POSITION-SENSITIVE ATTENTION BASED ON FULLY CONVOLUTIONAL NEURAL NETWORKS FOR LAND COVER CLASSIFICATION

Zirou Xiong, Zongqian Zhan*, Xin Wang

School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China
xzirou@whu.edu.cn, (zqzhan, xwang)@sgg.whu.edu.cn

Commission III, WG III/7

KEY WORDS: Land cover classification, semantic segmentation, skip connection, position-sensitive attention, remote sensing images

ABSTRACT:

Pixel-wise land cover classification is a fundamental task in remote sensing image interpretation, aiming to identify planimetric features (e.g., trees, waters, buildings etc.) from earth's surface. Recently, deep learning methods based on fully convolutional neural networks (FCN) become the mainstream approach for land cover classification, thanks to their superior performance in the image context perception and features learning. However, for high-resolution remote sensing images with huge quantity of object details, some deep learning based methods often ignore many important details by nature, specially, in the procedure of pooling operation and stacking convolutions in conventional FCN, it can leads to ambiguous classification of adjacent objects. To refine lost details caused by the stacking convolutions, we propose a position-sensitive attention (*PSA*) based on skip connections for land cover classification with high-resolution remote sensing images, which designs to deliver a weight that is sensitive to the spatial details in remote sensing images, the *PSA* module is able to improve pixel-level details scattered across spatial positions. Experimental results demonstrate that our method can be feasible to existing FCN-based models, 1% improvement in F1-score is obtained on 2021 "Shengteng Cup" competition dataset after using *PSA*, when comparing to several state-of-the-art methods, similar or even better performance is achieved on the ISPRS Vaihingen 2D dataset, but with less parameters.

1. INTRODUCTION

Land cover classification aims to locate geographic objects at the pixel-level and assign them with feature category labels (Yang et al., 2019 and 2020), this is in principle identical with the goal of semantic segmentation in computer vision which explores dense pixel-wise classification. As some relevant technologies (such as sensors, electronics spacecraft etc.) develop, remote sensing images from various platforms such as satellites, airplanes and drones can observe large area with high resolution, and act as an important data source which is widely studied for land cover classification (Kussul et al., 2017; Zhan et al., 2020; Deng et al., 2020). However, abundant details and large variations of objects in high-resolution images spawn new challenges for the extraction of features (Zheng et al., 2020a) for land cover classification. Recently, Convolutional Neural Networks (CNN) have been widely applied in land cover classification tasks, thanks to its outstanding performance in spatial feature extraction (Chen et al., 2018a). Inspired by the superior achievement of the first end-to-end fully convolutional neural networks (FCN), variants of relevant networks based on the encoder-decoder architecture outperform traditional machine learning algorithms, e.g., Wavelet Transform (Myint et al. 2004), SVM and MRF (Tarabalka et al. 2010; Camps-Valls et al. 2014) etc., and become the mainstream approach for land cover classification (Long et al., 2015; Badrinarayanan et al., 2017; Demir et al., 2018).

Over the last years, image classification record has been continuously broken by various networks, for example, *VGGNet* (Simonyan et al., 2014), *GoogleNet* (Szegedy et al., 2015) and *ResNet* (He et al., 2016), the discriminability of the extracted features from these networks have been gradually increased, while one of remaining problems is the lost details during encoding procedure caused by the pooling operation and convolution operations across pixels (Yang et al., 2020). To cope with details lost by the pooling operation, dilated convolutions were suggested to replace the pooling layer for down-sampling (Yu et al., 2016), and the feature maps from encoder and the corresponding output of decoder are spliced on channel dimension, some details are reintroduced through skip connections (Ronneberger et al., 2016). However, dilated convolution is a sparse operation, i.e., stacking of dilated convolutions results in “gridding issue” (Wang et al., 2018) which appears as gridding shadow on the prediction results, and skip connections cannot transport sufficient semantic information from encoder, leading to misclassification of some pixels. Sun et al. (2019) proposed the advanced high-resolution networks (*HRNet*) which makes the pooling layer out of the model, but the structural complexity and the number of parameters increased. In addition, details are still not recovered somehow when stacking convolutions with stride of 2 for down-sampling in *HRNet*, thus, the problem of lost details caused by convolution operations across pixels remains unresolved. Recently, attention based methods are demonstrated to be effective for context information collection as a weighting mechanism (Wang et al., 2017; Fu et al., 2019; Tao et al., 2020), standing on these previous works, the corresponding weighting capability in spatial positions is investigated in this paper to refine details lost by convolution operations.

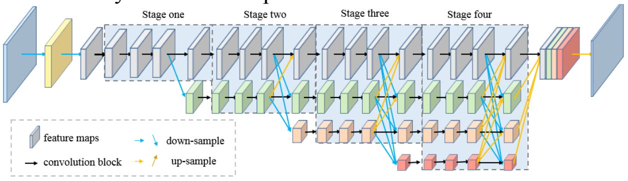

Figure 1. A simple example of the high-resolution networks. There are four stages. The 1st stage consists of a high-resolution branch (branch one), each stage adds branches in order (branch two to four). More details can be found in Section 3 (Sun et al., 2019).

In this paper, we investigate the feature maps extracted from different branches of *HRNet* at different stages by visualizing the corresponding response peaks on the channel dimension (Fig. 1 and Fig. 2). In general, it can be found that as the stage of network layers become deeper and the number of branches increases, richer semantic information can be contained by the corresponding feature map, and more detailed information will be lost as well. On

*Corresponding author

the contrary, feature maps at earlier stages include more detailed information. In order to restore the lost details, we propose a position-sensitive attention (*PSA*) that can capture the spatial details from one of those feature maps which are from one early stage, and integrate it into later stages with skip pathways. Some feature maps extracted from other networks (such as *ResNet*) are also studied, basically, the same finding appears on corresponding visualization of the similar feature maps, and the experimental results demonstrate that our attention module can be applied to other networks as well. In addition, comparison experiments with several state-of-the-art methods on different remote sensing image datasets show that our method can achieve similar or better results with less parameters. The contributions of this work are as follows:
1. We propose a simple yet effective *PSA* module which is designed to take care of spatial details of remote sensing images.
2. The proposed attention can be easily applied to various models that derived from FCN backbones by skip connection.
3. Comparable performance can be achieved after integrating with our attention, but with less cost.

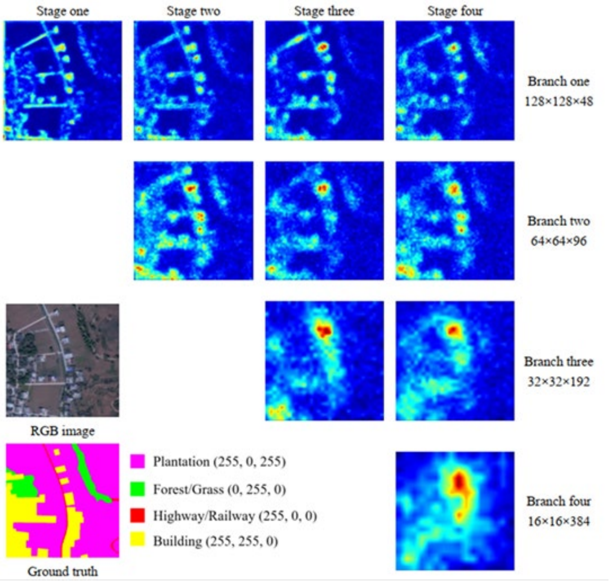

Figure 2. Example for feature map visualizations in *HRNet* (Fig. 1). Each of them is from the last feature map of the corresponding stage and branch, and up-sampled to the same size as the input image (i.e., 512×512).

The rest of this paper is organized as follows. Related works are reviewed in Section 2. The details of our attention are illustrated in Section 3. The performance of our works on different datasets and the corresponding ablation experiments are reported in Section 4. Finally, conclusions and an outlook are drawn in Section 5.

2. RELATED WORK

Extensive works have been researched on land cover classification or semantic segmentation by employing the variants of FCN (Zhu et al., 2017, Chen et al., 2018a). In this section, we briefly review works on models based on FCN for semantic segmentation, architectures based on skip connection, and attention mechanism relevant to our works.

2.1 FCN and relevant networks

FCN removes the fully connected layers in CNN for image level classification, and replace it with up-sample and generate pixel-by-pixel prediction to achieve end-to-end output (Long et al., 2015). By considering the relatively simple deconvolution procedure of FCN, Noh et al. (2015) proposed the deconvolution networks that is symmetric to the convolution network for up-sampling (i.e., unpooling to pooling, deconvolution to convolution). Similar symmetrical encoder-decoder architecture is applied in *SegNet* (Badrinarayanan et al., 2017), in which encoder is applied for down-sampling to expand the receptive field and feature extraction, while decoder is used for up-sampling and information fusion. As multi-scale analysis plays an important role in learning global information, pyramid architectures are deployed in Feature Pyramid Network (Lin et al., 2017) and Pyramid Scene Parsing Network (Zhao et al., 2017). Combining the feature pyramid and the dilated convolution (Yu et al., 2016), *Deeplab* (Chen et al., 2018a) applied Atrous Spatial Pyramid Pooling (*ASPP*) module and achieved great performance. In order to improve the results of the semantic segmentation networks for better classification, *HRNet* (Sun et al., 2019) continuously performs down-sampling, feature extraction, up-sampling, and features are fused in sequence without encoder-decoder architecture and 3×3 convolutions with stride of 2 are applied for down-sampling as a replacement.

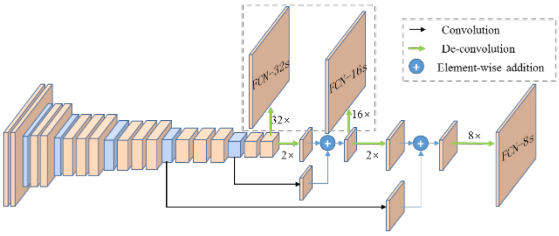

Figure 3. A simple example of the high-resolution networks.

2.2 Skip connection

Skip connection is a universal architecture used in CNN to convey skip information, which has been proved to be effective in pixel-wise classification. As shown in Fig. 3, FCN (Long et al., 2015) obtains feature maps with decreasing size and increasing number of channels by stacking convolutional layers and pooling layers, and feature maps are generated during down-sampling. In order to achieve end-to-end and pixel-by-pixel classification results, feature map from the bottom layer is up-sampled by zero-padded transposed convolution. Furthermore, element-wise addition is applied to merge multi-layer outputs by skip connections, so that FCN is able to recover the lost details caused by pooling layers to some extent. Different from element-wise addition in FCN, *UNet* (Ronneberger et al., 2015) combines feature maps from symmetrical positions of encoder and decoder on channel dimension to restore details through skip connections. Both of the two strategies have shown the skip connection is indeed helpful to improve land cover classification. In addition to the role of delivering details, skip connection is also considered as a shortcut in residual networks (*ResNet*, He et al., 2016) to keep gradient (feature map with small response value make gradient vanish). As shown in Fig. 4, if the convolution block is not helpful for improving the networks, it will be learned to become zero kernels, achieving the effect that its performance is at least not worse than the original block. As a consequence, *ResNet* makes the networks deeper by solving the problem of vanishing gradient.

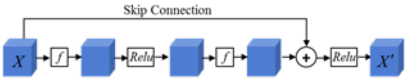

Figure 4. Example for Residual Block, where $X, f, Relu, X^{'}, \oplus$ denote the input feature map, convolution operation, the *Relu* activation function, the output feature map and the element-wise addition.

2.3 Attention mechanism

Attention is essentially a weighting mechanism, networks can adaptively weight different features with learned attention, so that negative features will be suppressed and positive features will be enhanced. *SENet* (Hu et al., 2020) uses global pooling and fully connected layers for a feature map to calculate the relationship between different channel features, and then weights them regarding the channel dimension. To capture long-range dependencies, the *Non-local* module (Wang et al., 2017) flattens two spatial dimensions of a feature map into one dimension, then multiplies the flat and its transposition as the global weight. As a combination of *SENet* and the *Non-local* module, the *CBAM* module (Woo et al., 2018) adds the results of global maximum pooling and global average pooling on each channel to implement channel attention, and splices the results of channel maximum pooling and channel average pooling on each pixel to capture spatial attention. Recently, attention mechanism has been applied to various feature maps, Mao et al. (2020) use attention from the deepest feature map to weight feature maps delivered by skip connections in *UNet*. Via analyzing the extracted feature map, in the work of Zheng et al. (2020b), foreground-aware attention is injected into feature maps from decoder to balance information between foreground and background. Being aware of the unbalanced information at different scales, Tao et al. (2020) explore the relationship between input images at different scales to build their attention, and Dai et al. (2021) apply multi-scale channel attention to weight feature maps at different scales.

Different to the FCN based methods mentioned in section 2.1, we integrate a positive-sensitive attention which is based on selected feature maps extracted from the networks of FCN and the general idea of skip connection is applied.

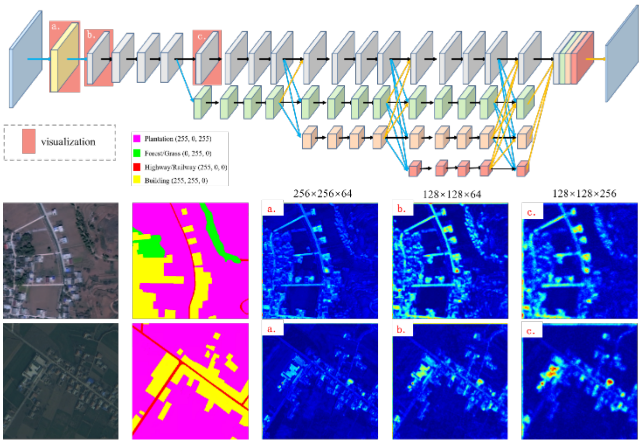

Figure 5. The visualizations of three selected feature maps (*a, b, c*) in *HRNet* with different two input images, one of which is the same as Fig. 2.

3. METHODOLOGY

To restore the lost details during stacking convolutions, we visualize feature maps from the FCN-based networks to find the one that is most identical to human interpretation, and generate the position-sensitive attention (*PSA*) from it. With skip connections, we integrate *PSA* into feature maps which are insensitive to positional details.

3.4 Visualization of feature map

First of all, we visualize the feature map with pixel response peaks on the channel dimension and up-sample them to the size of input as a heatmap. In this paper, the feature maps of *HRNet* (Sun et al., 2019) are selected as examples for visualization and investigation.

HRNet applies 3×3 convolutions with stride of 2 for down-sampling throughout the entire process, and there is no pooling operation, which avoids loss of detail and expands the receptive field to obtain more contextual information at the same time. After using convolutions to perform two consecutive double down-sampling, it starts to enter the multi-resolution branches of *HRNet*. As shown in the Fig.1, *HRNet* uses four stages to extract features, while down-sampling is applied at the end of each stage. As a result, *HRNet* maintains a maximum of four branches to extract feature at various resolution in parallel. In the process of deep feature extraction, element-wise addition is used to integrate deep features form different resolution branches at the end of each stage (except the first stage), which make the final feature map be fused with high-level semantic information and low-level semantic information. With less lost details and fusion of multi-scale local information, *HRNet* can effectively improve the discriminability of the final feature map.

3.2 Selection of the feature map

Considering the requirement of relatively clear boundaries with less loss of detail, we pick three from those feature maps of *HRNet* (Fig. 5). In fact, we get similar heatmaps form other networks (e.g., *ResNet*, Fig. 6) with different input images. As the main segmentation process is performed on the feature map that is $\frac{1}{8}$ the size of the input (same as branch one), and the visualization of the feature map is expected to contain relatively clear boundaries together with distinctive semantic information, we choose the middle one (*b*, Fig. 5) to generate our attention. Moreover, we conduct ablation experiments to verify our speculation.

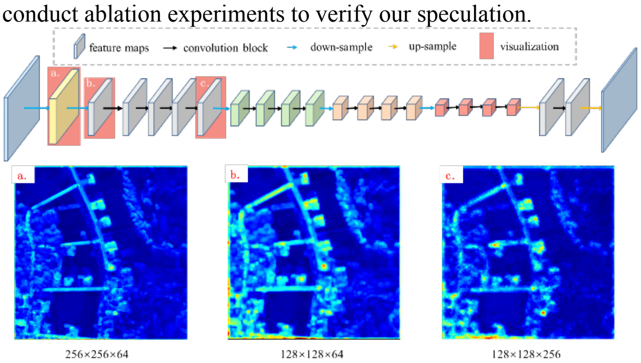

Figure 6. Similar visualizations of feature maps (*a, b, c*) in *ResNet*, the input image is the same as Fig. 2.

3.3 Attention Generation

Similar to the attention calculation of Tao et al. (2020), we use two 3×3 convolutions without changing the number of channels to aggregate local information, and then use 1×1 convolution to compress the result to one channel. As shown in Fig. 7, the sigmoid activation function is used to range the value of attention between [0, 1], which acts as a weight tensor in the end.

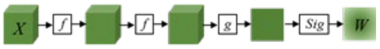

Figure 7. The workflow of our attention generation, i.e., the *PSA* module, where X, f, g, Sig, W denote the feature map, 3×3 convolution, 1×1 convolution, the *Sigmoid* activation function and the attention.

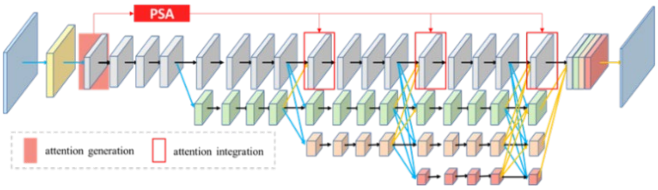

Figure 8. The workflow of *PSA* module applied in *HRNet*

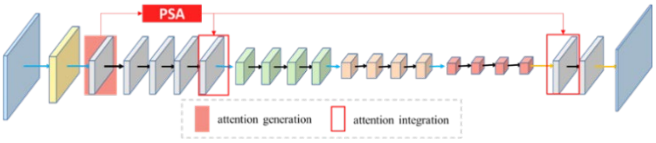

Figure 9. The workflow of *PSA* module applied in *ResNet*

3.4 Attention Integration

As information exchanging between different branches of *HRNet* leads to incomplete detailed information on the branch one, we apply our attention with skip connections after each exchange to refine details that belong to the branch one (Fig. 8).

$$attention = \varphi(x^{\circ}) = Sigmoid(x^{\circ} \otimes f \otimes f \otimes g) \quad (1)$$

$$x' = \omega(x, x^{\circ}) = x + x \cdot attention = x + x \cdot \varphi(x^{\circ}) \quad (2)$$

where x°, $x^{'}$, x represent the feature map for attention generation, the output feature map and the input feature map respectively, while f, g, $Sigmoid$, φ, ω denote 3×3 convolution, 1×1 convolution, the *Sigmoid* activation function, the attention generation and the attention integration ⊗ and · denote the convolution operation and the element-wise multiplication. As a result, the number of parameters are related to x° caused by the three convolutions, the corresponding computation relates to x° and $x^{'}$. Note that *UNet* applies skip connection for delivering details and *ResNet* uses it for delivering gradient, both of them connect two feature maps, our strategy of skip connection actually delivers weights by connecting the attention and corresponding feature maps. As a consequence, our attention is not supposed to restore the lost details caused by the pooling operation because it carries no feature information such as *UNet*. Instead, our method can in principle make an improvement in refining details lost by convolutions. With skip connection, our attention is able to weight feature maps with severe lost details instead of being calculated to weight its own feature map (similar to $x = x^{\circ}$) for incorporating context information (such as *Non-local*, *CBAM*, etc). In addition, different from channel-based foreground-aware attention, our method is a spatial attention, we are able to weight spatial details while foreground-aware attention weights features that are beneficial to the representation of the foreground. While Transformer (Vaswani et al., 2017; Zheng et al., 2020c) applies $query$, key, $value$ for attention generation and attention integration, we use convolutions to directly calculate weights and weight feature maps x (similar to $value$ in attention integration) by element-wise multiplication.

3.5 Loss Function

We use the conventional Categorical Cross-Entropy Loss to measure the distance between learning results and labels.

$$Loss = -\frac{1}{N}\sum_{p \in I}^{N}\sum_{i=1}^{C} y_i \log(p_i) \quad (3)$$

where $p = [p_1, \cdots, p_C]$ is a probability distribution of each pixel, p_i represents the probability that the pixel belongs to class i, $y = [y_1, \cdots, y_C]$ is the onehot label of corresponding pixel, when the pixel belongs to class i, $y_i = 1$, otherwise $y_i = 0$, C is the number of classes, I represents the input image and N is the number of pixels.

From the introduced interpretation, we can apply the proposed *PSA* module in different backbones (e.g., *ResNet*, *HRNet*) as they have similar architectures and feature maps, relevant experiments are given in next section 4.

4. EXPERIMENTS

The proposed method was tested on two remote sensing image datasets, including a satellite imagery dataset from 2021 "Shengteng Cup" Remote Sensing Image Intelligent Processing Algorithm Competition Fine-grained Semantic Segmentation Track (RSIPAC), and a relatively large-scale aerial image dataset, i.e., ISPRS Vaihingen 2D (Wegner et al., 2017). In order to comprehensively verify the weighting capacity and generalizability of our attention model, extensive studies were conducted for different backbones (i.e., *HRNet*, *ResNet*, *Deeplab v3+* and *UNet*) on the RSIPAC preliminary dataset. In the following subsections, we will give more details in regard to the dataset, experimental settings and our experimental results.

4.1 Dataset and implementation details

4.1.1 RSIPAC Dataset: The RSIPAC preliminary dataset is a satellite imagery benchmark for pixel-level remote sensing feature classification competition[1], which contains 35000 RGB images collected from satellites of China, with spatial resolution of 0.8 to 2 meters. Each image is with 512 × 512 pixels and 8 high quality annotated semantic classes and background. We randomly divide this dataset into two parts: 34000 images for training, 1000 images for both validation and test as we don't apply validation set for adjusting training procession. The corresponding 9 land cover classes in this dataset are *Background (B. g.), Plantation (Plan.), Forest and Grass (F&G.), Building (build.), Highway and Railway (H&R.), Structure (Str.), Artificial Surface of Accumulation and Excavation (Art. Surf.), Desert and Bare Soil (Soil.), Waters*.

4.1.2 ISPRS Vaihingen Challenge Dataset: This is an ISPRS 2D semantic labeling challenge benchmark dataset, including 33 very high-resolution true orthophoto (TOP) images (GSD ~ 9 cm) with size of 2500 × 2000 pixels. In addition, NIR, Red, Green bands, and two sets of auxiliary data, namely, the Digital Surface Model (DSM) and Normalized Digital Surface Model data (NDSM) are also available. This dataset was officially split into 16 areas for training and 17 areas for testing. In our experiments, we crop training areas into images with size of 512 × 512 pixels and obtained 4326 training images and 111 validation images through augmentation methods such as rotation, flip, etc. We also use the validation set to evaluate the test accuracy, and only TOP images are used in our experiments. The 6 land cover classes in this dataset are *Impervious Surface (Imp. Surf.), Building (Build.), Low Vegetation (Low Veg.), Tree, Car* and *Clutter* (Wegner et al., 2017).

4.1.3 Experimental settings: Our method is implemented with the Pytorch framework. The base learning rate is set to 0.001. A poly learning rate policy is employed, in which the initial learning rate is multiplied by $\left(1 - \frac{epoch}{total\_epoch}\right)^{1.0}$ during each epoch. All models in the experiments are trained with the SGD optimizer on NVIDIA GTX 3090ti GPUs. The momentum value is 0.9 and the weight decay value is 1e-4. For each experiment, the training procedure is with 100 epochs and validation is applied every 5 epochs. During each training epoch, we use all training images in the ISPRS Vaihingen Dataset and randomly take half for training in the RSIPAC Dataset, and total iteration numbers is determined by the batch size which is chosen according to each method, more training information is introduced in the following subsections. And in all experiments, we use corresponding pre-trained models on ImageNet dataset (Russakovsky et al. 2015).

4.1.4 Evaluation metrics: The performance of models on different datasets is assessed by intersection over union (IOU), precision, recall, F1-score, and overall accuracy (OA). The evaluation was based on an accumulated confusion matrix, from which IOU, precision, recall, F1-score, and overall accuracy can be derived:

$$IOU_k = \frac{TP_k}{TP_k + FP_k + FN_k} \quad (5)$$

$$Precision_k = \frac{TP_k}{TP_k + FP_k}; Recall_k = \frac{TP_k}{TP_k + FN_k} \quad (6)$$

$$F1score_k = 2 \cdot \frac{Precision_k \cdot Recall_k}{Precision_k + Recall_k} \quad (7)$$

$$OverAccuracy = \frac{\sum_{k=1}^{N} TP_k}{\sum_{k=1}^{N} TP_k + FP_k + TN_k + FN_k} \quad (8)$$

where TP_k, FP_k, TN_k, FN_k denote the true positive, false positive, true negative and false negative pixels, respectively, k is the class index. We use mean intersection over union (mIOU) and average F1-score (avg. F1) to represent mean results for all classes.

4.2 Ablation study for attention generation

All experiments in this section are conducted on the RSIPAC dataset. The contribution of the feature maps for attention

[1] More details related to RSIPAC can be found at: http://rsipac.whu.edu.cn/subject_one

Networks	F1 [%]									mIOU [%]	avg.F1 [%]	OA
	B.g.	*Plan.*	*F&G.*	*Build.*	*H&R.*	*Str.*	*Art. Surf.*	*Soil.*	*Waters*			
HRNet	67.76	87.95	93.26	81.18	62.48	**56.13**	55.52	**80.41**	90.02	61.93	74.97	89.55
PSA+HRNet	**70.67**	**87.99**	**93.38**	**81.22**	**62.55**	55.36	**62.86**	80.03	**90.08**	**63.05**	**76.02**	**89.67**
ResNet152	72.82	87.68	93.30	**80.18**	55.22	**59.60**	**63.57**	81.12	90.89	63.16	76.04	89.65
PSA+ResNet152	**75.70**	**88.05**	**93.54**	80.13	**58.43**	58.35	62.47	**83.44**	**91.06**	**64.15**	**76.80**	**90.02**
UNet	**74.68**	**87.92**	**93.44**	80.90	**61.99**	**55.17**	56.97	81.40	89.76	62.95	**75.80**	**89.81**
PSA+UNet	73.76	87.71	93.43	**80.95**	61.51	51.59	**59.61**	**83.48**	**89.99**	**63.03**	75.78	89.78
Deeplab v3+	**77.51**	88.08	**93.44**	**81.43**	**61.63**	58.52	60.18	**81.12**	**90.97**	64.33	76.99	**89.93**
PSA+Deeplab v3+	76.07	**88.37**	93.33	81.12	61.47	**59.44**	**65.06**	80.31	90.78	**64.60**	**77.33**	89.91

Table 2. Results of land cover classification for ablation experiments on the RSIPAC dataset. Best scores are in bold font.

generation is demonstrated by the final accuracy, this is to confirm our speculation in section 4.2, and a, b, c (same as in Fig. 5) is used to test corresponding experiments. The training batch size is set to 8 for *HRNet* (2126 iterations per epoch) and other settings follow the description in section 4.1.2. The results are listed in Tab. 1.

In Tab. 1, we can find that the numerical results are basically consistent with the mentioned qualitative discussion of the feature maps (in Fig. 5 and 6), and we visualize our attention as heatmaps in Fig. 10, the heatmap *b* is visually the best match as human interpretation. Quantitatively, the *HRNet+PSA(b)* outperforms the other two by 0.52% and 0.85% in mIOU, 0.51% and 0.98% in avg.F1, respectively. Compared to the baseline (*HRNet*), the mIOU and F1-score are improved by 1.12% and 1.05% when using the *HRNet+PSA(b)*, and the other two (using *PSA* with *a*, *c*) also have few improvements, this shows the feasibility of the proposed attention. In general, the feature map *b* (Fig. 5) is obviously the best choice for our *PSA* module with both relatively clear boundaries and rich sematic information, and it is advocated to generate attention as our method in the following experiments.

Networks	mIOU [%]	avg.F1 [%]	OA [%]
HRNet	61.93	74.97	89.55
HRNet+PSA(a)	62.53	75.51	89.53
HRNet+PSA(b)	**63.05**	**76.02**	89.67
HRNet+PSA(c)	62.20	75.04	**89.79**

Table 1. Results of ablation experiments for the *PSA* generation on the RSIPAC dataset. Best scores are in bold font.

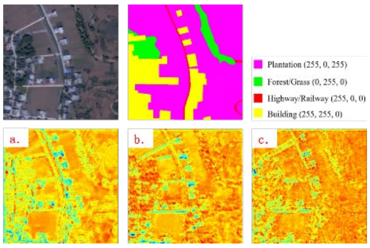

Figure 10. Visualization of the attentions generated from feature maps *a*, *b*, *c* (the same as in Tab. 1) in the bottom.

4.3 General study for the *PSA* module on various networks

This general experiment on the RSIPAC dataset is to validate the applicability of the *PSA* module into different backbones, i.e., *HRNet* (Sun et al., 2019), *ResNet152* (He et al., 2016), *UNet* (Ronneberger et al.,2015), and *Deeplab v3+* (Chen et al., 2018a). Apart from the evaluation metrics in section 4.1.4, the amount of calculations, parameters and inference time are also explored in this section. The training batch size is set to 8 for *HRNet* and *Deeplab v3+* (2126 iterations per epoch), 12 for others (1417 iterations per epoch), and other settings follow section 4.1.2. The results of experiments are listed in Tab. 2, and the cost of each method is reported in Tab. 4. Result visualizations is shown in Fig. 11, and Tab. 3 provides the label colors for each land cover class.

In Tab. 2, the results show that the *PSA* module improve 0.27%-1.12% in mIOU, 0.34%-1.05% in F1-score on *HRNet*, *ResNet152* and *Deeplab v3+*, especially the first two models, achieving about 1% improvement in both mIOU and F1-score. However, our method can hardly improve the performance based on *UNet*, and barely improves on *Deeplab v3+*. This could be due to the fact that skip connections are already applied in both *UNet* and *Deeplab v3+*, which leads to the channel concatenation of feature maps from encoder and decoder, the incomplete semantic information from encoder presumably is considered to stem the *PSA* module from refining details in decoder. In addition, skip connection in the *PSA* module works between the same feature maps as in *UNet*, which means skip connections reused on the same architecture probably result in negative influence, even though they provide different functions. In general, the *PSA* module can make improvement over most backbones in our experiments, but the results of *H&R.*, *Str.* and *Art. Surf.* remains poor, the possible reason is the relatively small coverage of these categories. Furthermore, the spectral response of *Str.* is close to *Build.*, and the inter-class discrimination between *Art. Surf.* and *Soil.* is not clearly defined.

Class	*B. g.*	*Plan.*	*F&G.*
Color	⬛ (255, 255, 0)	🟪 (255, 0, 255)	🟩 (0, 255, 0)
Class	*Build.*	*H&R.*	*Str.*
Color	🟨 (255, 255, 0)	🟥 (255, 0, 0)	🟥 (255, 124, 128)
Class	*Art. Surf.*	*Soil.*	*Waters*
Color	⬜ (165, 165, 165)	🟫 (102, 51, 0)	🟦 (0, 0, 255)

Table 3. The land cover classes and corresponding colors on labels of RSIPAC dataset

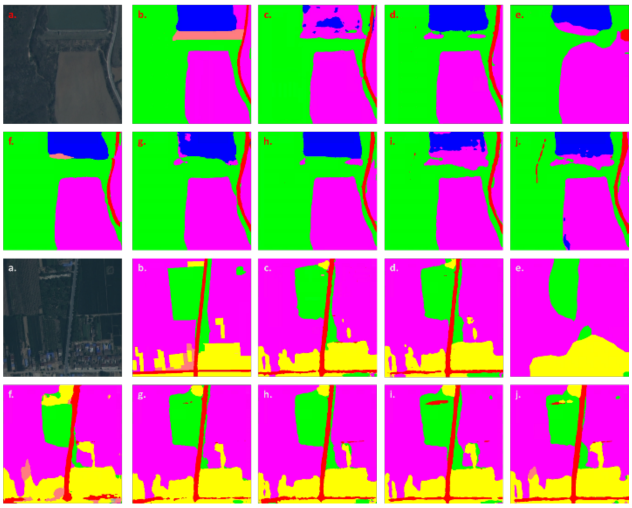

Figure 11. Examples for classification results of general study on the RSIPAC dataset, images marked with *a* are the input images, and with *b*, *c*, *d*, *e*, *f*, *g*, *h*, *i*, *j* denote the ground truth labels and corresponding results of *HRNet, PSA + HRNet, ResNet152, PSA + ResNet152, UNet, PSA + UNet, Deeplab v3+, PSA + Deeplab v3+*. Corresponding colors for land cover classes are listed in Tab. 3.

In Fig. 11, it can be seen that the *PSA* module improve the

https://doi.org/10.5194/isprs-annals-V-3-2022-281-2022 | © Author(s) 2022. CC BY 4.0 License.

prediction of *Waters* and *H&R.*, where the lost details in original models result in incomplete classification. Especially, the improvement is pretty clear when our method is applied in *HRNet* and *ResNet152*. The prediction of *Build.* is also more consistent with the ground truth when the *PSA* module is applied. However, the classification of *Str.* remains a problem as the reason is due to the imbalance of training samples caused by the small coverage of *Str.*, and the distribution of *Str.* is similar to *Build.* in the feature space, which leads to the ambiguity when classification is conducted between these two categories.

In Tab. 4, the Float-Point Operations Per Second (FLOPS) which measures the computing power of a computer, and the GFLOPS means 1 billion floating-point operations per second are provided. The Params., Time in Tab. 4 denote parameters in models and the time for a single image (batch size set to one) to be predicted on the GPU (NVIDIA GTX 3090ti GPUs). These constitute the cost of all models, smaller is better.

Networks	GFLOPS[G]	Params.[M]	Time[ms]
HRNet	93.63	65.85	76.17
PSA+HRNet	94.86	65.93	76.30
ResNet152	61.74	63.46	32.54
PSA+ResNet152	148.62	63.53	36.03
UNet	215.29	104.89	24.94
PSA+UNet	216.51	104.97	27.80
Deeplab v3+	262.48	59.35	41.89
PSA+ Deeplab v3+	274.58	60.09	44.51

Table 4. The cost of all models.

From Tab. 4, it can be seen that the *PSA* module barely increases the cost of parameters as we only apply three convolution kernels for the attention generation, and other parts have no contribution to the additional parameters. The GFLOPS is relevant to the number of channels in the feature maps where we integrate our attention, and the corresponding number is 2048 in *ResNet152*, leading to a surge in the cost of computation when the *PSA* module is applied in *ResNet152*. The increment in the inference time caused by the *PSA* module is also acceptable.

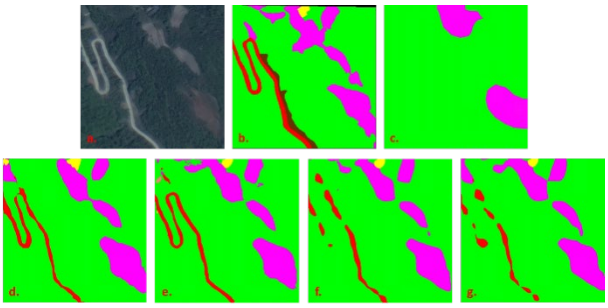

Figure 12. Example for classification results on the RSIPAC dataset, the image marked with *a* is the input image, and *b*, *c*, *d*, *e*, *f*, *g*, denote the ground truth label, result of the baseline (*ResNet152*), and corresponding results applied with *PSA, LKPP, SENet, Non-local*, respectively. Corresponding colors for land cover classes are listed in Tab. 3.

4.4 Comparison to the state-of-art

Several state-of-art methods are compared on the RSIPAC dataset and the ISPRS Vaihingen dataset. We use *ResNet152* (He et al., 2016) as the baseline, and apply *LKPP* (Zheng et al., 2020a), *SENet* (Hu et al., 2020), *Non-local* (Wang et al., 2017) and the proposed *PSA* on it, the cost of computation and parameters are also explored in this section. In addition, the up-sampling strategy of the baseline applied with *LKPP* follows the *EaNet* (Zheng et al., 2020a), which reduces the number of channels and up-sampling layer-by-layer together with convolutions, while others remain the same linear interpolation up-sampling as the baseline. The training batch size is set to 12 for each method (1417 iterations per epoch), and other settings follow the section 4.1.2. The results of experiments are evaluated in Tab. 7, and the cost of each method is reported in Tab. 6. Visualization of one example in the results on the two datasets is shown in Fig. 12 and Fig. 13, respectively, corresponding colors for each class on the RSIPAC dataset is same as section 4.3, and colors on the ISPRS Vaihingen dataset are listed in Tab. 5.

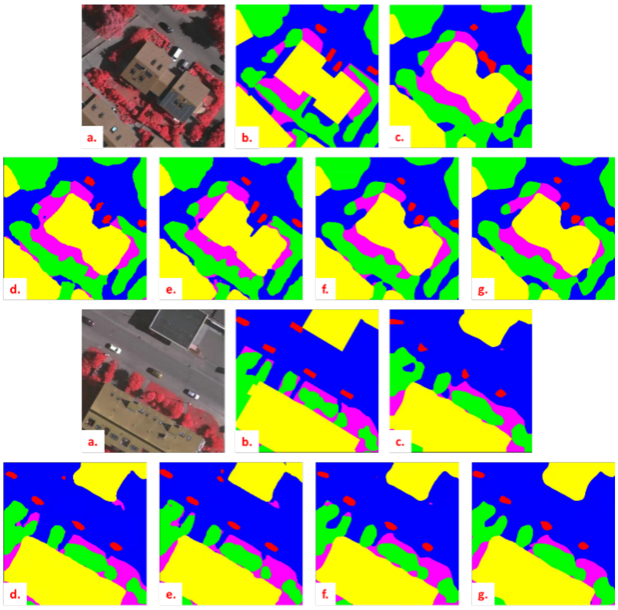

Figure 13. Examples for classification results on the ISPRS Vaihingen dataset, images marked with *a* are the input images, and *b*, *c*, *d*, *e*, *f*, *g*, denote the ground truth labels, results of the baseline (*ResNet152*), and corresponding results applied with *PSA, LKPP, SENet, Non-local*, respectively. Corresponding colors for land cover classes are listed in Tab. 5.

Class	*Imp. Surf.*	*Build.*	*Low Veg.*
Color	🟦 (0, 0, 255)	🟨 (255, 255, 0)	🟪 (255, 0, 255)
Class	*Tree*	*Car*	*Clutter*
Color	🟩 (0, 255, 0)	🟥 (255, 0, 0)	⬛ (0, 0, 0)

Table 5. The land cover classes and the corresponding colors on labels of the ISPRS Vaihingen dataset.

From Tab. 7, it is clear that the *PSA* module achieves the highest prediction scores in terms of the mIOU, F1-score and OA metrics on the RSIPAC dataset while the other methods can hardly improve the performance of the baseline. Compared to the baseline on the ISPRS Vaihingen dataset, the *PSA* module improve 4.8% in mIOU, 4.07 % in F1-score and 1.62% in OA. However, the performance is clearly better when *LKPP* is applied into the baseline, this is due to its different strategy for up-sampling, which also explains the corresponding GFLOPS doesn't grow as the same as the cost of parameters does (Tab. 6). In contrast to *LKPP*, for all the other methods including ours, the GFLOPS is doubled while the cost of parameters barely increases. We want to remind that the *PSA* module still has the smallest increment regarding inference time among all methods in Tab. 6.

Networks	GFLOPS[G]	Params.[M]	Time[ms]
Baseline	61.74	63.46	32.54
PSA	148.62	63.53	36.03
LKPP	83.30	116.16	50.74
SENet	147.41	63.98	36.28
Non-local	148.61	68.18	36.53

Table 6. The cost of computation and parameters compared to the state-of-art methods. The baseline is *ResNet152*.

Dataset	Methods	F1 [%]									mIOU [%]	avg.F1 [%]	OA
		B.g.	*Plan.*	*F&G.*	*Build.*	*H&R.*	*Str.*	*Art. Surf.*	*Soil.*	*Waters*			
RSIPAC	*Baseline*	72.82	87.68	93.30	80.18	55.22	**59.60**	**63.57**	81.12	90.89	63.16	76.04	89.65
	PSA	75.70	**88.05**	**93.54**	80.13	58.43	58.35	62.47	83.44	**91.06**	**64.15**	**76.80**	**90.02**
	LKPP	74.41	87.87	93.37	**80.38**	**58.64**	54.89	60.71	82.45	90.18	63.06	75.88	89.70
	SENet	76.25	87.94	93.49	79.65	52.92	55.63	62.87	**84.19**	89.57	63.24	75.95	89.83
	Non-local	**77.23**	87.55	93.22	79.42	51.69	55.64	60.96	81.64	89.35	62.32	75.19	89.49
		Imp. Surf.		*Build.*		*Low Veg.*		*Tree*	*Car*				
ISPRS Vaihingen	*Baseline*	85.80		92.06		73.30		82.93	53.70		65.16	77.56	83.52
	PSA	87.44		92.78		74.76		**84.93**	68.26		69.90	81.63	85.14
	LKPP	**88.57**		**93.57**		**76.06**		84.86	**76.97**		**73.63**	**84.01**	**85.94**
	SENet	87.39		92.97		75.19		84.48	68.48		70.67	81.70	85.10
	Non-local	86.66		92.31		74.02		83.85	62.90		70.98	79.95	84.32

Table 7. Results of land cover classification using *PSA* and the state-of-art methods on the RSIPAC dataset and the ISPRS Vaihingen dataset. The baseline is *ResNet152*. Best scores are in bold font.

An example for classification results on the RSIPAC dataset is given in Fig. 12, it can be seen that all the four methods improve the prediction of *H&R.* and *Build.*, while the baseline fails to identify them due to the loss of detail. In particular, the performance is inspiring when the *PSA* module or the *LKPP* module is applied in the baseline. It is puzzling that the visualization results of methods with *LKPP, SENet, Non-local* outperform the results of the baseline, whereas quantitative the evaluations barely show the identical improvement in Tab. 7. We consider that the low coverage of land cover classes (i.e., *H&R.*, *Str.*, *Art. Surf.*) pose difficulty in classification, and the baseline tend to ignore classification errors in these categories in order to maintain the accuracy of others (e.g., *Plan.*, *F&G.*). In addition, due to the similarity of spectral response to *H&R.* and *Soil.*, all the methods fail to identify *Soil.* in Fig. 12. As for the visualization results given in Fig. 13, it is clearly shown that the *PSA* module is helpful for refining the details, and the *LKPP* module actually performs better, especially on *Car*.

5. CONCLUSION

In this paper, we have proposed a position-sensitive attention (*PSA*) to refine the lost details caused by stacking convolutions in FCN models for land cover classification. We investigated the performance of the *PSA* module and applied it in different backbones, the experimental results show that our method is able to improve mIOU by 0.27%-1.12% and average F1-score by 0.34%-1.05% on the RSIPAC dataset with just very small additional increasing in parameters and computations. Compared to the existing methods, i.e., *LKPP* (Zheng et al., 2020a), *SENet* (Hu et al., 2020), *Non-local* (Wang et al., 2017), the *PSA* module works well on the RSIPAC dataset while the others barely improve the baseline (*ResNet*, He et al., 2016), and the additional parameters and inference time of our *PSA* is the lowest as well. Furthermore, after integrating *ResNet152* with *PSA*, the performance on the aerial imagery dataset (i.e., the ISPRS Vaihingen Dataset) is also promising, specifically, the improvement is 4.8% in mIOU, 4.07% in average F1-score and 1.62% in OA. Although it is unexpectedly mundane when comparing to other methods applied in *ResNet152*, this probably resulted from the fact that these two datasets are with different resolution, which needs further detailed investigation.

In general, our method of the *PSA* module performs well on the RSIPAC dataset and the ISPRS Vaihingen Dataset, but there are still several issues to be further considered: First, only visual effect is studied to select the feature maps for the attention generation, which is a subjective choice and can be affected by various factors, it will be helpful to explore a feasible criterion for the feature map selection. Second, we would like to explore the manually experimental settings (e.g., hyper-parameters) for better performance and extensively refer to other works such as *EaNet* (Zheng et al., 2020a) in the future.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 61871295).

REFERENCES

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12): 2481-2495.

Camps-Valls, G., Tuia, D., Bruzzone, L., Atli Benediktsson, J., 2014. Advances in Hyperspectral Image Classification: Earth Monitoring with Statistical Learning Methods. IEEE Signal Processing Magazine, 31(1):45-54.

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, L., 2018a. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40 (4):834-848.

Chen, L., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018b. Encoder-decoder with atrous separable convolution for semantic image segmentation. European Conference on Computer Vision, pp. 801-818.

Dai, Y., Gieseke, F., Oehmcke, S., Wu, Y., Barnard, K., 2021. Attentional Feature Fusion. IEEE Winter Conference on Applications of Computer Vision, arXiv preprint arXiv: 2009.14082.

Demir, I., et al., 2018. DeepGlobe 2018: A challenge to parse the Earth through satellite images. IEEE Conference on Computer Vision and Pattern Recognition Workshops, arXiv preprint arXiv: 1805.06561.

Deng, X., Zhu, Y., Tian, Y., Newsam, S., 2020. Scale Aware Adaptation for Land-Cover Classification in Remote Sensing Imagery. arXiv preprint arXiv: 2012.04222.

Fu, J., Liu, J., Tian, H., Fang, Z., Lu, H., Li, Y., Bao, Y., 2019. Dual attention network for scene segmentation. IEEE Conference on Computer Vision and Pattern Recognition, pp. 3141-3149.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. IEEE conference on computer vision and pattern recognition, pp. 770-778.

Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E., 2020. Squeeze-and-Excitation Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(8):2011-2023.

Kussul, N., Lavreniuk, M., Skakun S., Shelestov, A., 2017. Deep Learning Classification of Land Cover and Crop Types Using

Remote Sensing Data. IEEE Geoscience and Remote Sensing Letters, 14(5):778-782.

Lin, T. -Y., Dollár, P., Girshick, R., et al., 2017. Feature pyramid networks for object detection. IEEE Conference on Computer Vision and Pattern Recognition, pp: 2117-2125.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431-3440.

Mao, X., Zhao, Y., Chen, B., Ma, Y., 2020. Deep Learning with Skip Connection Attention for Choroid Layer Segmentation in OCT Images. Annual International Conference of the IEEE Engineering in Medicine & Biology Society, pp. 1641-1645.

Myint, SW., Lam, NSN., Tyler, JM., 2004. Wavelets for urban spatial feature discrimination: Comparison with Fractal, Spatial autocorrelation and spatial co-occurrence approaches. Photogrammetric Engineering and Remote Sensing, 70(7): 803-812.

Noh, H., Hong, S., Han, B., 2015. Learning deconvolution network for semantic segmentation. IEEE International Conference on Computer Vision, pp. 1520-1528.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional neworks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention, pp. 234-241.

Russakovsky, O., Su, H., Krause, J., Satheesh, S., et al., 2015. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3):211-252.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 1409.1556.

Sun, K., Zhao, Y., Jiang, B., Cheng, T., Xiao, B., Liu, D., et al., 2019. High-Resolution Representations for Labeling Pixels and Regions. arXiv preprint arXiv: 1904.04514.

Szegedy, C., Liu, W., Jia, Y., et al., 2015. Going deeper with convolutions. IEEE conference on computer vision and pattern recognition, pp. 1-9.

Tao, A., Sapra, K., Catanzaro, B., 2020. Hierarchical Multi-Scale Attention for Semantic Segmentation. arXiv preprint arXiv: 2005.10821.

Tarabalka, Y., Fauvel, M., Chanussot, J., Benediktsson JA., 2010. SVM- and MRF-Based Method for Accurate Classification of Hyperspectral Images. IEEE Geoscience and Remote Sensing Letters, 7(4): 736-740.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I., 2017. Attention Is All You Need. arXiv preprint arXiv: 1706.03762.

Volpi, M., 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.

Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., 2018. Understanding convolution for semantic segmentation. IEEE Winter Conference on Applications of Computer Vision, pp. 1451-1460.

Wang, X., Girshick, R., Gupta, A., He, K., 2017. Non-local Neural Networks. arXiv preprint arXiv: 1711.07971.

Wegner, J.D., Rottensteiner, F., Gerke, M., Sohn, G., 2017. The ISPRS labelling challenge. Available on: http://www2.isprs.org/commissions/comm3/wg4/semanticlabeling.html (accessed 20/01/2020).

Woo, S., Park, J., Lee, J-Y., Kweon I.S., 2018. CBAM: Convolutional Block Attention Module. European Conference on Computer Vision, Part VII, 11211:3-19.

Yang, C., Rottensteiner, F., Heipke, C., 2019. Towards better classification of land cover and land use based on convolutional neural networks. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XLII-2/W13, pp. 139-146.

Yang, C., Rottensteiner, F., Heipke, C., 2020. Investigations on Skip-Connections with an Additional Cosine Similarity Loss for Land Cover Classification. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 3, pp. 339-346.

Yu, F., Koltun, V., 2016. Multi-Scale Context Aggregation by Dilated Convolutions. International Conference of Legal Regulators on Computer Vision and Pattern Recognition, arXiv preprint arXiv: 1511.07122.

Zhan, Z., Zhang, X., Liu, Y., Sun, X., Pang, C., Zhao, C., 2020. Vegetation Land Use/Land Cover Extraction from High-Resolution Satellite Images Based on Adaptive Context Inference. IEEE Access, vol. 8, pp. 21036-21051.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid scene parsing network. IEEE Conference on Computer Vision and Pattern recognition, pp. 2881-2890.

Zheng, X., Huan, L., Xia, G-S., Gong, J., 2020a. Parsing very high resolution urban scene images by learning deep ConvNets with edge-aware loss. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 170, pp. 15-28.

Zheng, Z., Zhong, Y., Wang, J., Ma, A., 2020b. Foreground-Aware Relation Network for Geospatial Object Segmentation in High Spatial Resolution Remote Sensing Imagery. arXiv preprint arXiv: 2011.09766.

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., et al., 2020c. Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers. arXiv preprint arXiv: 2012.15840.

Zhu, X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., Fraundorfer, F. 2017. Deep learning in remote sensing: a comprehensive review and list of resources. IEEE Geoscience and Remote Sensing Magazine, 5(4):8-36.