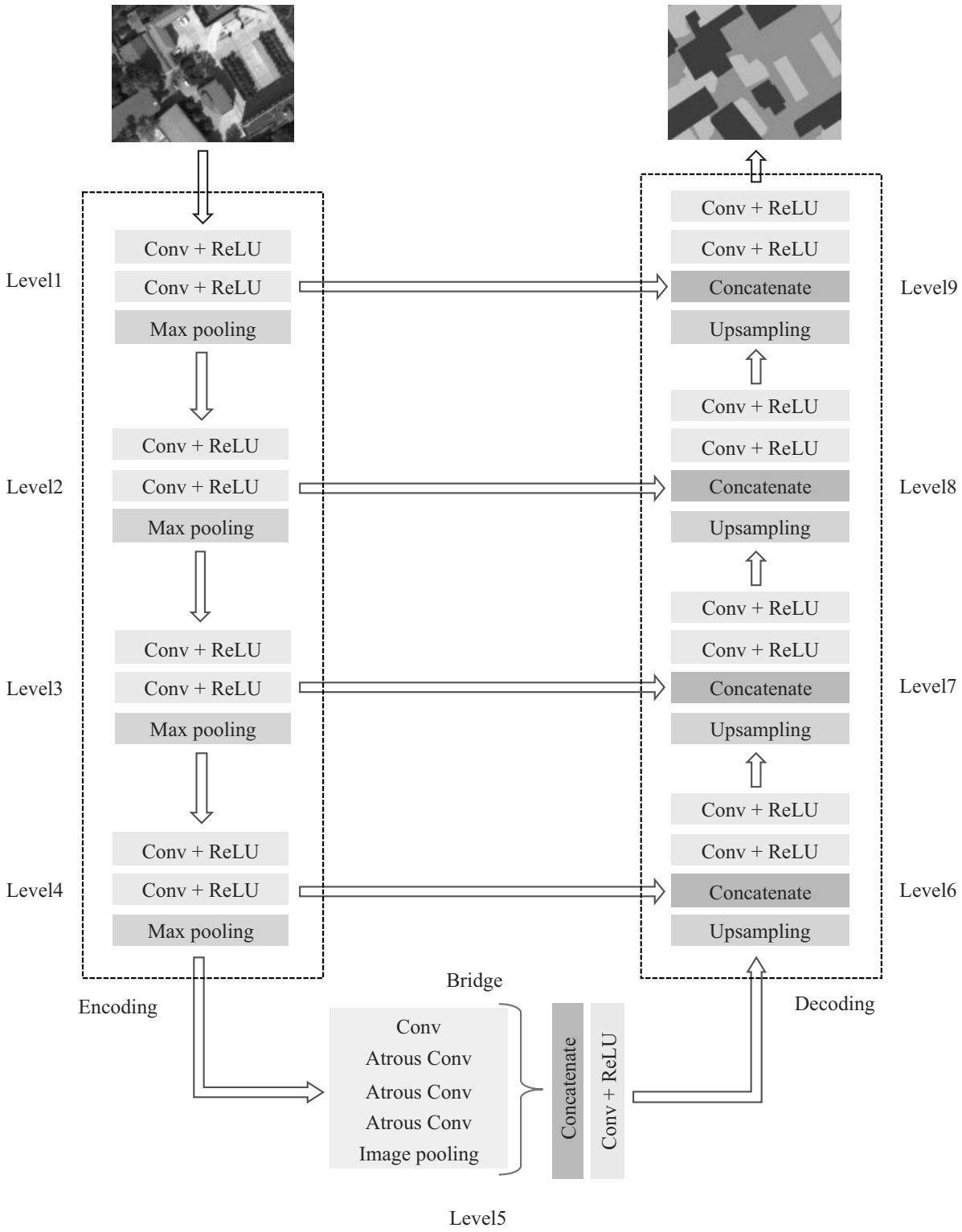


Fig. 3. Augmented ASPP-Unet.

Table 1. Network structure with augmented ASPP.

	Unit level	Conv2D layer		Filter	Output Shape	Number of parameters
Input					$512 \times 512 \times 3$	0
Encoder	Level1	conv2d_1		$3 \times 3/64$	$512 \times 512 \times 64$	1792
		conv2d_2		$3 \times 3/64$	$512 \times 512 \times 64$	36928
	Level2	conv2d_3		$3 \times 3/128$	$256 \times 256 \times 128$	73856
		conv2d_4		$3 \times 3/128$	$256 \times 256 \times 128$	147584
	Level3	conv2d_5		$3 \times 3/256$	$128 \times 128 \times 256$	295168
		conv2d_6		$3 \times 3/256$	$128 \times 128 \times 256$	590080
	Level4	conv2d_7		$3 \times 3/512$	$64 \times 64 \times 512$	1180160
		conv2d_8		$3 \times 3/512$	$64 \times 64 \times 512$	2359808
Bridge	Level5	Augmented ASPP	conv2d_r1	$1 \times 1/256$	$32 \times 32 \times 256$	131072
			conv2d_r2	$1 \times 1/256$	$32 \times 32 \times 256$	131072
			conv2d_r3	$1 \times 1/256$	$32 \times 32 \times 256$	131072
			conv2d_r4	$1 \times 1/256$	$32 \times 32 \times 256$	131072
			conv2d_r5	$1 \times 1/256$	$1 \times 1 \times 256$	131072
		conv2d_9		$3 \times 3/1024$	$32 \times 32 \times 1024$	9438208
Decoder	Level6	conv2d_10		$3 \times 3/512$	$64 \times 64 \times 512$	4719104
		conv2d_11		$3 \times 3/512$	$64 \times 64 \times 512$	2359808
	Level7	conv2d_12		$3 \times 3/256$	$128 \times 128 \times 256$	1179904
		conv2d_13		$3 \times 3/256$	$128 \times 128 \times 256$	590080
	Level8	conv2d_14		$3 \times 3/128$	$256 \times 256 \times 128$	295040
		conv2d_15		$3 \times 3/128$	$256 \times 256 \times 128$	147584
	Level9	conv2d_16		$3 \times 3/64$	$512 \times 512 \times 64$	73792
		conv2d_17		$3 \times 3/64$	$512 \times 512 \times 64$	36928
Output		conv2d_18		$3 \times 3/9$	$512 \times 512 \times 9$	5193

has four plain neural units. In each unit, the pooling operation is used to downsample the feature map size, max-pooling with a 2×2 window is applied to the last convolution block to reduce the feature map by half. Max-pooling is used to achieve translation invariance over small spatial shifts in the input image. Correspondingly, the decoder path consists of four plain neural units, too. Before each unit, there is an up-sampling of feature maps from lower level and a concatenation with the feature maps from the corresponding encoder path. To produce class probabilities for each pixel independently, the high-dimensional feature representation at the output of the final decoder is fed to a 1×1 convolution and a multi-class soft-max classifier

$$loss = -s^{y^{(i)}} + \log \sum_j e^{s^{(j)}} \tag{2}$$

$$s = f(x^{(i)}; w^{(i)}) \tag{3}$$

Given an image dataset, y is the label, x is the input image and w are weights. This soft-max classifies each pixel independently. The output of the soft-max classifier is a K channel image of probabilities where K is the number of classes. The predicted segmentation corresponds to the class with maximum probability at each pixel. The bridge part of the standard U-Net model follows the same structure of the plain neural unit, that is, two sequential convolution and ReLU operations. As shown in Fig. 3, we use the augmented ASPP to replace the first convolution block of the bridge part. As in the work of Zhang *et al.* (2018b), one 1×1 convolution and three 3×3 convolutions with rates equal to (6, 12, 18), and the image-level features are implemented in a parallel way to the input feature maps of the bridge part. The resulting features from all the branches are then concatenated and pass through another convolution block and ReLU operations before the decoder path. Through these procedures, our network retains a large number of feature maps in the encoder path and extend it by the decoder path to refine the segmentation results. In addition, multi-scale deep features are captured by employing augmented ASPP. Altogether, we employ 18 convolutional layers and augmented ASPP layers. It is worth noting that, in our work, the indispensable cropping in U-Net is not required in our network. The parameters and the output size of our network in each step are presented in Table 1.

3.3. Training the network. To train the network, input images and their corresponding segmentation maps are used. The models are implemented by Keras with TensorFlow as the backend, which is powered by

a workstation with Intel Xeon CPU E5-2630V4 (2.2 G\25MB\10 cores\85W), 32GB RAM and a single NVIDIA GTX 1080 Ti GPU. As the architectures are FCN-based models, theoretically, our networks can take an arbitrary-size image as input. However, it will need a certain amount of GPU memory to store the feature maps. In our work, the original images and corresponding labels are adjusted to the specified size same as the training images, 512×512, as described in Table 1. We then utilize fixed-sized training images to train the model. During the training process, the energy function is computed by a pixel-wise soft-max over the final feature map combined with the cross entropy loss function between prediction results and the ground truth. The Adam optimizer is employed with the learning rate $\alpha = 1 \times 10^{-4}$, the exponential decay rates for the moment estimates $\beta 1 = 0.9$ and $\beta 2 = 0.999$. Meanwhile, the constant value $\epsilon = 10^{-8}$ is applied. To avoid overfitting, a dropout strategy is utilized with a probability of 0.5. Due to the limited GPU memory, the batch sizes of the training and validation sets are fixed to 2. With 500 steps per epoch, 100 epochs are iterated over the entire training set in total to train the model.

Data augmentation is widely used in training, including training U-Net. For example, the U-Net by Ronneberger *et al*. (2015) works well after training based on a limit number of samples, provided that data augmentation is used. Accordingly, in order to maintain a reasonable amount of images, and above all to avoid overfitting by ensuring a sufficient invariance and robustness of the network, we followed Ronneberger *et al*. (2015) and applied real time data augmentation techniques to our training set. The satellite images are shifted, flipped and rotated, which allow us to train our model on a considerably larger set of images. The above task is done using the Keras framework which allowed us to augment the data in real time when feeding the network with batches. As a result, there are no memory processes engaged and the network parameters can be better learned from more training sets. In addition, the overfitting effect can be reduced and the generalization ability of the networks can be improved to a large extent. Hence, it is of significance to implement data augmentation so as to improve the segment accuracy. After training, test images could be fed into the trained model to generate prediction results.

4. Experiments

In this section, experiments are carried out to demonstrate the accuracy and efficiency of our networks. We present a description of the EVLab-SS benchmark (Mi and Hu, 2017), which is designed for evaluating the semantic segmentation results on satellite imagery and contains the images captured from different platforms with different types of spatial resolutions. Evaluation metrics are also provided in detail for a quantitative analysis of our method. Based on the selected benchmark, we compare our networks with three state-of-the-art methods, including FCN-8s (Long *et al*., 2015), SegNet (Badrinarayanan *et al*., 2017) and U-Net (Ronneberger *et al*., 2015). Finally, we give a comprehensive analysis of the experimental results.

4.1. Datasets and evaluation metrics. Our datasets are from the EvLab-SS benchmark, which aims to find a good deep learning architecture for the high resolution pixel-wise classification task in the remote sensing area. The EvLab-SS dataset is publicly available at `https://pan.baidu.com/s/1wQ2aLdjCscLNEJsJ1rtgiA`. The average resolution of the dataset is approximately 4500×4500 pixels and each image is fully annotated. There are 11 ground object categories in the dataset, comprising background, farmland, garden, woodland, grassland, building, road, structures, digging pile, desert, and waters. Currently, the dataset includes 60 frames of images with three channels captured by different platforms and sensors. In our experiment, we select two images from different satellites, GeoEye (re-sample GSD 0.5 m) and World-View 2 (re-sample GSD 0.2 m), with a resolution of 5001×5001 pixels from the EvLab-SS dataset. The background and garden classes are absent in the two images, so there are 9 classes in total in the images. We produce our datasets by applying the sliding window with a stride of 128 pixels to the images, thereby resulting in 1406 patches with a resolution of 480×360 pixels for each image. Thus, we have two datasets, both are divided into training, validation and testing sets in ratios of $8 : 1 : 1$, to train and evaluate the proposed model for experiments.

In order to verify the validity of our proposed method, four evaluation metrics are applied based on the comparisons between the prediction semantic maps and the ground truth maps, namely, precision P, recall R, F1-score F_1, and overall accuracy OA. R signifies the ratio of correctly predicted pixels with regard to the total number of pixels in that ground truth class. Here,

$$R = \frac{TP}{TP + FN}, \tag{4}$$

where TP is the number of true positive pixels and FN is the number of false negative pixels. P means the proportion of correctly predicted pixels with regard to the total number of pixels classified as this class in the final prediction.

$$P = \frac{TP}{TP + FP} \tag{5}$$

where FP is the number of false positive pixels. F_1 is

defined by

$$F_1 = 2\frac{P \times R}{P + R} \tag{6}$$

in which P and R are weighted equally. The last measure, OA, represents the ratio of correctly classified test samples globally. It can be seen that F_1 and OA reveal the overall performance, where their larger values show better performance.

4.2. Comparisons.

4.2.1. FCN-8s. Long *et al*. (2015) show that an FCN trained end-to-end, pixels-to-pixels on semantic segmentation exceeds the state-of-the-art net without further machinery. In fact, the above work take the initiative in training FCNs end-to-end for pixelwise prediction from supervised pre-training. The FCN-8s is a new FCN for segmentation that combines layers of the feature hierarchy and refines the spatial precision of the output. Fully convolutionalized classifiers can be fine-tuned to segmentation. Nevertheless, the output of such classifiers is dissatisfyingly coarse. The 32 pixel stride at the final prediction layer limits the scale of detail in the upsampled output. This issue is addressed by adding skips that combine the final prediction layer with lower layers with finer strides. As they see fewer pixels, the finer scale predictions should need fewer layers. Hence, it makes sense to form them from shallower net outputs. Combining fine layers and coarse layers lets the model make local predictions on the premise that the global structure is respected.

To be consistent with FCN-8s, we first divide the output stride in half by predicting from a 16 pixel stride layer. We add a 1×1 convolution layer on top of pool4 to produce additional class predictions. We fuse this output with the predictions computed on the final layer at stride 32 by adding a $2\times$ upsampling layer and summing both predictions. We initialize the $2\times$ upsampling to bilinear interpolation. Finally, the stride 16 predictions are upsampled back to the image. This net is called FCN-16s. FCN-16s is learned end-to-end, initialized with the parameters of the last, coarser net, which is called FCN-32s. The new parameters acting on pool4 are zero-initialized so that the net starts with unmodified predictions. The learning rate is decreased by a factor of 100. We continue in this fashion by fusing predictions from pool3 with a $2\times$ upsampling of predictions fused from pool4 and the final layer, building the net FCN-8s. Training is done by stochastic gradient descent (SGD) with momentum. We use a minibatch size of 2 images and a learning rate of 10^{-4}. We use a momentum of 0.9, a weight decay of 2^{-4}, and the doubled learning rate for biases. We zero-initialize the class scoring layer, as random initialization yielded neither better performance nor faster convergence. Dropout is included where used in the original classifier nets.

4.2.2. SegNet. SegNet (Badrinarayanan *et al*., 2017), primarily inspired by the unsupervised feature learning architecture, is designed to be an efficient architecture for pixel-wise semantic segmentation. The key learning module of SegNet is an encoder-decoder network, followed by a final pixelwise classification layer. SegNet is trained jointly for a supervised learning task. Hence, the decoders are an integral part of the network of SegNet in test time. Meanwhile, the encoder network consists of 13 convolutional layers which correspond to the first 13 convolutional layers in the VGG16 network (Simonyan and Zisserman, 2014) designed for object classification. The fully connected layers can be discarded in order to retain higher resolution feature maps at the deepest encoder output and reduce the number of parameters in the SegNet encoder network significantly compared with other recent architectures. Each encoder layer has a corresponding decoder layer and hence the decoder network has 13 layers. The final decoder output is fed to a multi-class soft-max classifier. Thus, class probabilities for each pixel are produced independently.

In order to train and validate the SegNet model, we use the same version of SegNet by Badrinarayanan *et al*. (2017), which has 4 encoders and 4 decoders. All the encoders in SegNet perform max-pooling and subsampling, while the corresponding decoders upsample its input using the received max-pooling indices. Batch normalization is used after each convolutional layer in both the encoder and decoder networks. No biases are used after convolutions meanwhile no ReLU nonlinearity is present in the decoder network. Further, a constant kernel size of 7×7 over all the encoder and decoder layers is chosen to provide a wide context for smooth labelling. This allows us to train SegNet in reasonable time. We use the same SGD solver with a fixed learning rate of 10^{-4} and a momentum of 0.9. The optimization is performed for more than 100 epochs through the dataset until no further performance increase is observed. A dropout of 0.5 is added to the end of deeper convolutional layers in all models to prevent overfitting. For our two datasets which have 9 classes, we use a mini-batch size of 2.

4.3. Results.

4.3.1. GeoEye results. Based on the whole ground truth dataset, the evaluation results of the test images from GeoEye data are presented in Table 2. It can be seen that the augmented ASPP-UNet achieves the best performance compared with other models with 88.56% in OA. ASPP-UNet yields a slightly lower performance with OA 86.84%. SegNet produces the lowest accuracy

with an OA 81.25%. Concerning the performance of each class, in most cases, precision P, recall R and F1-score F_1 of our method are mostly higher than that of FCN-8s, SegNet, U-Net and ASPP-Unet.

We then compare the results of our model with the peers by using a two-sided Wilcoxon rank sum test (Gibbons and Chakraborti, 2011). Details are given in Table 3. It can be seen that our model is significantly different from SegNet and ASPP-UNet.

To visualize the evaluation results on the test patches with different methods, we demonstrate four examples in Fig. 4. It can be seen that our method leads to an improvement for almost all types of land cover. For example, in the first and third row in Fig. 4, our method is able to predict the building class correctly while the other four approaches show poor performance. Especially SegNet and U-Net produce significant salt-and-pepper noise. For the water objects in the third row, SegNet mistakes water as grassland or woodland, U-Net and ASPP-Unet correctly predict just a part of water. With respect to road and desert, our method shows a good performance in a complex situation as shown in the second row in Fig. 4. The FCN and the ASPP-Unet do not fully predict roads while SegNet and U-Net mix desert and structures. Our network distinguishes road in structures while other methods confuse them. For grassland in the last row, the result by the FCN, the ASPP-Unet and our method all perform well. However, parts of grassland are mistaken as structures in the prediction of SegNet and U-Net.

4.3.2. World-View 2 data. Similarly to the GeoEye data, we predict the test patches in the World-View 2 dataset to evaluate the proposed model. We compare the results with those for FCN, SegNet, U-Net and ASPP-Unet. Table 4 demonstrates the accuracies of all the methods using the whole ground truth. Our network achieves the best performance among the five methods with 86.22% in OA. Specifically, the OAs of the four methods being compared are similar. Except woodland, grassland and structures, the precision of the other land cover types always yields an OA value higher than 80.00%. According to Table 4, with respect to F1 of each class, our method is the best.

Also, we compare the results of our method with the other methods by using a two-sided Wilcoxon rank sum test. Details are given in Table 5. It shows that our model is significantly better from among all of the involved methods.

Four examples of evaluation results on the test patches in the World-View 2 dataset are shown in Fig. 5. With respect to the woodland and road classes, our method is able to predict the road irrespective of whether the woodland is on the road or on the edge of the road according to Fig. 5. However, in the first row, ASPP-Unet

Table 2. Evaluation of results in the GeoEye dataset using the testing set.

	FCN-8s			SegNet			U-Net			ASPP-Unet			Our method		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Farmland	85.66	84.27	84.96	81.66	88.57	84.99	89.10	81.64	85.22	88.96	87.38	88.16	89.47	84.62	86.98
Woodland	84.79	82.35	83.55	86.97	69.18	77.18	83.25	83.84	83.54	84.44	87.37	85.88	87.18	84.34	85.74
Grassland	83.53	83.76	83.64	88.19	78.44	83.06	80.34	87.56	83.81	89.28	85.72	87.47	89.44	86.07	87.73
Building	83.40	87.14	85.23	82.47	84.92	83.67	88.13	82.41	85.19	88.74	86.41	87.56	87.62	86.39	87.00
Road	84.77	82.28	83.50	86.69	82.28	84.43	83.44	87.87	85.60	88.98	86.75	87.85	89.66	84.46	86.99
Structures	85.45	83.94	84.69	77.95	85.25	81.45	82.41	86.16	84.25	86.83	86.87	86.85	85.45	87.93	86.68
Digging pile	85.92	89.85	87.85	88.76	89.73	89.24	88.91	89.70	89.30	89.63	89.31	89.47	89.97	86.89	88.41
Desert	88.73	83.77	86.19	89.96	73.36	80.91	75.61	77.84	76.71	89.60	87.71	88.65	89.33	88.31	88.82
waters	85.83	84.81	85.32	83.00	88.36	85.60	82.41	88.41	85.32	83.09	89.94	86.39	87.27	87.44	87.36
OA			84.6			81.25			84.37			86.84			88.56

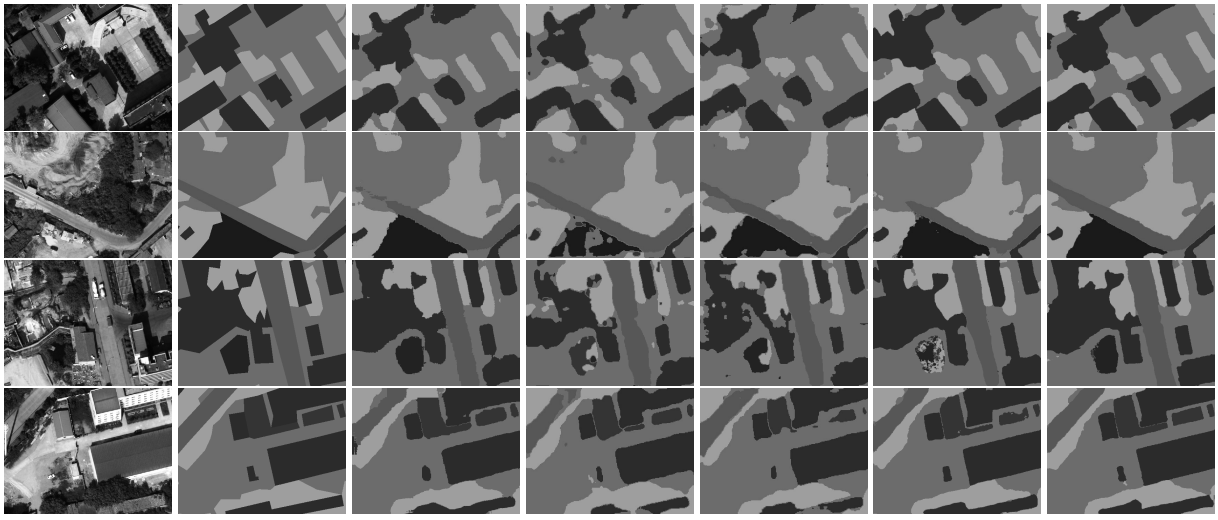


Fig. 4. GeoEye data classification results with different methods on the test patches. The original images are shown in column (a), while ground truth images are shown in column (b). Results by FCN, SegNet, U-Net, ASPP-Unet, and our method are displayed in columns (c)–(g), respectively

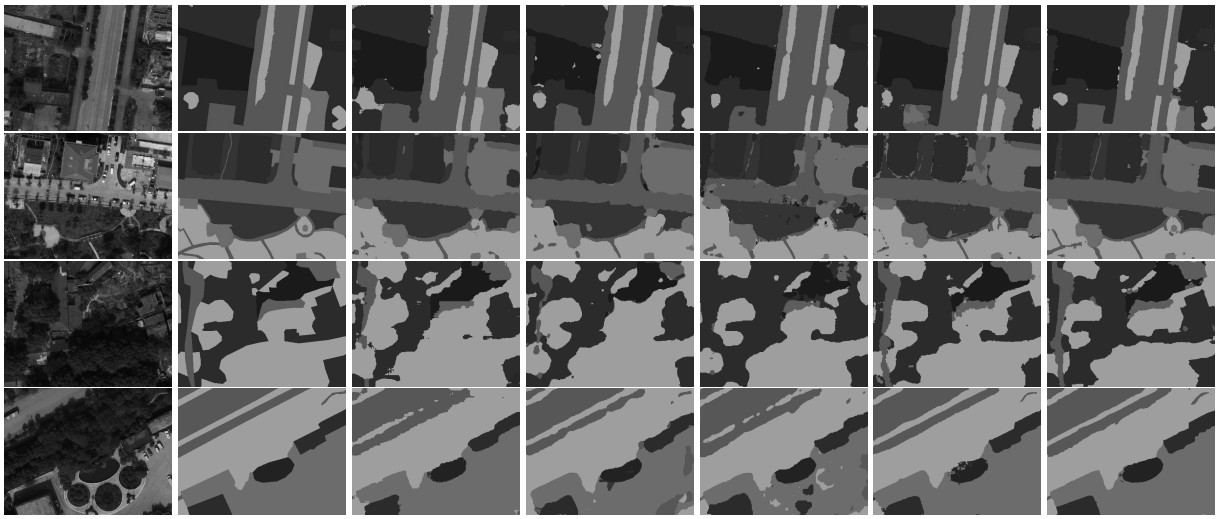


Fig. 5. World-View 2 data classification results with different methods on the test patches. The original images are shown in column(a), while ground truth images are shown in column(b). Results by FCN, SegNet, U-Net, ASPP-Unet, and our method are displayed in columns (c)–(g), respectively.

falsely label road as structures. In the third row, U-Net does not predict the road at all. Besides, the FCN and the U-Net cannot distinguish the boundaries between woodland and road clearly in the second and last rows. From the perspective of farmland, our method shows an improvement compared with the other methods. For instance, in the third column in Fig. 5, our method correctly predicts farmland. However, SegNet mistakes farmland as woodland. The other methods recognize part of farmland with a rough boundary. In terms of water, we observe that the predictions by all the methods are accurate, although woodland around water has similarities in color.

4.4. Discussion. The effectiveness of the proposed method is comprehensively examined based on the high resolution satellite image datasets. Further, the superiority of the proposed method is verified through the quantitative and qualitative analysis against several deep learning based methods. Compared with the U-Net model, ASPP-Unet incorporates multiple atrous convolutions in

Table 3. Wilcoxon test results for the experiment based on the GeoEye dataset. The result p is to test the null hypothesis at the 5% significance level while h is a logical value indicating the test decision: $h = 1$ indicates a rejection of the null hypothesis and $h = 0$ indicates a failure to reject the null hypothesis.

	FCN-8s	SegNet	U-Net	ASPP-Unet
p	0.0020	0.0503	0.0102	0.4363
h	1	0	1	0

the bottom layer, and thus increases the receptive fields compared to the 3×3 convolution in U-Net and enables capturing multi-scale features. Further, we improve the ASPP-Unet model by adding global average pooling in ASPP. This indicates that object contextual features at different spatial scales can be represented with an increasingly enlarged field-of-view. It should be noted that, due to the usage of ASPP with image-level features, the proposed approach is robust to the types of land which not only are different in scales and sizes, but also range from a narrow shape road to a large area building. This means that our method can capture multi-scale object information, which is critical for segmenting objects with sharp changes in sizes and scales on high resolution satellite images. It intuitively can be seen from Figs. 4 and 5 that our method can better delineate the boundary of an object. The results demonstrate the superiority of the utilization of augmented ASPP, which can more accurately classify each pixel with varying spatial resolutions.

Although the proposed network shows improvements in the accuracy and object boundaries, there are still several potential limitations. First, the input images of our network contain only three bands. Although most cases in the computer vision field deal with color images just containing the three channels (red, green, and blue), high resolution satellite images generally contain four bands, i.e., red, green, blue and infrared bands. In fact, we only use the three bands, red, green and blue, in our experiments. Furthermore, because the proposed architecture contains millions of parameters, a large number of training samples are needed. Due to different sizes and locations of the object changes, it is quite labor-intensive to obtain enough ground truth maps with high accuracy. Thus, recently developed deep learning techniques, such as transfer learning, reinforcement learning and weakly supervised learning, should be exploited for improving our network to solve the issues of a limited number of training samples.

Table 4. Evaluation of results in the World-View 2 dataset using the testing set.

	FCN-8s			SegNet			U-Net			ASPP-Unet			Our method		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Farmland	87.38	66.08	75.42	81.44	75.15	78.18	84.47	81.67	83.05	82.97	77.91	80.37	88.33	73.03	84.03
Woodland	76.23	86.84	81.22	79.66	84.55	82.04	85.24	78.80	81.91	86.10	77.57	81.64	86.57	80.26	83.31
Grassland	82.08	86.45	84.21	79.74	87.47	83.44	81.50	87.67	84.48	82.61	86.35	84.44	86.18	87.73	86.95
Building	89.46	74.51	81.38	83.33	86.42	84.85	81.00	88.44	84.58	82.63	88.24	85.35	81.81	88.68	86.12
Road	86.40	88.65	87.51	86.60	86.55	86.58	86.67	86.69	86.68	86.53	83.61	85.05	85.59	89.27	88.40
Structures	74.08	88.47	80.71	84.30	74.78	79.29	84.40	77.27	80.69	77.83	86.99	82.12	85.65	74.18	82.17
Digging pile	82.12	88.82	85.35	86.33	77.75	81.84	87.51	75.74	81.25	89.34	81.58	85.31	89.63	83.50	86.47
Desert	80.11	89.39	84.52	87.83	82.31	84.99	88.61	83.67	86.08	88.20	88.05	88.13	89.79	85.45	88.57
waters	89.87	73.45	80.93	89.11	69.23	78.06	87.24	77.74	82.25	86.79	89.65	86.20	87.99	86.02	87.00
OA		84.72			84.59			83.87			84.25			86.22	

5. Conclusion

In this paper, we propose a classification algorithm for high spatial resolution satellite images based on

Table 5. Wilcoxon test results for the experiment based on the World-View2 dataset. The resulting p-value is to test the null hypothesis at the 5% significance level, while h is a logical value indicating the test decision. Here $h = 1$ indicates a rejection of the null hypothesis and $h = 0$ indicates a failure to reject the null hypothesis.

	FCN-8s	SegNet	U-Net	ASPP-Unet
p	0.2581	0.1359	0.1903	0.2973
h	0	0	0	0

U-Net. We use U-Net as our key learning module. In detail, a contracting path is used to encode the rich contextual information, while a symmetric expanding path is adopted to recover the object boundaries. At the bridge part, we use ASPP with image-level features to learn multi-scale feature maps from different semantic levels. The effectiveness of the proposed method is elaborately examined based on the experiments on two satellite image datasets. Our experiments show that the proposed approach yields the best performance on both visual comparison and quantitative metrics evaluation compared with other deep learning methods. Therefore, it is promising to exploit the potential of our models for multiclass object segmentation on satellite images. However, like any other deep learning architectures, the proposed architecture requires a large number of ground truth samples, which limits the widespread use in the real world application to a certain extent. In the future, weakly supervised learning and samples generation techniques may be developed to improve the applicability of our model for semantic segmentation tasks. Besides, the super-pixel segmentation can be applied as a pre-processing step to reduce the number of optimization elements in the proposed model.

Acknowledgment

The research was funded by the project of the Land and Resources Geographic Information Center of the Ningxia Hui Autonomous Region, *Comprehensive Supervision and Analysis of Large Spatial Data of Natural Resources Based on Deep Learning*.

References

Badrinarayanan, V., Kendall, A. and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation, *IEEE Transactions on Geoscience & Remote Sensing* **39**(12): 2481–2495.

Bei, Z., Bo, H. and Zhong, Y. (2017). Transfer learning with fully pretrained deep convolution networks for land-use classification, *IEEE Geoscience & Remote Sensing Letters* **PP**(99): 1–5.

Caesar, H., Uijlings, J. and Ferrari, V. (2018). Coco-stuff: Thing and stuff classes in context, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA,* pp. 1209–1218.

Carreira, J., Rui, C., Batista, J. and Sminchisescu, C. (2012). Semantic segmentation with second-order pooling, *European Conference on Computer Vision, Firenze, Italy*, pp. 430–443.

Carreira, J. and Sminchisescu, C. (2011). CPMC: Automatic object segmentation using constrained parametric min-cuts, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(7): 1312–1328.

Castelluccio, M., Poggi, G., Sansone, C. and Verdoliva, L. (2015). Land use classification in remote sensing images by convolutional neural networks, *Acta Ecologica Sinica* **28**(2): 627–635.

Chandra, S. and Kokkinos, I. (2016). Fast, exact and multi-scale inference for semantic image segmentation with deep Gaussian CRFS, *European Conference on Computer Vision, Amsterdam, The Netherlands*, pp. 402–418.

Chandra, S., Usunier, N. and Kokkinos, I. (2017). Dense and low-rank Gaussian CRFs using deep embeddings, *IEEE International Conference on Computer Vision, Honolulu, HI, USA,* pp. 5103–5112.

Chao, P., Zhang, X., Gang, Y., Luo, G. and Jian, S. (2017). Large kernel matters—Improve semantic segmentation by global convolutional network, *IEEE Conference on Computer Vision and Pattern Recognition, Venice, Italy,* pp. 4353–4361.

Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A.L. (2017a). DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, *IEEE Transactions on Pattern Analysis & Machine Intelligence* **40**(4): 834–848.

Chen, L.-C., Papandreou, G., Schroff, F. and Adam, H. (2017b). Rethinking atrous convolution for semantic image segmentation, *arXiv* 1706.05587.

Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F. and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation, *Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany,* pp. 801–818.

Cleve, C., Kelly, M., Kearns, F.R. and Moritz, M. (2008). Classification of the wildland–urban interface: A comparison of pixel- and object-based classifications using high-resolution aerial photography, *Computers Environment & Urban Systems* **32**(4): 317–326.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S. and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA,* pp. 3213–3223.

Fu, J., Jing, L., Wang, Y. and Lu, H. (2019). Stacked deconvolutional network for semantic segmentation, *IEEE Transactions on Image Processing* **PP**(99): 1–1.

Fulkerson, B., Vedaldi, A. and Soatto, S. (2009). Class segmentation and object localization with superpixel neighborhoods, *IEEE International Conference on Computer Vision, Kyoto, Japan,* pp. 670–677.

Gang, C., Weng, Q., Hay, G.J. and He, Y. (2018). Geographic object-based image analysis (geobia): Emerging trends and future opportunities, *GIScience & Remote Sensing* **55**(2): 159–182.

Gibbons, J. and Chakraborti, S. (2011). The Wilcoxon rank-sum test and confidence interval, *Nonparametric Statistical Inference* **59**(4): 290–293.

Gong, C., Han, J., Lei, G., Liu, Z., Bu, S. and Ren, J. (2015). Effective and efficient midlevel visual elements-oriented land-use classification using VHR remote sensing images, *IEEE Transactions on Geoscience & Remote Sensing* **53**(8): 4238–4249.

Grauman, K. and Darrell, T. (2005). Pyramid match kernels: Discriminative classification with sets of image features, *10th IEEE International Conference on Computer Vision, Beijing, China,* pp. 1458–1465.

He, K., Zhang, X., Ren, S. and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Transactions on Pattern Analysis & Machine Intelligence* **37**(9): 1904–16.

Kim, J.H., Lee, H., Hong, S.J., Kim, S., Park, J., Hwang, J.Y. and Choi, J.P. (2018). Objects segmentation from high-resolution aerial images using U-Net with pyramid pooling layers, *IEEE Geoscience and Remote Sensing Letters* **16**(1): 115-119.

Lazebnik, S., Schmid, C. and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, NY, USA,* pp. 2169–2178.

Lin, G., Milan, A., Shen, C. and Reid, I. (2017). Refinenet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation, *IEEE International Conference on Computer Vision, Venice, Italy,* pp. 1925–1934.

Liu, W., Rabinovich, A. and Berg, A. (2015). Parsenet: Looking wider to see better, *arXiv* 1506.04579.

Long, J., Shelhamer, E. and Darrell, T. (2015). Fully convolutional networks for semantic segmentation, *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA,* pp. 3431–3440.

Maggiori, E., Tarabalka, Y., Charpiat, G. and Alliez, P. (2016). Convolutional neural networks for large-scale remote sensing image classification, *IEEE Transactions on Geoscience & Remote Sensing* **55**(2): 645–657.

Marcos, D., Volpi, M., Kellenberger, B. and Tuia, D. (2018). Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models, *ISPRS Journal of Photogrammetry and Remote Sensing* **145**(5): 96–107.

Marmanis, D., Schindler, K., Wegner, J., Galliani, S., Datcu, M. and Stilla, U. (2016). Classification with an edge: Improving semantic image segmentation with boundary detection, *ISPRS Journal of Photogrammetry and Remote Sensing* **135**(7): 158–172.

Mi, Z. and Hu, X. (2017). Learning dual multi-scale manifold ranking for semantic segmentation of high-resolution images, *Remote Sensing* **9**(5): 500.

Miao, L., Zang, S., Bing, Z., Li, S. and Wu, C. (2014). A review of remote sensing image classification techniques: The role of spatio-contextual information, *European Journal of Remote Sensing* **47**(1): 389–411.

Noh, H., Hong, S. and Han, B. (2015). Learning deconvolution network for semantic segmentation, *IEEE International Conference on Computer Vision, Santiago, Chile,* pp. 1520–1528.

Peng, D., Zhang, Y. and Guan, H. (2019). End-to-end change detection for high resolution satellite images using improved UNet++, *Remote Sensing* **11**(11): 1382.

Pohlen, T., Hermans, A., Mathias, M. and Leibe, B. (2017). Full-resolution residual networks for semantic segmentation in street scenes, *IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA,* pp. 4151–4160.

Razavian, A.S., Azizpour, H., Sullivan, J. and Carlsson, S. (2014). CNN features off-the-shelf: An astounding baseline for recognition, *IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA,* pp. 806–813.

Ren, S., Girshick, R., Girshick, R. and Sun, J. (2017). Faster r-CNN: Towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis & Machine Intelligence* **39**(6): 1137–1149.

Ronneberger, O., Fischer, P. and Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation, *International Conference on Medical Image Computing & Computer-Assisted Intervention, Munich, Germany,* pp. 234–241.

Scott, G.J., England, M.R., Starms, W.A., Marcum, R.A. and Davis, C.H. (2017). Training deep convolutional neural networks for land-cover classification of high-resolution imagery, *IEEE Geoscience & Remote Sensing Letters* **14**(9): 1638–1642.

Sharma, A., Liu, X., Yang, X. and Shi, D. (2017). A patch-based convolutional neural network for remote sensing image classification, *Neural Networks* **95**(7): 19.

Shotton, J., Johnson, M. and Cipolla, R. (2008). Semantic texton forests for image categorization and segmentation, *Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA,* pp. 1–8.

Shotton, J., Winn, J., Rother, C. and Criminisi, A. (2009). Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context, *International Journal of Computer Vision* **81**(1): 2–23.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition, *arXiv* 1409.1556.

Sivic and Zisserman (2003). Video Google: A text retrieval approach to object matching in videos, *Proceedings of the 9th IEEE International Conference on Computer Vision, Nice, France*, Vol.2, pp. 1470–1477.

Tao, L., Abd-Elrahman, A., Morton, J. and Wilhelm, V.L. (2018). Comparing fully convolutional networks, random forest, support vector machine, and patch-based deep convolutional neural networks for object-based wetland mapping using images from small unmanned aircraft system, *GIScience & Remote Sensing* **55**(2): 243–264.

Vemulapalli, R., Tuzel, O., Liu, M.Y. and Chellappa, R. (2016). Gaussian conditional random field network for semantic segmentation, *Computer Vision & Pattern Recognition, Las Vegas, NV, USA,* pp. 3224–3233.

Volpi, M. and Tuia, D. (2017). Dense semantic labeling of subdecimeter resolution images with convolutional neural networks, *IEEE Transactions on Geoscience and Remote Sensing* **55**(2): 881–893.

Yang, H., Yu, B., Luo, J. and Chen, F. (2019). Semantic segmentation of high spatial resolution images with deep neural networks, *GIScience & Remote Sensing* **56**(5): 1–20.

Zhang, C., Xin, P., Li, H., Gardiner, A. and Atkinson, P.M. (2018a). A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification, *ISPRS Journal of Photogrammetry & Remote Sensing* **140**(7): 133–144.

Zhang, P., Ke, Y., Zhang, Z., Wang, M., Li, P. and Zhang, S. (2018b). Urban land use and land cover classification using novel deep learning models based on high spatial resolution satellite imagery, *IEEE Transactions on Geoscience & Remote Sensing* **18**(11): 3717.

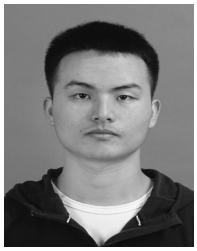
Zhao, H., Shi, J., Qi, X., Wang, X. and Jia, J. (2017). Pyramid scene parsing network, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA*, pp. 6230–6239.

Zhao, W. and Du, S. (2016). Learning multiscale and deep representations for classifying remotely sensed imagery, *ISPRS Journal of Photogrammetry & Remote Sensing* **113**(3): 155–165.

Zhou, B., Hang, Z., Puig, X., Fidler, S., Barriuso, A. and Torralba, A. (2017). Scene parsing through ADE20K dataset, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA,* pp. 633–641.

Zhuowen, T. and Xiang, B. (2010). Auto-context and its application to high-level vision tasks and 3D brain image segmentation, *IEEE Transactions on Pattern Analysis & Machine Intelligence* **32**(10): 1744–1757.

Yong Wang received his PhD degree from the China University of Geosciences, Wuhan, where he is currently an associate professor with the School of Computer Science. His present research interests include parallel storage and processing of spatial data, and the understanding of remote sensing images.



Dongfang Zhang received his BSc degree from the China University of Geosciences, Wuhan, where he is currently pursuing the PhD degree at the School of Computer Science. His present research interest includes the understanding of remote sensing images.

Guangming Dai received his PhD degree in computer software and theory from the Huazhong University of Science and Technology, Wuhan, China, in 2004. He is currently a professor with the School of Computer Science, China University of Geosciences. His present research interests include the algorithm design, multi-objective optimization, and visualization.

Received: 11 October 2019
Revised: 18 March 2020
Re-revised: 10 May 2020
Accepted: 29 May 2020