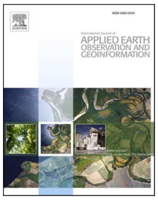
Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

journal homepage: www.elsevier.com/locate/jag



SCDNET: A novel convolutional network for semantic change detection in high resolution optical remote sensing imagery

Daifeng Peng [a,b,*], Lorenzo Bruzzone [b], Yongjun Zhang [c], Haiyan Guan [a], Pengfei He [a]

[a] School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China
[b] Department of Information Engineering and Computer Science, University of Trento, Trento 38123, Italy
[c] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China

ARTICLE INFO

Keywords:
Semantic change detection
Remote sensing
Siamese UNet
Multi-scale atrous convolution
Attention mechanism
Deep supervision

ABSTRACT

With the continuing improvement of remote-sensing (RS) sensors, it is crucial to monitor Earth surface changes at fine scale and in great detail. Thus, semantic change detection (SCD), which is capable of locating and identifying “from-to” change information simultaneously, is gaining growing attention in RS community. However, due to the limitation of large-scale SCD datasets, most existing SCD methods are focused on scene-level changes, where semantic change maps are generated with only coarse boundary or scarce category information. To address this issue, we propose a novel convolutional network for large-scale SCD (SCDNet). It is based on a Siamese UNet architecture, which consists of two encoders and two decoders with shared weights. First, multi-temporal images are given as input to the encoders to extract multi-scale deep representations. A multi-scale atrous convolution (MAC) unit is inserted at the end of the encoders to enlarge the receptive field as well as capturing multi-scale information. Then, difference feature maps are generated for each scale, which are combined with feature maps from the encoders to serve as inputs for the decoders. Attention mechanism and deep supervision strategy are further introduced to improve network performance. Finally, we utilize softmax layer to produce a semantic change map for each time image. Extensive experiments are carried out on two large-scale high-resolution SCD datasets, which demonstrates the effectiveness and superiority of the proposed method.

1. Introduction

Change detection (CD) is the process of detecting Earth surface changes by using geographically co-registered multi-temporal remote-sensing images (Bruzzone and Bovolo, 2012). In particular, with the increasing improvement of sensors, a large amount of RS imagery with diverse resolutions and modalities are available, which make it possible to observe and understand Earth surface at a finer scale. As a result, it is crucial to monitor land use/cover changes using CD technologies, which have achieved tremendous success in the areas of urban spreading monitoring, ecosystem assessment, resources management, and municipal planning (Gao and Liu, 2010; Doxani et al., 2012; Rokni et al., 2015; De Alwis Pitts and So, 2017; Li et al., 2018).

The CD procedure mainly consists of three steps, namely preprocessing, change analysis and change map generation. According to the type of semantic label information desired in the output change map, CD falls into two categories: binary change detection (BCD) and semantic change detection (SCD). In BCD, the change map distinguishes between changed and unchanged pixels by using a binary label. In contrast, both change extents and change types can be determined by SCD. Here, change types refers to land-cover transitions between bitemporal images, such as “from land to building”, “from forest to farmland”, etc. Consequently, “from-to” information can be well embedded through SCD, where problems related to both “where changes happen” and “how changes happen” are solved simultaneously. Therefore, compared with BCD, SCD is a more complex change detection task, where comprehensive change information can be obtained. An illustration of BCD and SCD is presented in Fig. 1. Notably, to facilitate deep convolutional network training, the proposed SCD results consist of two individual classification maps, with different colors denoting the unchanged and changed classes for each period of image. One can also combine the two classification maps into a single map, where each color represents a land-cover transition type. However, in that case, the label space will inevitably increase sharply. It should also be noted that the

[*] Corresponding author at: School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China.

E-mail address: daifeng@nuist.edu.cn (D. Peng).

https://doi.org/10.1016/j.jag.2021.102465
Received 28 April 2021; Received in revised form 16 July 2021; Accepted 23 July 2021
Available online 13 August 2021

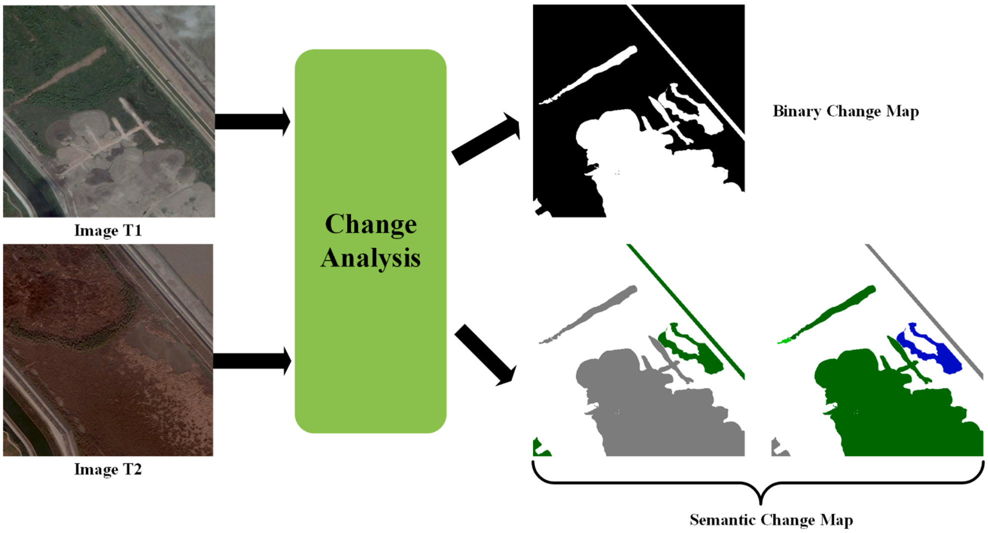


Fig. 1. Illustration of BCD and SCD tasks. The different colors in the semantic change maps denote different change classes in each period of image.

proposed SCD is different from traditional land use/cover transitions, where explicit land use/cover category is needed to be assigned to each pixel or patch without considering the unchanged class (Ru et al., 2020). Nevertheless, in our case, large proportions of pixels are labeled as the unchanged class, which is crucial for reducing the annotation work for large-scale RS datasets.

In the literature, the majority of the research work is focused on BCD due to its simplicity and lower requirements on input data. In general, BCD techniques evolve with the development of the RS sensors and benefit a lot from the ideas in computer vision. In the early stage, pixel-based BCD methods dominate the CD field, where pixels can be treated independently in low-resolution RS imagery (Bruzzone and Prieto, 2000). Later on, very high-resolution (VHR) RS images become available, making it possible for fine-grained observation of ground objects. In such context, object-based BCD methods are developed based on image objects generated by image segmentation techniques, which are more close to human perception. Change maps can thus be generated by comparing image object features (Leichtle et al., 2017) or class memberships (Volpi et al., 2013). Note that, on VHR images, object-based BCD methods far outperform pixel-based counterparts due to the usage of spatial context information and rich object features. Recently, with the development of computing resources and the availability of large-scale RS imagery, deep-learning based CD methods, especially convolutional neural network (CNN), are thriving (Shi et al., 2020). Benefiting from the powerful feature representation ability of CNN, change maps can be easily produced by comparing deep features of bi-temporal images (Hou et al., 2017). Furthermore, BCD can be seen as a binary segmentation problem, thus it is natural to learn the change maps directly from the input image pairs through an elaborated fully convolutional network (FCN) (Peng et al., 2019). In such case, the degree of automation and intelligence can be greatly improved. Note that to improve CD perforamnce, diverse attention mechanisms and Siamese architectures were proposed to generate discriminative features and obtain accurate change maps (Chen and Shi, 2020; Liu et al., 2020). However, large amounts of dense label maps are required to train FCNs with millions of parameters, which is labour intensive and costly to acquire for RS images. Therefore, weakly-supervised and semi-supervised techniques are further introduced to make it possible to train the FCNs with limited training data (Peng et al., 2020). In addition, instance-level sample augmentation techniques were also proposed to overcome the rarity and sparsity of the changed samples (Chen et al., 2021).

Despite tremendous success has been achieved in BCD, there exist large gaps for comprehensive change recognition and understanding due to the absence of semantic information. With the improvement of scene-level interpretation, many attempts have been made to introduce scene-level semantic label into CD (Bovolo and Bruzzone, 2015). Based on post-classification comparison strategy, a scene-level CD framework is proposed for VHR imagery (Wu et al., 2016). However, the considered handcrafted features are sensitive to scenes, and the classification error accumulation is prone to happen. To address these drawbacks, CNN was introduced to extract deep feature representations, where deep canonical correlation analysis (DCCA) was also coupled to capture potential correlation of the unchanged scene patches (Ru et al., 2019). However, obvious limitations exist for scene-level CD: 1) it is difficult to define a proper patch size for different RS scenes; 2) as the scene patch only denotes a rectangle area of the target object, it fails to delineate the object boundary, which is essential for accurate change information post-processing and statistical analysis, such as changed object vectorization and changed areas calculation. Therefore, it is crucial to implement SCD with pixel-level label information.

To overcome the above-mentioned limitations, a novel semantic CD network (SCDNet) is proposed, where pixel-level semantic change maps can be generated. Change transition status is therefore easily presented by the bi-temporal semantic change maps with “from-to” class labels. SCDNet, which is designed based on an encoder-decoder architecture, consists of two encoders and decoders, making it possible to generate semantic change maps by combining bi-temporal image information effectively. The contributions of this article can be summarized into two aspects:

- We propose a novel SCDNet for dealing with pixel-level SCD task, which is flexible and easy to implement in an end-to-end manner. To fully exploit the semantic change information, a Siamese UNet architecture with shared weights is adopted to implement multi-level feature representations and fusions effectively for bi-temporal images.
- To capture multi-scale changes, multi-scale atrous convolution units are employed in the encoders. For the benefit of improving feature fusion and avoiding gradient vanishing, attention mechanism and deep supervision strategy are further introduced in the decoding stage. To address the class imbalance issues, we define a novel class-wise loss function by combining the advantages of dice loss and focal loss. Code will be available at https://github.com/daifeng2016/Semantic-Change-Detection.

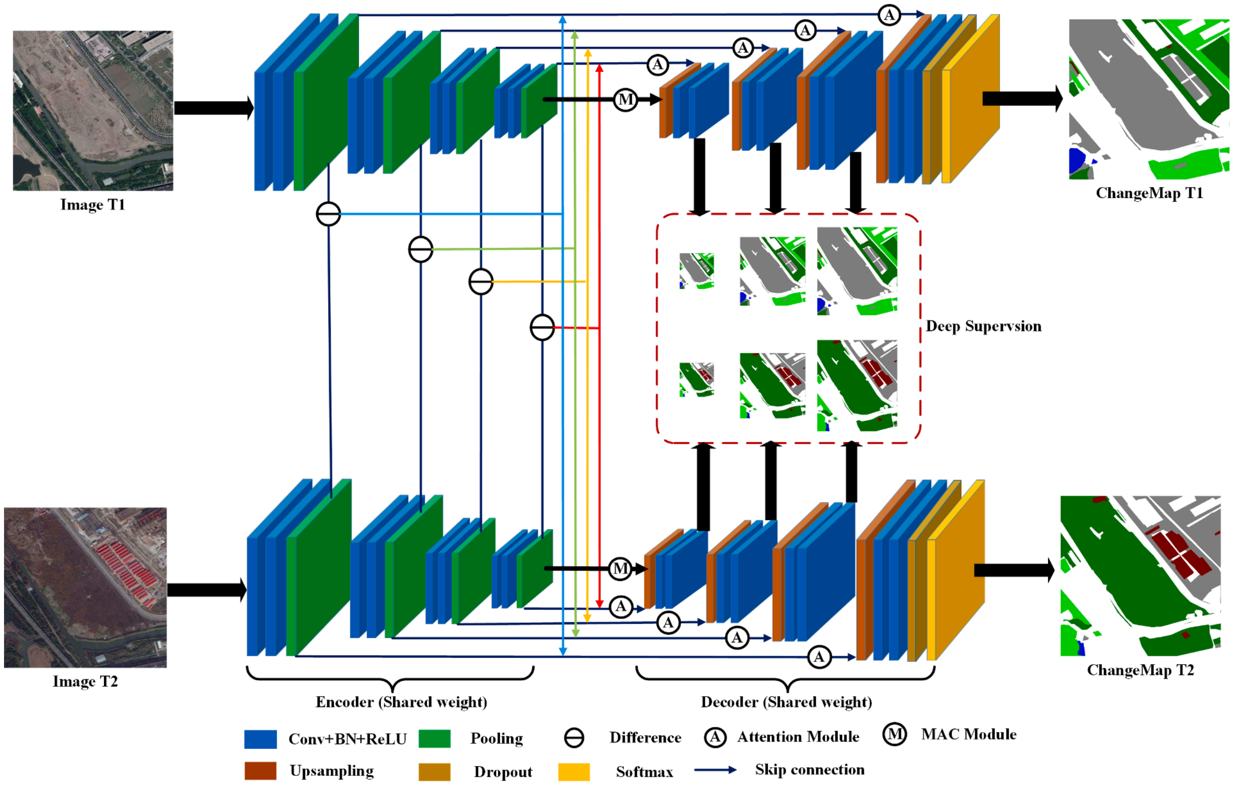


Fig. 2. Flowchart of the proposed SCDNet.

2. Related work

Based on the difference of semantic label interpretation unit, SCD can be dividied into two categories: scene-level semantic change detection (SLSCD) and pixel-level semantic change detection (PLSCD). In SLSCD, semantic label is assigned to individual scene unit, such as object instances in street-view scene or object patches in RS scene. Instead, semantic label is given to each pixel in PLSCD, where fine-grained semantic change map can be therefore generated.

2.1. Scene-level semantic change detection

In the computer vision field, identifying scene changes is the necessary step towards high-level scene understanding tasks such as autonomous driving, traffic control and infrastructure monitoring. In such context, street-view images or videos produced by drones or mobile mapping systems are used for scene-level change analysis, where the main focus is on newly added or reduced objects of the interest. Kataoka et al. (2016) decoupled SCD into two separate tasks of semantic segmentation and change detection. To obtain high-level performance, hypermaps and multi-scale feature representations are used for image patches. Alcantarilla et al. (2018) proposed CDNet for detecting structural changes using street-view videos, which consists of four stacking contraction and expansion blocks. In a similar work, Guo et al. (2018) proposed CosimNet for scene change detection, where a thresholded contrastive loss was used to learn more discriminative metrics. For the benefit of effective feature fusion, Lei et al. (2020) proposed HPCFNet to fuse features at multiple levels. However, only binary masks of the specified objects such as cars and traffic signs are generated. In order to locate and identify changes between image pairs simultaneously, Varghese et al. (2018) proposed ChangeNet, where multi-level outputs were combined to capture multi-level detail information of the objects. Nevertheless, the "from-to" issue, namely how the change happens, is still not solved. The main challenge can be attributed to the lack of the dataset with semantic label. To overcome the shortage of street-view SCD dataset, Sakurada et al. (2020) proposed a two-step SCD scheme by detecting change mask and estimating pixel-wise semantic labels separately. However, the method is too complex and can only be applied on the street-view dataset.

In RS domain, SLSCD is developed with the idea of single image scene classification, where image scene patches can be assigned a land use/cover label. With multi-temporal images available, it is natural to monitor land use/cover variation at the semantic level. An intuitive solution is to adopt post-classification comparison of scene patch, but it easily leads to classification errors accumulation effect. In addition, temporal correlation between image pairs is neglected. To overcome these drawbacks, Wu et al. (2017) introduced kernel KSFA to extract nonlinear temporally invariant features, followed by post-classification fusion to identify "from-to" types. However, Bag of Visual Words (BoVW) is utilized to serve as handcrafted features, which are not robust for large-scale dataset. In addition, the whole scheme cannot be jointly optimized. To overcome such limitations, Wang et al. (2019) adopted a CNN to extract spectral-spatial features, DCCA is further embedded to enhance the temporal correlations of multi-temporal images. However, due to the optimization problems of DCCA on a minibatch, the proposed method shows no superiority against BoVW-based methods. Based on deep features from pre-trained CNN and temporal correlation calculation by soft DCCA, Ru et al. (2020) further improved the feature representation ability of bi-temporal images by a correlation based feature fusion (CoffFusion) module, which achieves a remarkable performance gain on the proposed large-scale scene change detection dataset. However, patch size issues exist as ground objects change sharply in scale. In addition, although "from-to" transition type can be identified, the scene-level semantic CD results only consists of rough object patches without exact boundaries, making it impossible for accurate change analysis such as contour vectorization and area calculation of the changed regions.

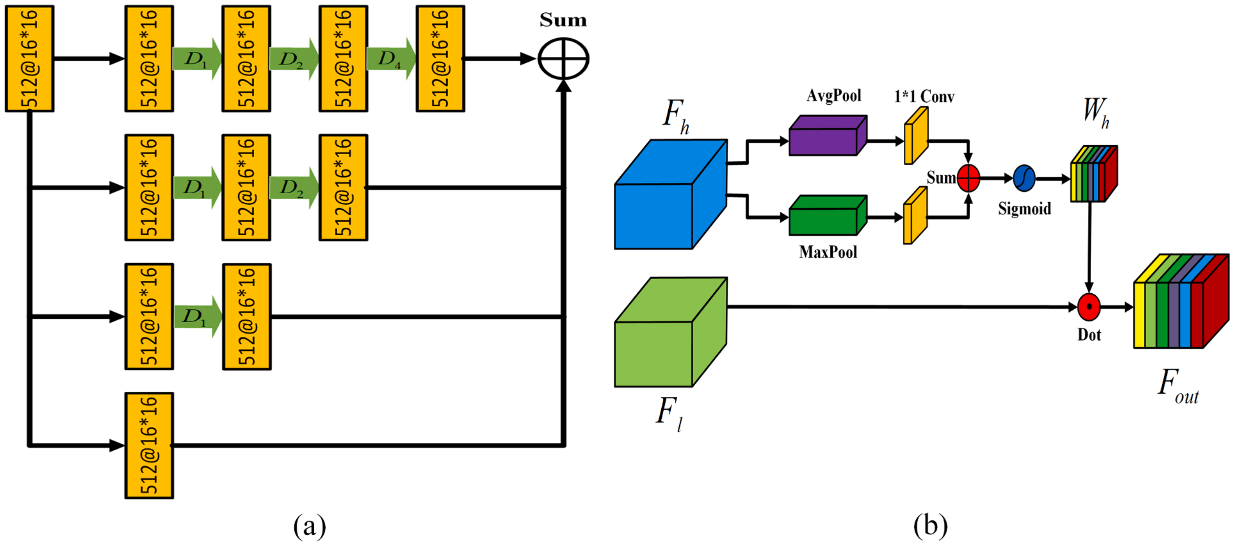


Fig. 3. Illustration of the multi-scale atrous convolution (MAC) unit and attention unit. (a) MAC unit. (b) attention unit.

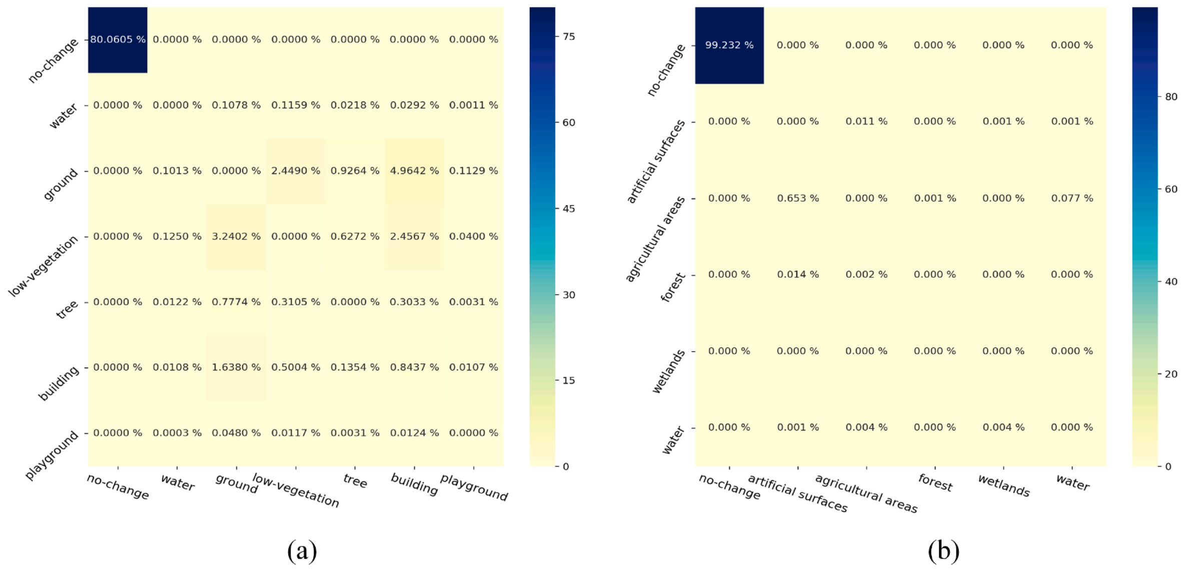


Fig. 4. Illustration of label distribution of two datasets. (a) Sensetime dataset. (b) HRSCD dataset.

2.2. *Pixel-level semantic change detection*

Pixel-level semantic annotations for input image pairs is required in pixel-level semantic change detection. Compared with pixel-level annotations in single image semantic segmentation, the annotations in PLSCD are much easier, as only the changed regions are needed to be annotated while the rest are labeled as the unchanged. However, due to the scarcity of such dataset, few work has been investigated to solve PLSCD issues. Recently, Daudt et al. (2019) proposed a multitask learning for large-scale SCD, where a large-scale high-resolution semantic change detection (HRSCD) dataset for RS community was first proposed. Notably, four strategies of SCD were compared and analysed systematically. However, the change type is only determined without “from-to” information between bi-temporal images. It is noteworthy that Cheng et al. (2020) proposed a SCD dataset using aerial images, named as SCPA-Wuhan City (SCPA-WC). Tian et al. (2020) proposed a large-scale SCD dasetset named Hi-UCD, which consists of tri-temporal images and their corresponding land cover change maps of nine object types. However, both the SCPA-WC and Hi-UCD datasets have not been open to public yet. To address the issues of categorical ambiguity of different changed classes, an asymmetric Siamese network for semantic change detection was proposed (Yang et al., 2020). A novel SCD dataset named SECOND was further presented and evaluated. More recently, an artificial intelligence remote sensing interpretation competition was held by the famous Sensetime company, where a large-scale pixel-level semantic change detection dataset was provided for SCD task.[1] The challenging dataset greatly promotes the research of PLSCD as well as motivating us to explore the deep learning methods to solve PLSCD task.

[1] https://rs.sensetime.com/competition/index.html.

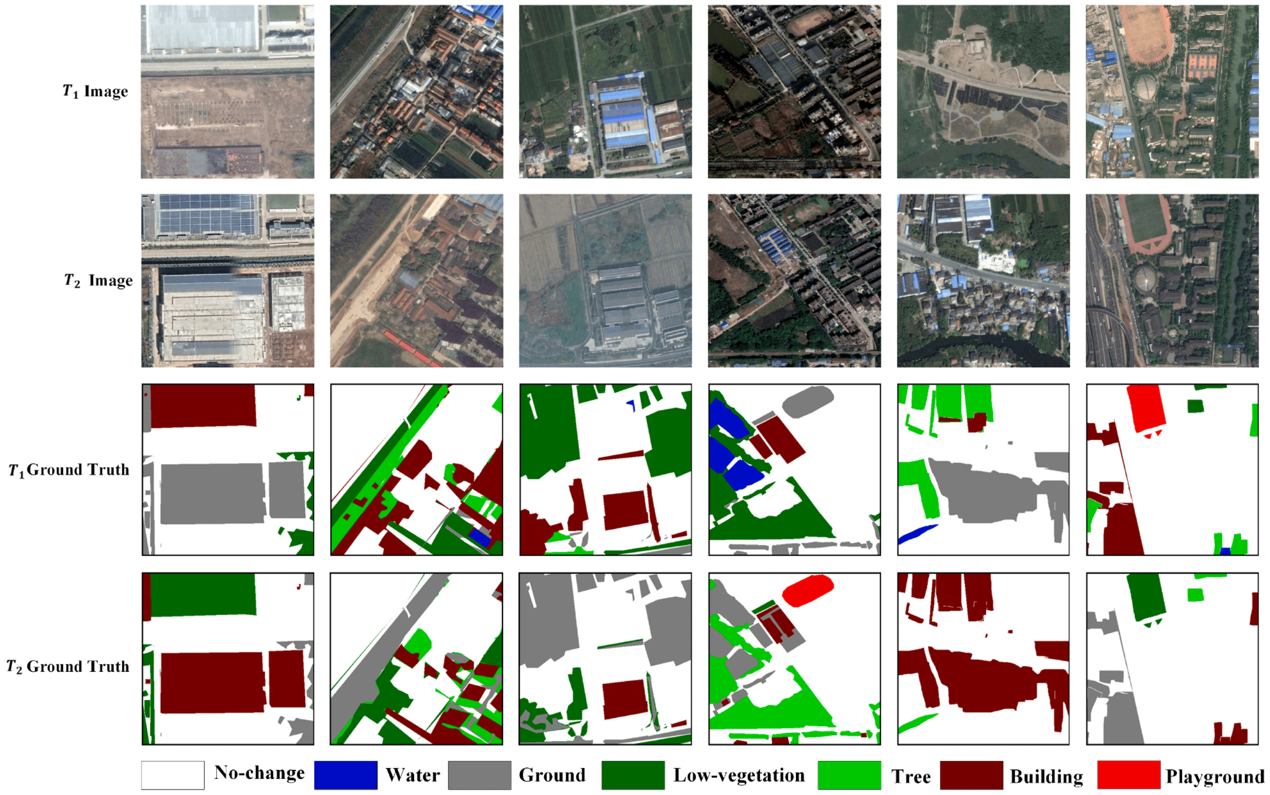


Fig. 5. Example images of Sensetime dataset.

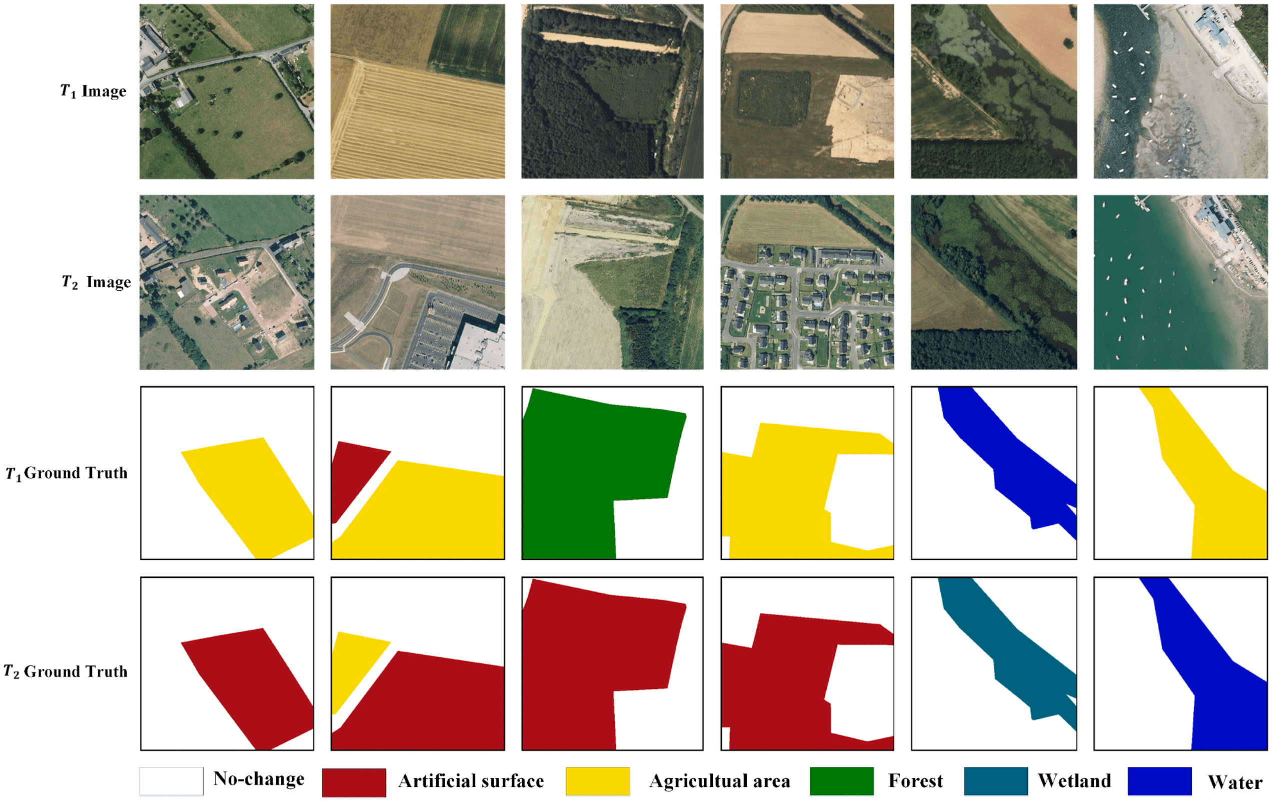


Fig. 6. Example images of HRSCD dataset.

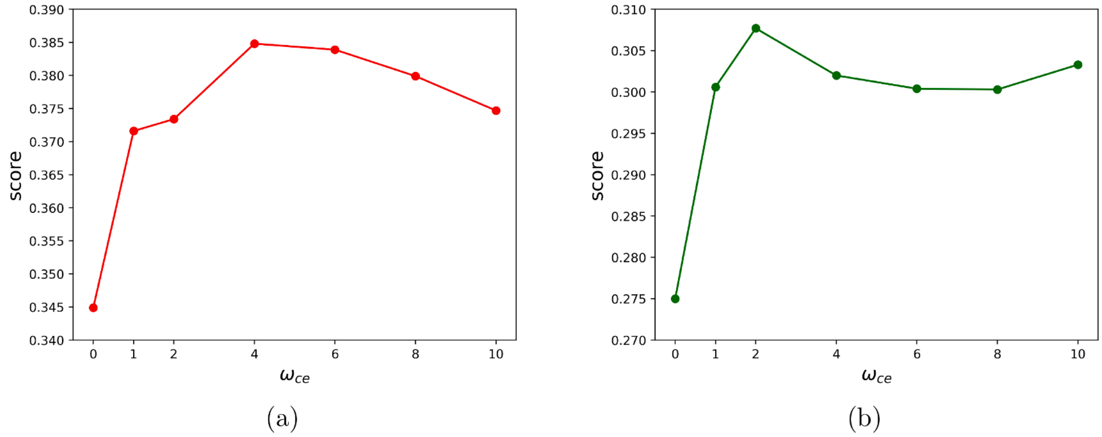


Fig. 7. Effects of parameter ω_{ce} on the quantitative performance of the proposed method. (a) Sensetime dataset. (b) HRSCD dataset.

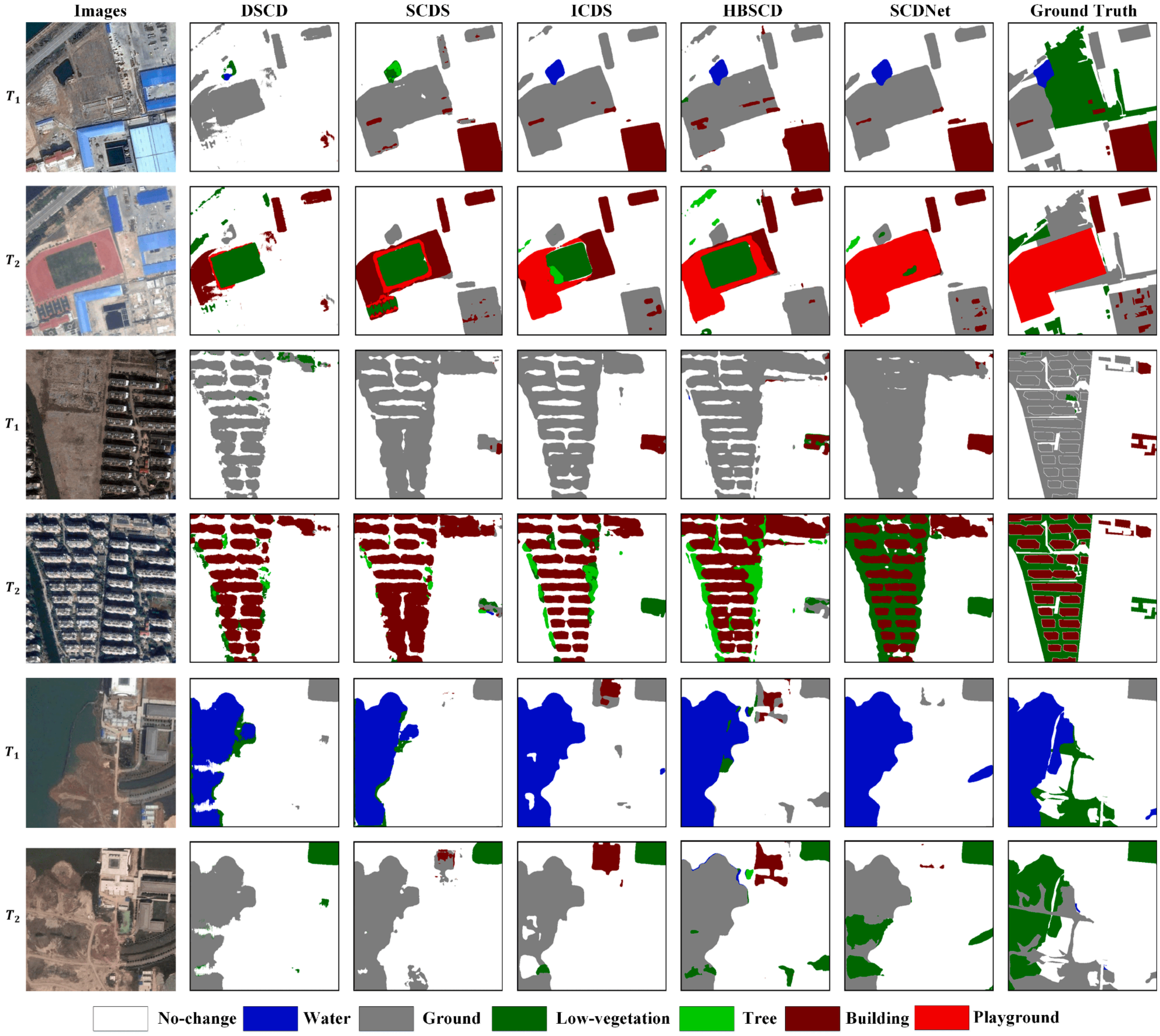


Fig. 8. Visual comparisons of the change maps obtained by different methods on the first three image pairs for Sensetime dataset.

3. Proposed SCDNet

3.1. Problem definition

Assume two periods of input images are X_1 and X_2 , which contains L object labels, namely $class(X_1)\subset\{0,1,\ldots,L-1\}$ and $class(X_2)\subset\{0,\ 1,\ \ldots L-1\}$. The output change maps are denoted as Y_1 and Y_2. Note that due to the inclusion of unchanged class, the possible label space in Y_1 and Y_2 is $L+1$, i.e., $class(Y_1)\subset\{0,1,\ldots,L\}$ and $class(Y_2)\subset\{0,1,\ldots,L\}$. Then the process of SCD is to find a mapping function f, so that:

$$f(X_1,X_2)=(Y_1,Y_2) \tag{1}$$

In particular, the two semantic change maps can be combined into a single map Y, namely $Y = C(Y_1,\ Y_2)$, where $C(.)$ is the combination operation. The label in Y encodes the unchanged and changed transition types between X_1 and X_2, thus the possible label space in Y is L^2-L+1, namely $class(Y)\subset\{0,1,\ldots,L^2-L\}$

3.2. SCDNet architecture

Fig. 2 presents the SCDNet architecture, which is made up of dual-branch encoders and decoders with shared weights. To accelerate network convergence, pre-trained ResNet34 is adopted as the backbone for the encoder. Difference feature maps of each scale are generated for change analysis. At the end of the encoder, a multi-scale atrous convolution (MAC) module is inserted to enlarge the receptive field. Then, the contracted feature maps are expanded by a series of upsampling and convolution operations to recover a full-resolution feature map. To embed change information for each period of image, image feature maps and difference feature maps from different levels are combined in skip-connections. Furthermore, to fuse feature maps between encoders and decoders effectively, an attention module is utilized to re-calibrate the features. At the end of the decoder, a dropout layer is inserted to improve the network generalization ability. Finally, semantic change maps are produced by using softmax layers. Note that to improve the convergence of deep networks and overcome the vanishing gradient problems, deep supervision strategy (Lee et al., 2015) was further adopted to facilitate the network training, where three auxiliary classifiers are employed to generate multi-scale intermediate semantic change maps, as shown in the dotted box area of Fig. 2.

3.3. Multi-scale atrous convolution unit

After a series of pooling operations in the encoder, the resulted feature maps are much smaller than the input images, in our case they are $\frac{1}{32}$ of the input size. To enlarge the receptive field (RF) and capture multi-scale information, atrous convolution is usually introduced. For example, in the classic Atrous Spatial Pyramid Pooling (ASPP) module, multiple atrous convolution units are employed in a parallel manner and fused by concatenation operation (Chen et al., 2017). Instead, three successive atrous units are cascaded and fused by sum operation in our MAC unit, which is capable of capturing multi-scale information at the cost of lower computational burden, as shown in Fig. 3(a). Note that 512@16*16 denotes the feature map has 512 channels and a spatial size of 16 × 16.

Let $D_i(i=1,2,4)$ denote atrous convolution layers with different dilation rates i, F_e denotes the output feature maps of the encoder. Then the output of the MAC can be expressed as:

$$F_m = F_e + F_e\otimes D_1 + F_e\otimes D_1\otimes D_2 + F_e\otimes D_1\otimes D_2\otimes D_4 \tag{2}$$

where $\otimes$ denotes the convolution operation. Note that when the convolution kernel size is 3, the RF after different convolution layers of D_1, D_2, D_4 is 3, 5, 9, respectively. However, when the convolution layers are cascaded, $D_1\otimes D_2$ results in a RF of 7, and $D_1\otimes D_2\otimes D_4$ leads to a RF of 15, which covers almost the full size of the feature maps F_e. As a result, multi-scale features with different RFs can be generated effectively through MAC operation, which is essential for capturing image objects with different scales.

3.4. Attention unit

Low-level feature maps in the encoder contain rich detailed information but few semantic cues, whereas their high-level counterparts in the decoder have more semantic cues but less detailed information. Therefore, it is natural to enhance the feature representation ability by combining the two kinds of feature maps through a skip connection operation. However, due to the semantic discrepancy, simple feature maps combination through concatenation or sum operation will easily lead to feature confusion. In addition, it is computationally complex and memory-consuming to generate discriminative features using self-attention mechanisms (Fu et al., 2019; Chen and Shi, 2020). In contrast, a novel light-weighted attention module is proposed, as shown in Fig. 3(b). An attention map W_h is generated by using high-level feature map F_h , then the low-level feature map F_l is re-calibrated, thus bridging the semantic gap for effective feature fusions and representations.

Let us assume $F_h\in\mathbb{R}^{C\times H\times W}, F_l\in\mathbb{R}^{C\times H\times W}$, where C, H and W denote the channel number, height and width of the feature maps, respectively. First, we use average pooling and max-pooling operations to capture different channel-wise attention clues. Then the channel correlations and weight distributions are learned by a 1 × 1 convolution layer. After combining the two outputs of 1 × 1 convolution layers, an attention map $W_h\in\mathbb{R}^{C\times1\times1}$ can be generated by applying a sigmoid layer. Finally, the re-weighted feature maps F_{out} can be generated by an element-wise multiplication between F_l and W_h:

$$W_h = \sigma[Conv(AvgPool(F_h)) + Conv(MaxPool(F_h))] \tag{3}$$

$$F_{out} = W_h\odot F_l \tag{4}$$

where σ and $\odot$ refers to the sigmoid operation and element-wise multiplication operation, respectively. Note that there exist two kinds of low-level feature maps, namely original feature maps and difference feature maps. Let F_{l1} and F_{l2} denote the original feature maps generated by the encoders, the difference feature maps are calculated as:

$$F_{ld} = abs(F_{l1} - F_{l2}) \tag{5}$$

where $abs(.)$ denotes the absolute difference operation.

3.5. Loss functions

For the proposed SCDNet, four output segmentation maps $y_i^p(1\leqslant i\leqslant 4)$ are generated for each period of image due to the usage of a deep supervision strategy. Thus, the single-temporal loss function $\mathcal{L}$ can be defined as a combination of the four side-output losses:

$$\mathcal{L} = \sum_{i=1}^{4}\mathcal{L}_{side}^{i}(y_i^p, y_i^t) \tag{6}$$

where $y_i^t(1\leqslant i\leqslant 4)$ denote the targets outputs corresponding to y_i^p, which are generated by bi-linear down-sampling. Let us assume y_4^t is the final output at full resolution, BD_s is a down-sampling operation at scale s. Then we can get $y_1^t = BD_{16}(y_4^t), y_2^t = BD_8(y_4^t), y_3^t = BD_4(y_4^t)$. The side-output loss consists of two parts: global-level cross-entropy loss $\mathcal{L}_{ce}$ and class-level combination loss $\mathcal{L}_{df}$.

$$\mathcal{L}_{side} = \omega_{ce}\mathcal{L}_{ce} + \mathcal{L}_{df} \tag{7}$$

where ω_{ce} is a trade-off parameter. The cross-entropy loss is used to penalize the inconsistency between prediction outputs y^p and target outputs y^t globally. It can be defined as:

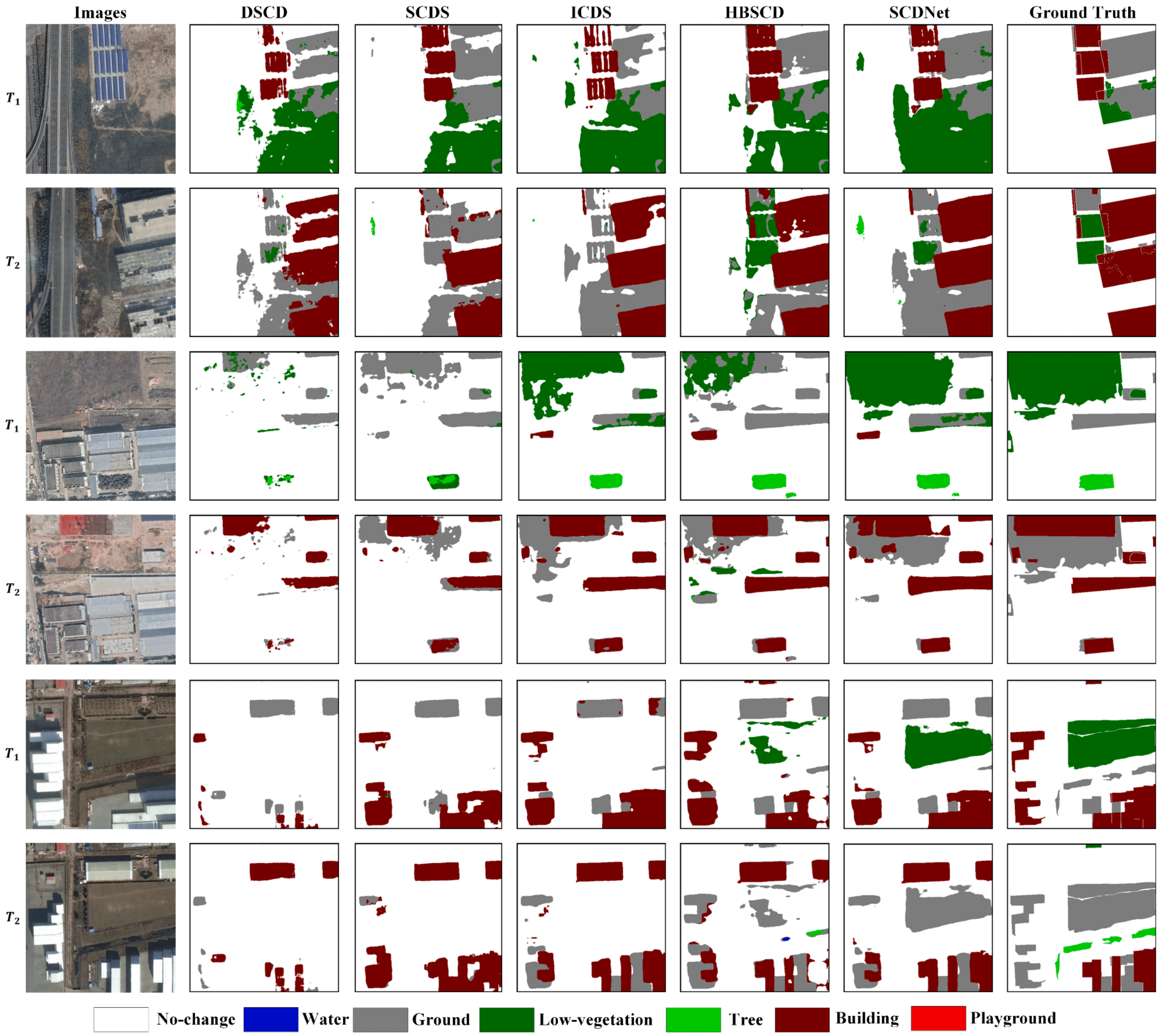


Fig. 9. Visual comparisons of the change maps obtained by different methods on the second three image pairs for Sensetime dataset.

Table 1
Quantitative results on the accuracy of different methods.

Method	Senetime dataset			HRSCD dataset		
	mIoU	Sek	Score	mIoU	Sek	Score
DSCD	0.6245	0.1020	0.2588	0.6415	0.0783	0.2473
SCDS	0.6918	0.1496	0.3122	0.6815	0.1028	0.2764
ICDS	0.7195	0.2183	0.3686	0.6853	0.1337	0.2992
HBSCD	0.7240	0.2146	0.3674	0.6238	0.0448	0.2185
Proposed SCDNet	**0.7306**	**0.2366**	**0.3848**	**0.6950**	**0.1418**	**0.3077**

$$\mathcal{L}_{ce} = -y^{t}\log(y^{p}) \tag{8}$$

To tackle the class imbalance issues, dice loss and focal loss are usually employed (Milletari et al., 2016; Lin et al., 2017). To be specific, the network is forced to pay more attention on the hard samples by using focal loss. However, focal loss is calculated pixel-by-pixel without considering spatial dependence, while dice loss is a region-aware loss with spatial constraints. To combine the advantages of two losses in a multi-class segmentation task more concisely, we use one-hot encoding strategy to generate binary ground truth for each category individually. Then, class-wise binary dice loss and focal loss can be combined effectively, and class weight can also be embedded flexibly.

$$\mathcal{L}_{df} = \sum_{j=1}^{C}\omega_j(\omega_{dice}\mathcal{L}_{dice} + \omega_{fl}\mathcal{L}_{fl}) \tag{9}$$

where C is the number of class categories, ω_j denotes the class weight for each class label $j \in C$, ω_{dice} and ω_{fl} denote the weights of dice coefficients loss $\mathcal{L}_{dice}$ and focal loss $\mathcal{L}_{fl}$, respectively. Given the one-hot encoded output y_p and its target output y_t, $\mathcal{L}_{dice}$ and $\mathcal{L}_{fl}$ can be calculated as:

$$\mathcal{L}_{dice} = 1 - \frac{2y_t y_p}{y_t + y_p} \tag{10}$$

$$p_t = (1 - y_t)(1 - y_p) + y_t y_p \tag{11}$$

$$\mathcal{L}_{fl} = - \ (1 - p_t)^{\gamma} \log(p_t) \tag{12}$$

where γ is used to adjust the influence of the easy samples, which is commonly set to 2 (Lin et al., 2017). Notably, here, y_t is one-hot encoded so as to include class-level information individually. In that case, $y_t, y_p \in \mathbb{R}^{C \times W \times H}$ can be regarded as a combination of targets and predictions of C binary output maps for each change category, respectively. In addition, assuming $r(j)(1 \leqslant j \leqslant C) \in [0, 1]$ denotes the class ratio of each change category, the class weight ω_j of each class j can be then defined as:

$$\omega_j = \begin{cases} f(r(j)) & if \ r(j) > 0 \\ 0 & otherwise \end{cases} \tag{13}$$

$$r'(j) = max\left(\frac{r(j)}{min(r(j))}\right) \Big/ \left(\frac{r(j)}{min(r(j))}\right) \tag{14}$$

$$f(r(j)) = \frac{r'(j)}{\sum(r'(j))} \tag{15}$$

Finally, the total loss function $\mathcal{L}_{total}$ can be defined by summing over the losses between each period of image:

$$\mathcal{L}_{total} = \mathcal{L}_{t1} + \mathcal{L}_{t2} \tag{16}$$

where $\mathcal{L}_{t1}$ and $\mathcal{L}_{t2}$ denote the loss function of images t_1 and t_2, respectively.

4. Experimental results and discussion

4.1. Datasets description

The effectiveness of the proposed SCDNet is verified on two VHR remote-sensing SCD datasets, namely Sensetime dataset and HRSCD dataset (Daudt et al., 2019).

Sensetime Dataset. This SCD dataset consists of 2968 training image pairs and 847 testing image pairs. For each pair of training data, both original images and their corresponding semantic change maps are provided, while only original images are available for the testing data. Thus, we split the training data into training set and testing set randomly using a ratio of 9:1. Fig. 5 presents the example images of the Sensetime dataset. To be specific, these image pairs have a size of 512 × 512 pixels, covering six types of land-cover classes, i.e. water, ground, low-vegetation, tree, building, and playground, which leads to 31 "from-to" change types in total. It should be noted that, the annotations of the dataset is highly imbalanced. Fig. 4(a) presents the label distribution of bi-temporal image pixels. As one can see, non-changed pixels accounts for more than 80% of the total pixels, while the 31 changed types only take up small proportions. In addition, many changed types are under 0.1%, which poses great challenges for the SCD method.

HRSCD Dataset. This dataset is made up of 291 RGB aerial image pairs of 10000 × 10000 pixels at a resolution of 50 cm per pixel, which are acquired in 2005/2006 and 2012 for each period of image, respectively. Both original images, binary change maps (BCM) and their land-cover maps (LCM) are provided. To make this dataset consistent with the Sensetime dataset, we generate semantic change maps based on the BCM and LCM. To facilitate GPU training, original large images are cropped into non-overlapping 512 × 512 image pairs, where the example images are presented in Fig. 6. The training set and testing set are also randomly split with the ratio of 9:1. Note that, the five land-cover classes only result in 11 "from-to" change types due to the missing of many change types in the annotations. Fig. 4(b) illustrates the label distribution of bi-temporal image pixels, where the non-changed types occupy more than 99.2% of the total pixels, while the changed types only take up less than 0.8%. That results in a serious class imbalance issue. In addition, compared with Sensetime dataset, the annotions in HRSCD dataset are coarser, many noisy labels such as inaccurate boundaries and false annotations exist, which brings even more challenges for SCD task.

4.2. Comparative methods and evaluation metrics

For comparative analysis, four SOTA SCD methods are compared and analyzed comprehensively:

1) Direct SCD (DSCD). In DSCD, each possible type of change is regarded as an independent semantic label, thus SCD can be treated as a simple semantic segmentation task (Daudt et al., 2019). In that case, "from-to" change is encoded in a single label, a semantic label map is therefore generated based on the semantic change maps from two periods of images.
2) Separate CD and segmentation (SCDS), where SCD is decoupled into two separate tasks, namely BCD and semantic segmentation (Daudt et al., 2019). Therefore, two different networks, BCD Network (BCDNet) and Semantic segmentaion Network (SSNet), are designed and trained separately to tackle each task. In our case, image pairs are fed into the BCDNet to generate binary change maps, which are then combined with each period of image to generate the related semantic segmentation maps.
3) Intergrated CD and segmentation (ICDS). In ICDS, a multitask network is designed to solve BCD and semantic segmentation simultaneously (Daudt et al., 2019), where three output maps are generated, i.e. one binary change map and two semantic segmentation maps.
4) HRNet based semantic change detection (HBSCD). This is the winner method of the Sensetime change detection competition, where HRNet40 is served as the backbone to extract deep features for bi-temporal images, then two semantic segmentation heads and one binary change detection head are used to generate the semantic change maps.[2]

Due to the high label imbalance issues, traditional evaluation metrics, such as overall accuracy (OA), fail to provide a reasonable accuracy metric. For example, as the no-change pixels account for more than 80% of the total pixels, the OA is always larger than 0.8 when all the pixels are classified as the no-change type. Therefore, two novel evaluation metrics are used, namely mean Intersection over Union (mIoU) and separate Kappa (Sek). The former is used to evaluate the SCD results from the perspective of BCD, while the latter is in view of SCD. To be specific, mIoU is calculated by averaging the IoUs between the non-changed and changed classes:

$$IoU_1 = \frac{TN}{TN + FP + FN} \tag{17}$$

$$IoU_2 = \frac{TP}{TP + FN + FP} \tag{18}$$

$$mIoU = 0.5^*(IoU_1 + IoU_2) \tag{19}$$

where IoU_1 and IoU_2 denote the IoU of non-changed type and changed type, respectively. TP, FP, TN and FN refer to the number of true positives, false positives, true negatives, and false negatives, respectively. They are defined using the BCD confusion matrix $B_{ij}(0 \leqslant i \leqslant 1, 0 \leqslant j \leqslant 1)$.

However, mIoU, which is dominated by the non-changed pixels, still suffers from the label imbalance issue. To address this drawback, Sek is defined by combining IOU_2 and a novel Kappa after removing true predictions of the non-changed class. Assuming the label category in the

[2] https://github.com/LiheYoung/SenseEarth2020-ChangeDetection.

semantic change map is C, where label '0' represents the non-changed class, the SCD confusion matrix is $S_{ij}(0 \leqslant i \leqslant C-1, 0 \leqslant j \leqslant C-1)$. To remove the true predictions of the non-changed, S_{00} is set to 0, SeK can be then calculated as:

$$Sek = Kappa * e^{IoU_2 - 1} \tag{20}$$

$$Kappa = \frac{p_0 - p_e}{1 - p_e} \tag{21}$$

$$p_0 = \frac{\sum_{i=0}^{C-1} S_{ii}}{\sum_{i=0}^{C-1} \sum_{j=0}^{C-1} S_{ij}} \tag{22}$$

$$p_e = \frac{\sum_{i=0}^{C-1} (S_{i+} * S_{+i})}{\left(\sum_{i=0}^{C-1} \sum_{j=0}^{C-1} S_{ij} \right)^2} \tag{23}$$

where S_{i+} denotes the row sum of the confusion matrix S_{ij}, while S_{+i} denotes the column sum. Based on mIoU and Sek, a comprehensive score can be calculated:

$$Score = 0.3 * mIoU + 0.7 * Sek \tag{24}$$

4.3. Training details

Based on Pytorch framework,[3] we implement SCDNet using a single 1080Ti GPU. Due to the fast convergence and weight regularization performance, AdamW optimizer is used, which combines the advantages of Adam and $L2$ regularization (Loshchilov and Hutter, 2017). The base learning rate (*base_lr*) is set to 1e-4. To stabilize network parameters learning, a learning rate warm-up strategy is adopted (He et al., 2016), where the learning rate is first increased from 0 to *base_lr* by a cosine method and then decreased by the cosine annealing scheme for each epoch. Note that the total epochs is set to 30, and warm-up epochs is set to 4. Due to the limitation of GPU, the batch size is set to 4 during the training stage. To overcome overfitting effect and improve network generalization ability, a scheduled dropblock strategy is adopted (Ghiasi et al., 2018) before the final softmax layer, where random window blocks with size of 7×7 in the feature maps are dropped. To further improve the performance, the drop ratio is linearly increased to the defined value of 0.1.

In training process, the training data are augmented online through randomly scaling, flipping and rotating by 90°, 180°, and 270°. In addition, 10% training data is chosen as the validation set to calculate the score metric for each epoch, so that the best model can be selected. The convolution kernel size is set to 3×3, and the number of convolution kernels in the encoder is set to {64, 128, 256, 512}. During the testing stage, test time augmentation (TTA) strategy is utilized to improve prediction performance, where the augmentation is carried out through rotation by $90^\circ, 180^\circ$, and 270°.

4.4. Results

4.4.1. Parameter setting

In the loss function of Eq. (7), ω_{ce} is used to balance the influence of global-level loss $\mathcal{L}_{ce}$ and class-level loss $\mathcal{L}_{df}$. To verify its sensitivity, ω_{ce} is varied from 0 to 10 for both datasets, and the score metric is calculated accordingly. As one can see in Fig. 7, when ω_{ce} is 0, only class-level combination loss is used, leading to lowest scores for both datasets. That means the global-level cross-entropy loss is indispensable for stable SCDNet training. Then, with the increase of the parameter ω_{ce}, the global-level cross-entropy is playing a more important role and the score value is also increased gradually. It reaches the peak when ω_{ce} is set to 4 and 2 for the Sensetime dataset and HRSCD dataset, respectively. However, by further increasing the value of ω_{ce}, the score value presents a decreasing trend on both datasets, which implies that higher global-level cross-entropy loss may reduce the network performance. Therefore, to achieve better performance, ω_{ce} is set to 4 and 2 for Sensetime dataset and HRSCD dataset, respectively. In addition, in Eq. (9) focal loss is assigned a higher weight to overcome sample imbalance issue, where ω_{dice} and ω_{fl} are experimentally set to 0.5 and 2.0, respectively.

4.4.2. Performance analysis

Sensetime Dataset. For the benefit of visual comparison, six typical areas are selected and presented in Fig. 8 and Fig. 9. One can observe that SCDNet achieves the best visual performance. In particular, compared with DSCD method, small-scale changes can be better detected by SCDNet, such as playground and low-vegetation. In addition, missed detections such as building and ground can also be largely reduced. Note that the unchanged class is consistent between the two corresponding semantic change maps for the DSCD and HBSCD methods, while there exist some discrepancy for the other considered methods, which can be seen as false detections. The reasons lie in the fact that only one decoder is used for generating change maps in DSCD, resulting in a single label for each pixel in the output map, which will be transformed into two individual semantic change maps while maintaining the label consistency. In HBSCD, the outputs of the two segmentation maps are constrained by the binary change map, thus generating consistent semantic change maps. On the contrary, for the other considered methods, two decoders are employed to generate two semantic change maps directly, which will inevitably lead to the inconsistency of the unchanged class due to random segmentation errors. Table 1 reports the quantitative evaluation results, we can observe that the proposed SCDNet achieves the highest mIoU, Sek, and Score values. It is noteworthy that DSCD method achieves the lowest values on the three metrics, which are 0.6425, 0.1020 and 0.2588, respectively. This maybe due to the fact that output space is increased sharply when "from-to" label is introduced. For example, the class number is increased from 7 to 32 for Sensetime dataset. That leads to even more serious class imbalance problems as well as making it difficult to train the network using the available training data. By decoupling the SCD into two simple sub-tasks of binary change detection (BCD) and semantic segmentation, the output space can be thus largely reduced, which reduces class imbalance issues and facilitates network training. As a result, compared with DSCD, SCDS achieves a gain of 6.73%, 4.76% and 5.34% for mIoU, Sek and Score, respectively. However, the two individual networks, namely BCD network and semantic segmentation network, have to be designed and trained sequentially, which inevitably leads to error accumulation effect and increases the training burden to a large extent. On the contrary, ICDS combines BCD and semantic segmentation into a unified framework, where two segmentation maps and one binary change map are generated simultaneously. Therefore, it is possible to optimize the two tasks jointly through end-to-end training, which overcomes the drawbacks of error accumulation. Consequently, compared with SCDS, ICDS achieves an improvement of 2.77%, 6.87% and 5.64% for mIoU, Sek and Score, respectively.However, it is hard to balance the influence between the BCD and semantic segmentation tasks, which are defined by different loss terms. Note that as two segmentation maps and one binary change map are also produced by HBSCD, it achieves similar quantitative performance with ICDS. Inevitably, the drawbacks of the balance issues between different tasks still exist. Rather than generating binary change maps explicitly, multi-scale difference feature maps are generated to serve as guidance to produce semantic change maps in our proposed SCDNet. This strategy brings two obvious advantages: 1) SCD can be regarded as a semantic segmentation

[3] https://pytorch.org/.

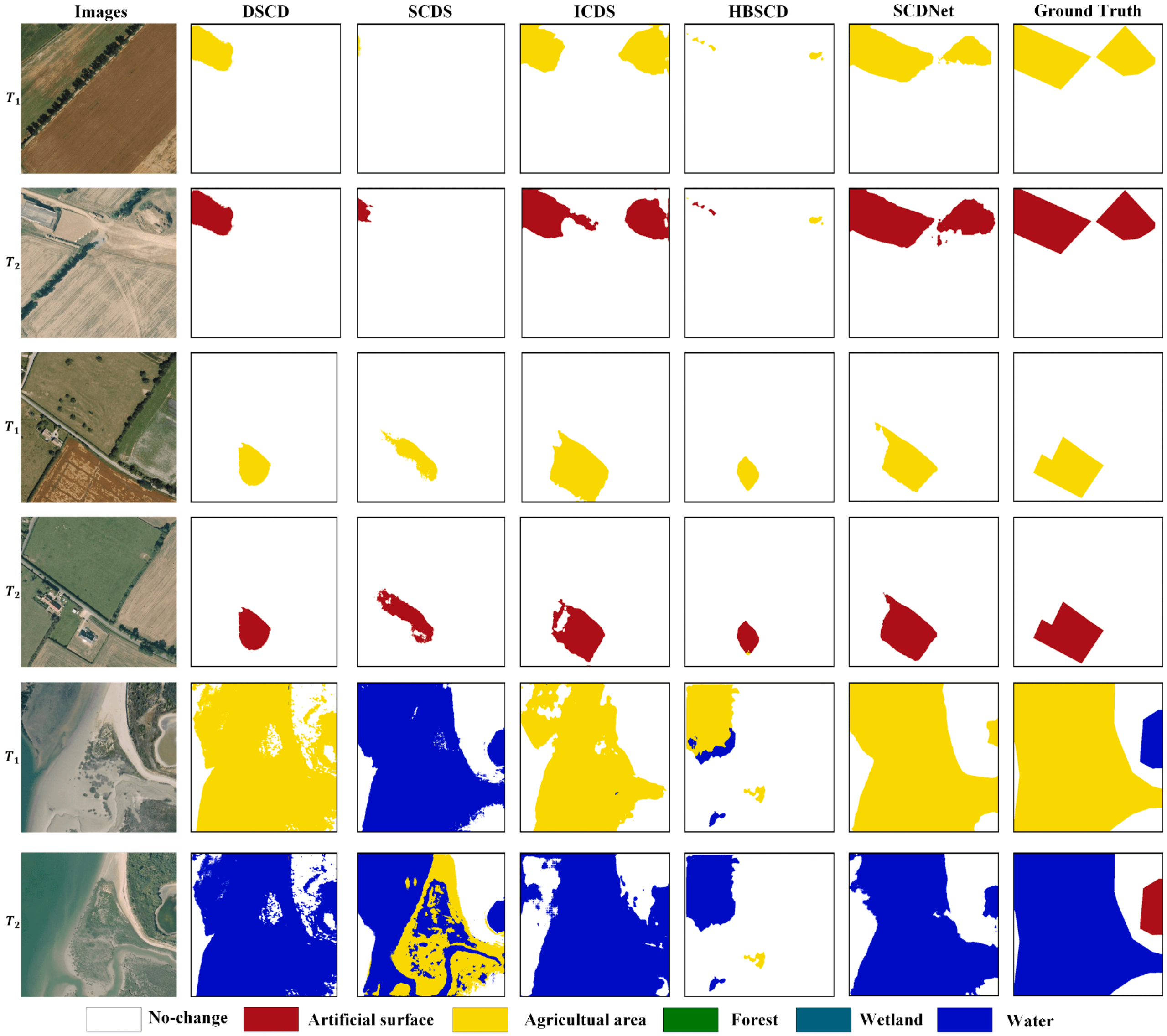


Fig. 10. Visual comparisons of the change maps obtained by different methods on the first three image pairs for HRSCD dataset.

task with two outputs, where multi-scale difference information can be used, 2) the network can be trained in an end-to-end way without the consideration of troublesome task balance issues between BCD and semantic segmentation. To capture multi-scale information and fuse multi-level features effectively, MAC unit and attention unit are employed. Furthermore, a deep-supervision strategy is utilized to suppress gradient vanishing and learn useful information from intermediate layers. Therefore, compared with ICDS method, the proposed SCDNet achieves better performance with a gain of 1.11%, 1.83% and 1.62% for mIoU, Sek and Score, respectively.

HRSCD Dataset. For a qualitative analysis, visual comparisons the change maps obtained by different methods on six typical areas are presented in Fig. 10 and 11. We can conclude that SCDNet achieves the best performance against other SOTA methods. Notably, for DCDS and SCDS, due to the large output space and error accumulation issues, noisy predictions are more easily generated, as shown in the last two rows in Fig. 10 and 11. Furthermore, some small-scale changes such as forest and wetland can not be detected for all the methods. This may due to the extreme class imbalance in the dataset. For example, no wetland class exist in the first period of images in HRSCD dataset, while the forest class only occupies 0.001% of the total pixels in the second period of images. In such case, it is almost impossible to detect such changes with the trained network.

Based on the quantitative results in Table 1, we can conclude that SCDNet still outperforms other comparative methods. Notably, one can observe that HBSCD method achieves the worst quantitative performance among the compared methods. This is due to the fact that only the changed pixels are used for optimizing the two segmentation heads in HBSCD, while the unchanged pixels are only used for optimizing the binary change detection head. In such case, the network cannot be converged with limited labeled data, leading to a sharp drop of the quantitative performance. In particular, only 0.8% changed pixels are annotated in the HRSCD dataset, which is far from the need of the supervised training for HBSCD. In addition, due to the large output space caused by "from-to" label, it is difficult to train DSCD using the available training data. Consequently, DSCD achieves a poor performance, with a mIoU of 0.6415, a Sek of 0.0783 and a Score of 0.2473. The decoupling of the SCD task into two sequential sub-tasks of BCD and semantic segmentation results in a better optimization of the corresponding networks using the available training data. As a result, SCDS outperforms DSCD by a large margin, with a mIoU increase of 4%, a Sek increase of 2.45%, and a Score increase of 2.91%, respectively. However, the performance of semantic segmentation rely on the initial results of BCD, which easily lead to error propagation issues. Additionally, the two-

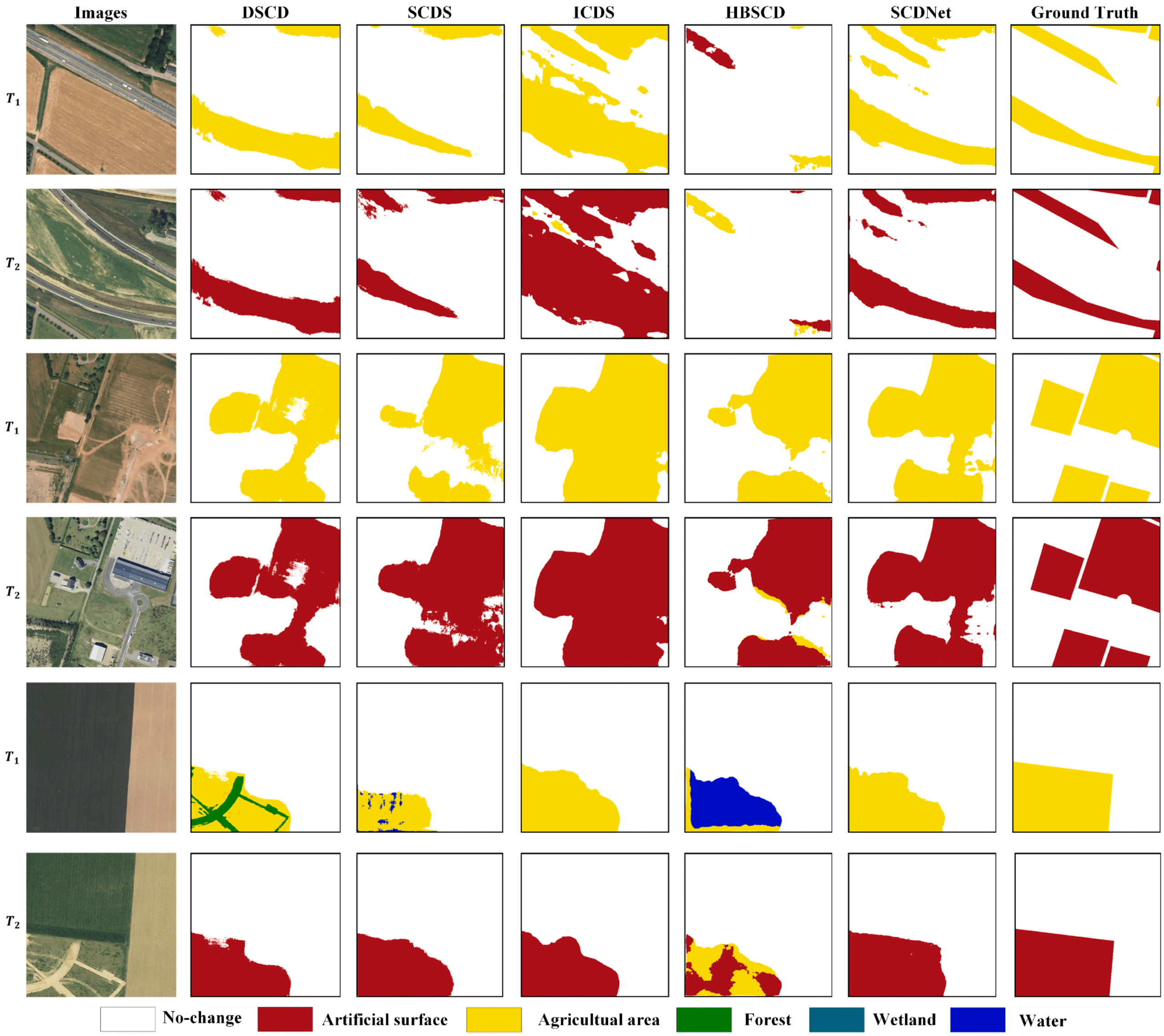


Fig. 11. Visual comparisons of the change maps obtained by different methods on the second three image pairs for HRSCD dataset.

Table 2
Summary of model parameters and computational complexity for different methods

Method	Params(M)	Flops(GMac)
DSCD	42.03	109.47
SCDS	79.97	285.67
ICDS	44.45	122.44
HBSCD	46.17	128.87
Proposed SCDNet	39.62	116.98

stage training mode in SCDS also makes it impossible to optimize the objective function jointly. Through generating binary change maps and semantic change maps simultaneously, it is possible to solve SCD task in an end-to-end way, where BCD and semantic segmentation can be optimized jointly. Benefiting from such advantages, ICDS outperforms SCDS with a mIoU gain of 0.38%, a Sek gain of 3.09%, and a Score gain of 2.28%. It is noteworthy that our proposed SCDNet can further improve the performance over ICDS, with an improvement of mIoU of 0.97%, an improvement of Sek of 0.81%, and an improvement of Score of 0.85%, respectively. This is due to the reason that binary change maps have to be generated explicitly to serve as guidance for semantic

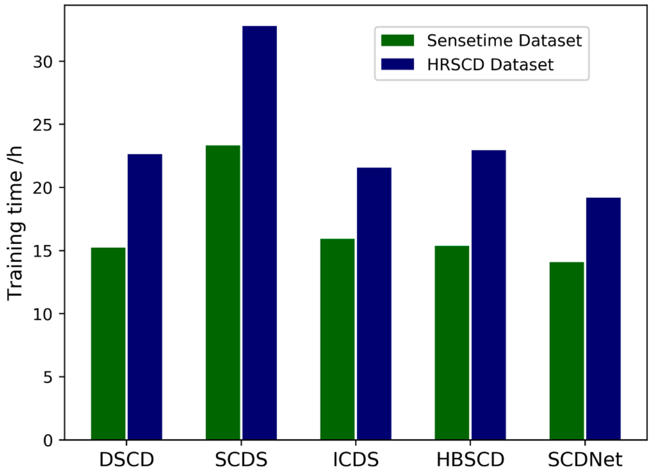
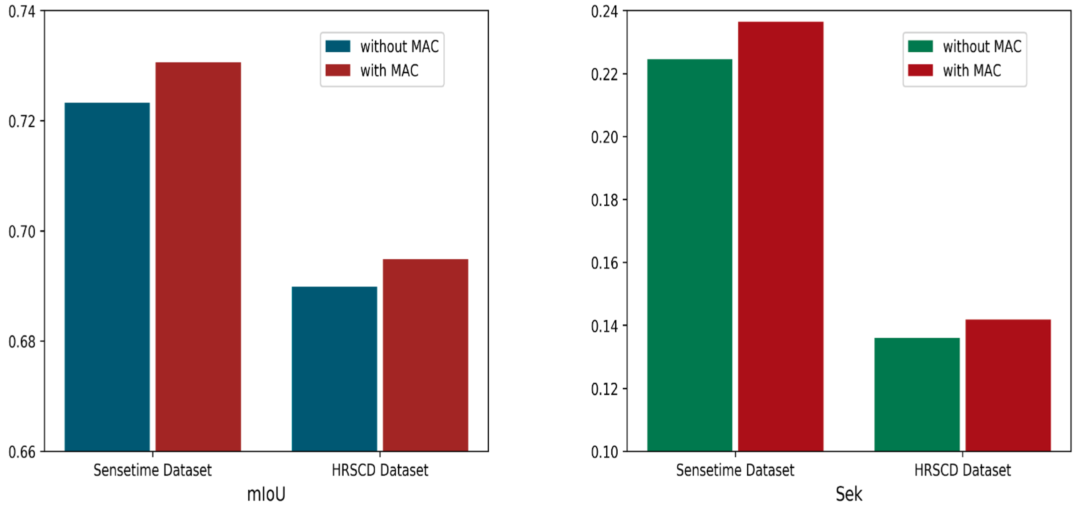


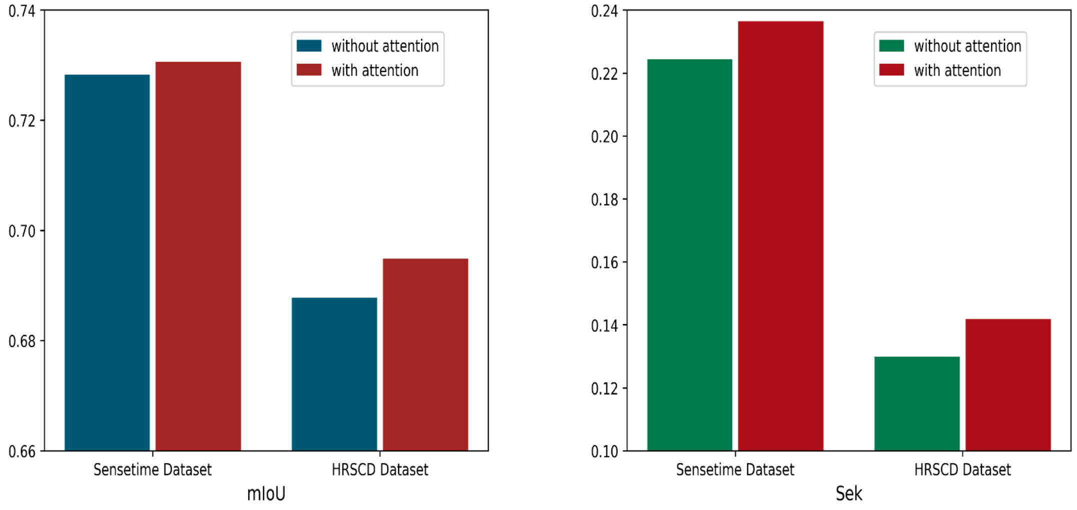
Fig. 12. Visual comparison of training time for different methods.



(a)

(b)

Fig. 13. The effect of MAC unit on the quantitative performance of the proposed SCDNet method. (a) Sensetime dataset. (b) HRSCD dataset.



(a)

(b)

Fig. 14. The effect of attention unit on the quantitative performance of the proposed SCDNet method. (a) Sensetime dataset. (b) HRSCD dataset.

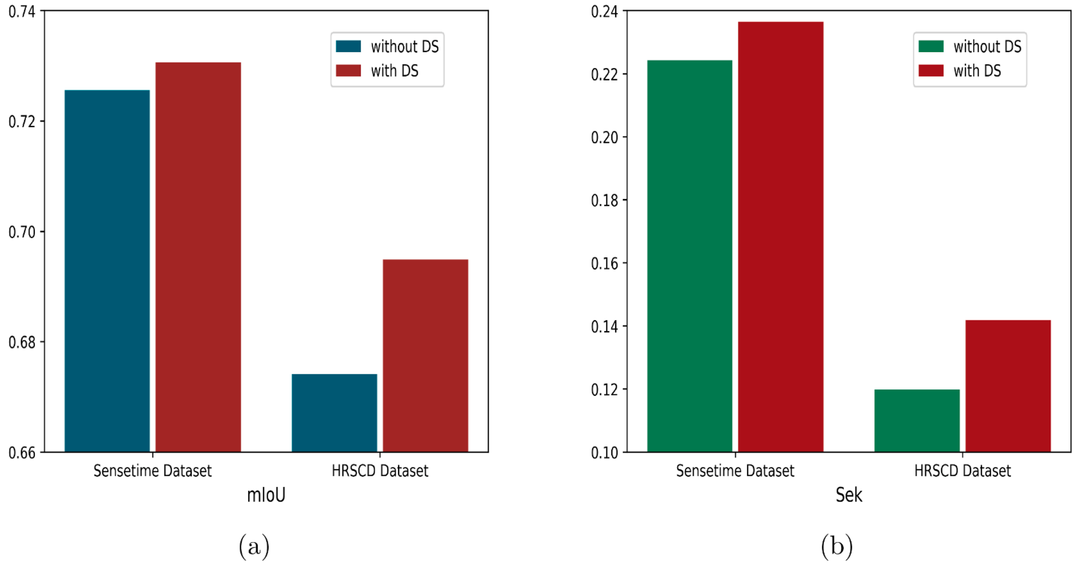
segmentation in ICDS, which inevitably leads to the troublesome balance issues between different loss terms. Nevertheless, in the proposed SCDNet, difference information is embedded into the network by fusing multi-scale difference feature maps and image feature maps, thus mitigating the limitations of balance issues effectively.

For a more comprehensive comparison, model parameters and computational complexity of different methods are calculated and reported in Table 2. One can observe that our proposed method outperforms other SOTA methods with less model parameters and lower computational complexity. This is of great significance when considering training large-scale remote-sensing dataset. Fig. 12 presents the visual comparison of training efficiency by different methods. Due to the usage of two-stage training mode, SCDS requires the highest training time for both datasets. On the contrary, DSCD and ICDS methods benefit from the advantages of end-to-end training, thus the required training time is largely reduced. Almost the same time is needed in DSCD and ICDS for Sensetime dataset and HRSCD dataset. Compared with DSCD and ICDS, HBSCD requires similar training time due to comparable model parameters and computational complexity. Note that due to smaller computational complexity and end-to-end training mode, SCDNet requires shorter training time among the considered methods. Therefore, compared with other methods, the proposed SCDNet achieves the best performance on both accuracy and efficiency.

4.5. *Discussion*

4.5.1. *Effect of the MAC unit*

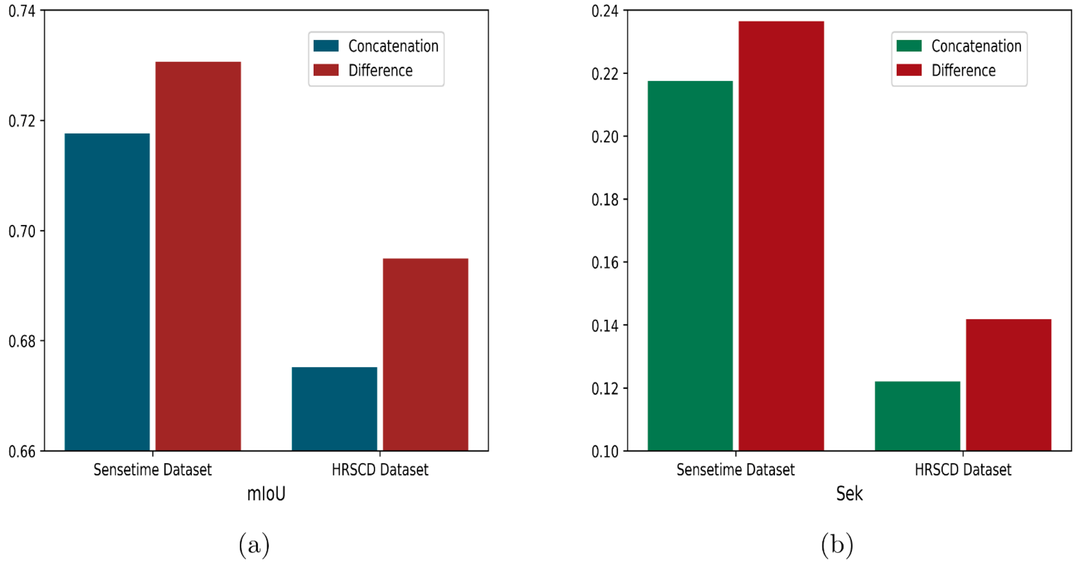
In order to enlarge the receptive field after a series of convolution and pooling operations, a MAC unit is used by cascading different dilated convolution operations in a parallel manner, where multi-scale information can be well exploited. To verify the effectiveness of this unit, an ablation study has been conducted, as shown in Fig. 13. One can observe that better performance can be achieved by using the proposed MAC unit for both datasets, with a mIoU increase of 0.73% and 0.5%, and a Sek increase of 1.19% and 0.58% for the Sensetime dataset and HRSCD dataset, respectively. This demonstrates the effectiveness of the MAC unit in capturing multi-scale object changes for SCDNet.



(a)

(b)

Fig. 15. The effect of deep supervision on the quantitative performance of the proposed SCDNet method. (a) Sensetime dataset. (b) HRSCD dataset.



(a)

(b)

Fig. 16. The effect of feature concatenation and difference on the quantitative performance of the proposed SCDNet method. (a) Sensetime dataset. (b) HRSCD dataset.

4.5.2. *Effect of the attention unit*

Due to the existence of semantic gap between high- and low-level feature maps, direct fusing through concatenation operation will easily lead to information confusion. To address this issue, an attention unit is adopted to re-weight input features. To be specific, an attention map is learned by using high-level feature maps, which aims to re-calibrate the low-level feature maps before concatenation operation. Fig. 14 presents the influence of the attention unit. We can conclude that the SCDNet benefits from the proposed attention mechanism for both datasets, with a mIoU improvement of 0.23% and 0.71%, and a Sek improvement of 1.21% and 1.19% for the Sensetime and the HRSCD datasets, repectively. The reason lies in the fact that low-level features are re-calibrated by a learned attention map, whereby high- and low-level features can be combined more effectively.

4.5.3. *Effect of the deep supervision*

To overcome gradient vanishing and improve network performance, a deep supervision (DS) strategy is employed, where semantic change maps from three intermediate layers are generated for final loss calculation. To validate the effectiveness of this strategy, we have conducted an ablation study. Fig. 15 reports the effect of deep supervision strategy on the accuracy of SCDNet in terms of mIoU and Sek. As one can see, the DS strategy improves the model performance, with a gain of mIoU of 0.5% and 2.08%, and a gain of Sek of 1.22% and 2.2% for the Sensetime and the HRSCD datasets, respectively. This can be explained by the following reasons: 1) more gradient information is propagated into intermediate layers during backward gradient propagation process, whereby gradient vanishing is largely reduced; 2) instead of only producing main outputs only from feature maps of the last layer, multi-scale feature maps are used effectively to generate side outputs, thus improving the network convergence performance as well as providing more regularization constraints.

4.5.4. *Effect of feature difference versus concatenation*

In the process of generating the semantic change map for each period of image, feature maps from two periods of images have to be fused

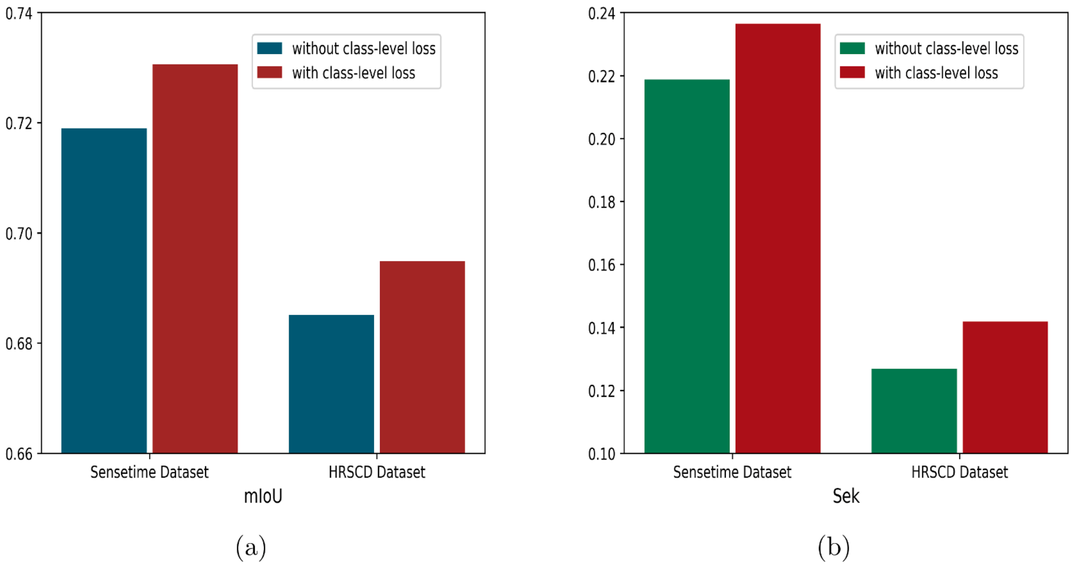


Fig. 17. The effect of class-level combination loss on the quantitative performance of the proposed SCDNet method. (a) Sensetime dataset. (b) HRSCD dataset.

either by feature map difference or by concatenation operation. To validate the effect of the different strategies, we have conducted a comparative study. Fig. 16 presents the accuracy of different approaches. We can conclude that feature difference strategy stably outperforms the feature concatenation strategy, with an improvement of mIoU of 1.3% and 1.97%, and an improvement of Sek of 1.9% and 1.98% for the Sensetime and the HRSCD datasets, respectively. This is due to: 1) difference image is generated through feature difference operation, where change information is embedded into the network implicitly; 2) to maintain channel consistency of feature maps, a convolutional layer is needed after the concatenation operation for the purpose of channel-dimension reduction, which inevitably leads to inferior performance caused by the increase of network parameters and training burden.

4.5.5. *Effect of the loss function*

To address the class imbalance issues, a class-level combination loss $\mathcal{L}_{df}$ is employed by using class-wise dice loss and focal loss. To verify its effectiveness, an ablation study is conducted. Fig. 17 presents the effect of the class-level combination loss on the quantitative accuracy of proposed SCDNet. One can observe a performance gain for both datasets. For the Sensetime dataset, the class-level combination loss yields an improvement of 1.16% in mIoU and 1.77% in Sek. The gain is of 0.98% in mIoU and 1.49% in Sek for the HRSCD dataset. This is due to the reason that more weight is assigned to the objects of small proportions, thus enforcing the network to generalize better to such categories. In addition, the advantages of dice loss and focal loss are combined to further improve the network performance.

5. Conclusion

In this paper, we propose a novel semantic change detection architecture named SCDNet, which is aimed to solve SCD task for large-scale RS datasets in an end-to-end manner. To generate semantic change maps for each period of input image, SCDNet consists of two parallel encoders and decoders with shared weights. The former is intended to extract multi-scale deep feature maps, while the latter is used to decode change information by combining deep feature maps and difference feature maps. To exploit multi-scale information, a MAC unit is introduced at the end of the encoders. An attention mechanism is also adopted to fuse feature maps between encoders and decoders effectively. To avoid gradient vanishing and improve network performance, a deep supervision strategy is used by generating multi-scale semantic change maps for intermediate layers. Dropblock module is further included before the softmax layers, which aims to improve network generalization ability. The effectiveness of the proposed method is verified on two VHR SCD datasets. The experimental results demonstrate that SCDNet stably surpasses other SOTA methods on both visual comparisons and quantitative accuracy metrics. In the future, more effective CNN architectures will be investigated to further improve SCD performance.

CRediT authorship contribution statement

Daifeng Peng: Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing, Funding acquisition. **Lorenzo Bruzzone:** Conceptualization, Formal analysis, Writing – review & editing. **Yongjun Zhang:** Resources, Writing – review & editing. **Haiyan Guan:** Formal analysis, Visualization, Funding acquisition. **Pengfei He:** Investigation, Validation, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported the National Natural Science Foundation of China (41801386, 41671454), the Natural Science Foundation of Jiangsu Province (BK20180797, BK20190785), the Startup Project for Introducing Talent of Nanjing University of Information Science and Technology (NUIST) (2018r029), and the China Scholarship Council (201908320183).

References

Alcantarilla, P.F., Stent, S., Ros, G., Arroyo, R., Gherardi, R., 2018. Street-view change detection with deconvolutional networks. Autonom. Robots 42, 1301–1322.

Bovolo, F., Bruzzone, L., 2015. The time variable in data fusion: A change detection perspective. IEEE Geosci. Remote Sens. Mag. 3, 8–26.

Bruzzone, L., Bovolo, F., 2012. A novel framework for the design of change-detection systems for very-high-resolution remote sensing images. Proc. IEEE 101, 609–630.

Bruzzone, L., Prieto, D.F., 2000. An adaptive parcel-based technique for unsupervised change detection. Int. J. Remote Sens. 21, 817–822.

Chen, H., Li, W., Shi, Z., 2021. Adversarial instance augmentation for building change detection in remote sensing images. IEEE Trans. Geosci. Remote Sens.

Chen, H., Shi, Z., 2020. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. Remote Sens. 12, 1662.

Chen, L.C., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv: 1706.05587.

Cheng, W., Zhang, Y., Lei, X., Yang, W., Xia, G., 2020. Semantic change pattern analysis. arXiv preprint arXiv: 2003.03492.

Daudt, R.C., Le Saux, B., Boulch, A., Gousseau, Y., 2019. Multitask learning for large-scale semantic change detection. Comput. Vis. Image Underst. 187, 102783.

De Alwis Pitts, D.A., So, E., 2017. Enhanced change detection index for disaster response, recovery assessment and monitoring of accessibility and open spaces (camp sites). Int. J. Appl. Earth Observ. Geoinform. 57, 49–60.

Doxani, G., Karantzalos, K., Tsakiri-Strati, M., 2012. Monitoring urban changes based on scale-space filtering and object-oriented classification. Int. J. Appl. Earth Obs. Geoinf. 15, 38–48.

Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H., 2019. Dual attention network for scene segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3146–3154.

Gao, J., Liu, Y., 2010. Determination of land degradation causes in tongyu county, northeast china via land cover change detection. Int. J. Appl. Earth Obs. Geoinf. 12, 9–16.

Ghiasi, G., Lin, T.Y., Le, Q.V., 2018. Dropblock: A regularization method for convolutional networks. Adv. Neural Inform. Process. Syst. 31, 10727–10737.

Guo, E., Fu, X., Zhu, J., Deng, M., Liu, Y., Zhu, Q., Li, H., 2018. Learning to measure change: Fully convolutional siamese metric networks for scene change detection. arXiv preprint arXiv: 1810.09111.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hou, B., Wang, Y., Liu, Q., 2017. Change detection based on deep features and low rank. IEEE Geosci. Remote Sens. Lett. 14, 2418–2422.

Kataoka, H., Shirakabe, S., Miyashita, Y., Nakamura, A., Iwata, K., Satoh, Y., 2016. Semantic change detection with hypermaps. arXiv preprint arXiv: 1604.07513 2.

Lee, C.Y., Xie, S., Gallagher, P., Zhang, Z., Tu, Z., 2015. Deeply-supervised nets. In: Artificial intelligence and statistics. PMLR, pp. 562–570.

Lei, Y., Peng, D., Zhang, P., Ke, Q., Li, H., 2020. Hierarchical paired channel fusion network for street scene change detection. IEEE Trans. Image Process. 30, 55–67.

Leichtle, T., Geiß, C., Wurm, M., Lakes, T., Taubenböck, H., 2017. Unsupervised change detection in vhr remote sensing imagery–an object-based clustering approach in a dynamic urban environment. Int. J. Appl. Earth Obs. Geoinf. 54, 15–27.

Li, Y., Martinis, S., Plank, S., Ludwig, R., 2018. An automatic change detection approach for rapid flood mapping in sentinel-1 sar data. Int. J. Appl. Earth Observ. Geoinform. 73, 123–135.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Liu, Y., Pang, C., Zhan, Z., Zhang, X., Yang, X., 2020. Building change detection for remote sensing images using a dual-task constrained deep siamese convolutional network model. IEEE Geosci. Remote Sens. Lett.

Loshchilov, I., Hutter, F., 2017. Decoupled weight decay regularization. arXiv preprint arXiv: 1711.05101.

Milletari, F., Navab, N., Ahmadi, S.A., 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). IEEE, pp. 565–571.

Peng, D., Bruzzone, L., Zhang, Y., Guan, H., Ding, H., Huang, X., 2020. Semicdnet: a semisupervised convolutional neural network for change detection in high resolution remote-sensing images. IEEE Trans. Geosci. Remote Sens.

Peng, D., Zhang, Y., Guan, H., 2019. End-to-end change detection for high resolution satellite images using improved unet++. Remote Sens. 11, 1382.

Rokni, K., Ahmad, A., Solaimani, K., Hazini, S., 2015. A new approach for surface water change detection: Integration of pixel level image fusion and image classification techniques. Int. J. Appl. Earth Obs. Geoinf. 34, 226–234.

Ru, L., Du, B., Wu, C., 2020. Multi-temporal scene classification and scene change detection with correlation based fusion. arXiv preprint arXiv:2006.02176.

Ru, L., Wu, C., Du, B., Zhang, L., 2019. Deep canonical correlation analysis network for scene change detection of multi-temporal vhr imagery. In: 2019 10th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp). IEEE, pp. 1–4.

Sakurada, K., Shibuya, M., Wang, W., 2020. Weakly supervised silhouette-based semantic scene change detection. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 6861–6867.

Shi, W., Zhang, M., Zhang, R., Chen, S., Zhan, Z., 2020. Change detection based on artificial intelligence: State-of-the-art and challenges. Remote Sens. 12, 1688.

Tian, S., Ma, A., Zheng, Z., Zhong, Y., 2020. Hi-ucd: A large-scale dataset for urban semantic change detection in remote sensing imagery. arXiv preprint arXiv: 2011.03247.

Varghese, A., Gubbi, J., Ramaswamy, A., Balamuralidhar, P., 2018. Changenet: A deep learning architecture for visual change detection. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 0–0.

Volpi, M., Tuia, D., Bovolo, F., Kanevski, M., Bruzzone, L., 2013. Supervised change detection in vhr images using contextual information and support vector machines. Int. J. Appl. Earth Obs. Geoinf. 20, 77–85.

Wang, Y., Du, B., Ru, L., Wu, C., Luo, H., 2019. Scene change detection via deep convolution canonical correlation analysis neural network. In: IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 198–201.

Wu, C., Zhang, L., Du, B., 2017. Kernel slow feature analysis for scene change detection. IEEE Trans. Geosci. Remote Sens. 55, 2367–2384.

Wu, C., Zhang, L., Zhang, L., 2016. A scene change detection framework for multi-temporal very high resolution remote sensing images. Signal Process. 124, 184–197.

Yang, K., Xia, G.S., Liu, Z., Du, B., Yang, W., Pelillo, M., 2020. Asymmetric siamese networks for semantic change detection. arXiv preprint arXiv: 2010.05687.