SiUNet3+-CD: a full-scale connected Siamese network for change detection of VHR images

Bo Zhao*, Panpan Tang*, Xiaoyan Luo*, Linze Li* and Shi Bai*

*Research Center of Big Data Technology, Nanhu Laboratory, Jiaxing, Zhejiang, China; *School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, Hubei, China

ABSTRACT
Change detection is a core issue in the study of global change. Inspired by recent success of the UNet3+ architecture originally designed for image semantic segmentation, in this article we proposed a densely connected siamese network for change detection, namely Pre-SiUNet3+-CD (the combination of Pre-processing, Siamese network and UNet3+). First, our proposed pre-processing algorithm can mitigate effects of poor co-registration between bitemporal images, thus alleviating the loss of localization information in the change map. Second, several modifications have been made to UNet3+ in order to improve its fit for change detection tasks using high-resolution imagery and generate highly discriminative and informative representations to locate changed pixels. The effectiveness of the proposed method is demonstrated on several datasets, and experimental results indicated that our model provides very competitive accuracies in terms of precision, recall, F1-score, and visual performance among all the compared methods. This is because it inherits most of the advantages of UNet3+ in object location and boundary production, and the introduction of pre-processing also gives it a significant accuracy boost.

Introduction
Land use/cover change is the result of the combined effects of natural processes and human activities and is thus a core issue in the study of global change (F. Huang et al., 2019). The objective of land use/cover change detection (CD) is to detect pixels with “semantic change” between multitemporal remote sensing (RS) images acquired at different times and in the same area (Hussain et al., 2013). It has a wide range of applications in disaster assessment, environmental monitoring, urban planning, map revision, agriculture investigation, and so on (Tang et al., 2021). While the remote sensing change detection algorithms have shown many benefits in various fields of applications, it faces several serious challenges (Onur et al., 2009). For example, the final change map should not contain “non-semantic” change, such as changes caused by sensor noise, illumination variation, shadow, camera motion, misregistration error (Wiratama & Sim, 2019), etc. Another difficulty in CD is that the definition of “change” may vary depending on the application and the subjective consciousness of the person (M. Wang et al., 2020). For example, in many cases, bi-temporal images are obtained from different seasons, and “Change” is defined as changes in man-made facilities such as buildings and cars, while seasonal changes in trees, agricultural fields, etc., are regarded as interference factors (X. Huang et al., 2014; Liu et al., 2004; Q. Wang et al., 2019). Therefore, most of the traditional CD methods, e.g., Slow Feature Analysis (Wu et al., 2014), Robust Change Vector Analysis (RCVA; Thonfeld et al., 2016), Image Difference/Ratio, PCA & K-means (Celik, 2009), which can achieve effective results in some simple scenarios, often perform poorly in these complex scenarios, especially for very-high-resolution (VHR) RS imagery.

In the past few decades, the interest of the RS community towards deep learning methods for CD is growing fast, for the benefits of human-like reasoning and robust features which embody the semantics of input images (John & Walters, 2019). Many attempts have been made to solve CD problems using deep learning techniques. First, UNet-based models (Rodrigo et al., 2018) such as FC-EF, FC-Siam-conc, and FC-Siam-diff takes the lead and establish a benchmark model; then, the Siamese network with shared weights is widely used and become the standard method for RS change detection. To improve the performance of CD further, there are a lot of efforts on deep feature extraction and refinement. For example, in Z. Cao et al. (2020), the High-Resolution module (Sun et al., 2019) was transferred into the Siamese network. It can maintain high-resolution representation throughout the whole process and repeatedly fuse multi-resolution representations to obtain rich feature representations. In particular, it can reduce information loss for small objects. Zhang and Lu (2019) proposed a Spectral-Spatial Joint Learning...
Network for change detection. First, the spectral-spatial joint representation is extracted from the network similar to the Siamese CNN. Second, the extracted features are fused to represent the difference information that proves to be effective for CD tasks. Third, the discrimination learning is presented to explore the underlying information of obtained fused features to better represent the discrimination. Also the attention mechanisms are introduced to refine features and obtain better feature representations, such as the spatial and channel attentions (J. Chen et al., 2020), self-attention (Hao & Shi, 2020), gated attention (Zhang et al. 2020a), etc. In addition, several novel algorithms for data augmentation and change detection were proposed leveraging generative adversarial networks (GANs; Chen et al., 2021). A good overview of deep learning-based remote sensing CD technologies can be found in Shi et al. (2020) which focuses on the state-of-the-art algorithms, applications, and challenges of deep learning for CD.

Recently, the semi-supervised CD (M. Yang et al., 2019; Peng et al., 2021; Zheng et al., 2021, etc.), semantic CD (K. Yang et al., 2021), and Transformers-based CD (Chen et al., 2022) have gradually become the focus of RS community. First, a common limitation of using deep learning algorithms for CD is the poor availability of already-labeled datasets (Zhang & Lu, 2019), and semi-supervised CD technologies were put forward to extract discriminative and useful features from a large amount of unlabeled data in addition to limited labeled samples. As a mainstream algorithm, the discriminator of a well-trained GAN is just right for this (Jiang et al., 2019). Second, the reason why people paid much attention on semantic CD technologies is that: in practical application, we hope to simultaneously extract changed regions and identify their land use/cover classes in bitemporal images. As a mainstream algorithm, semantic CD models are usually composed of two CNN networks: one for binary CD - delineating the change range, and the other for image semantic segmentation – detecting the change of all object types (M. Wang et al., 2020), but they are computationally inefficient. Third, recently, using convolution-free transformer architecture for CD tasks has become a new research direction in the literature (Chen et al., 2022; Zheng et al., 2021). At present, Transformers are state-of-the-art in deep learning analysis, and it has more advantages in reducing architecture complexity and training efficiency, exploring scalability, and improving CD performance. However, these methods are not matured enough yet, and a detailed discussion of them is outside the scope of this article. Fully supervised CNN is still an important methodology in CD tasks, and is the focus of our research as well.

Inspired by recent success of the UNet3+ model (H. Huang et al., 2020) originally designed for image semantic segmentation, in this paper we report a novel Siamese UNet3+ network for CD. With the help of full-scale skip connections between encoder and decoder, and between decoder and decoder, UNet3+ can maintain diagnostic, high-resolution, and fine-grained feature representations. To be specific, unlike UNet and UNet++ (Zhou et al., 2018), UNet 3+ takes advantage of full-scale skip connections: each decoder layer in UNet 3+ incorporates both smaller- and same-scale feature maps from encoder and larger-scale feature maps from decoder (illustrated in Figure 1), in other words, it can incorporate low-level details with high-level semantics from feature maps in different scales, so it is especially benefiting for change objects that appear at varying scales. However, there are less reports on the study of siamese products of UNet3+ for CD purpose, and the aim of this article is to bridge this gap. To this end, our solutions, which are also the main contributions of this article, are threefold as follows: (1) preliminarily solve the problem of mis-registration errors between two input images by adding a novel pre-processing module; (2) further optimize the network architecture so that it is applicable to CD cases; and (3) design and incorporate a new outputting module for deep supervision. For example, using siamese UNet++ as backbone, the deep supervision in Peng et al. (2019) is implemented by incorporating the MOSF (multiple side-output fusion) module as the outputting module, and the accuracy of CD is thus improved. Likewise, Fang et al. (2021) incorporated the ECAM (ensemble channel attention mechanism) module for deep supervision, through which the most representative features of different semantic levels can be refined and used for the final classification. For simplicity, we name the proposed algorithm as SiUNet3+-CD, where “Si” is short for “Siamese”.

The organization of this paper is as follows. In Section 2, our proposed network model will be described in detail, which consists of the network architecture, how to train the network and how to predict the testing samples by the trained model. In Section 3, the effectiveness and generalization of SiUNet3+-CD is verified on real data sets. Section 4 gives the result analysis and discussion. Section 5 concludes the article.

Methodology

A novel pre-processing module

RCVA

It is reported that given two co-registered images taken at different times, the illumination variations and misregistration errors often cause local spectral variation and overwhelm the real object changes. One way to settle these problems is to consider pixel neighborhood effects, and the Robust Change Vector Analysis (RCVA) proposed by Thonfeld et al. (2016) is such an example. The calculation of RCVA is performed in two steps:
As formulated below, RCVA is an improvement of the widely used CVA (change vector analysis) algorithm. It is developed to account for pixel neighborhood effects by computing the least difference of each pixel in multiple bands in a moving window of $2w + 1$ in size. In other words, it analyzes not only the pixels with position $j, k$ in images $t_1(j, k)$ and $t_2(j, k)$, but also the pixels in the adjacent neighborhood $t_1$, $t_2$ $(j \pm w, k \pm w)$. It proceeds under the basic assumption that a pixel in $t_2(j \pm w, k \pm w)$ showing the least spectral variance to $t_1(j, k)$ is the pixel containing most of the corresponding ground information of $t_1(j, k)$. Here we assign $w = 1$, resulting in a $3 \times 3$ moving window.

$$x_{\text{diff}}(j, k) = \min_{(p, q) \in \{j-\pm w, k \pm w\}} \left\{ \sum_{i=1}^{N} (t'_i(j, k) - t'_i(p, q))^2 \right\}$$

$$x_{\text{diff}}(j, k) = \min_{(p, q) \in \{j-\pm w, k \pm w\}} \left\{ \sum_{i=1}^{N} (t'_i(j, k) - t'_i(p, q))^2 \right\}$$

where $N$ is the number of image bands.

We combine the two difference images in a second calculation as:

- $M = x_{\text{diff}}(j, k)$, if $x_{\text{diff}}(j, k) \geq x_{\text{diff}}(j, k)$
- $M = x_{\text{diff}}(j, k)$, if $x_{\text{diff}}(j, k) < x_{\text{diff}}(j, k)$

Finally, the RCVA-based change magnitude map $(M)$ is obtained.

**Convert image $t_2$ to image $t_2'$**

RCVA provides an effective scheme for pixel correspondence between two images, or say, image matching (Thonfeld et al., 2016). As shown in Figure 2, the proposed solution is to rearrange the position of pixels in the time-$t_2$ image, and the implementing steps are as follows:

Step 1: corresponding to an arbitrary pixel $(i, j)$ in Image $t_1$, there are nine pixels in a moving window in Image $t_2$, and then, we can obtain nine Xdiff values. Find the pixel pair (or say “link”) having the minimum Xdiff. For example, in Figure 2 we suppose that link 1 ($(i, j)$ in Image $t_1$ versus $(i-1, j-1)$ in Image $t_2$) has the minimum Xdiff.

Step 2: exchange the pixel values between R/G/B(i, j) and R/G/B(i-1, j-1) or other pixel values in the adjacent neighborhood, so that the time-$t_2$ image can be reorganized.

Step 3: repeat the above procedure and traverse all the pixels in Image $t_1$ and Image $t_2$, and then a reorganized time-$t_2$ image – Image $t_2'$ will be created.

To produce Image $t_2'$, we stipulate that the pixel in Image $t_2$ that has already been replaced/exchanged cannot be replaced/exchanged again.

It is less rigorous to claim that the misregistration error between $t_1$ and $t_2$ is reduced using RCVA; however, the negative impacts of the misregistration errors on local spectral variation between $t_1$ and $t_2$ is indeed mitigated. In other words, RCVA considers a neighborhood around each pixel to mitigate effects...
of poor co-registration between multitemporal images. Inasmuch, SiUNet3+-CD will operate between Image \( t_1 \) and Image \( t_2' \), rather than between \( t_1 \) and \( t_2 \).

On the other hand, our proposed method may fail when facing a changed region, and as a result, the relevant changing information will be weakened. It may miss small changes or disturb the boundaries of changed regions due to the use of the moving window. Worse still, the rationality of using RCAV for pixel correspondence might be in doubt in practical application because of high intra-class difference, illustration and seasonal changes in high-resolution images. In spite of these, considering the worser performance yielded by CD algorithms without pre-processing or with traditional pre-processing operations like SIFT, template matching, etc. (see the article below), our proposed algorithm is still helpful in improving the CD accuracy and sensitivity. Its rationality will be further discussed in Section 4.1.

**Modifications of the UNet3+ Network architecture**

To make the original UNet3+ model applicable to CD, four modifications are suggested in this article:

1. As illustrated in Figure 3(a), using UNet3+ as backbone, SiUNet3+ is a typical encoder-decoder architecture and the siamese network act as encoder. Two temporal RS images are input into two branches of siamese network, respectively, and the siamese network separately extracts bi-temporal feature representations, but the parameters between them are shared (Z. Cao et al., 2020). There are five blocks attached to each siamese branch. Since in CD we are trying to detect differences between the two temporal images: \( t_1 \) and \( t_2' \), and inspired by the success of FC-Siam-diff proposed by Rodrigo et al. (2018), here the absolute values of the difference between block \( X^a_t \) and block \( X^b_t \) (\( n = 1, \ldots, 5 \)) are recorded, and then, they are concatenated to the decoder block \( X^d_t \) from five scales.

2. Conventional siamese networks identify changes by concatenating/subtracting low-level features in the encoder stage, that is to say: they conduct an early change feature extraction of two image inputs, which is sensitive to noisy conditions such as geometric distortion and different viewing angles (Wiratama et al., 2018; Wiratama & Sim, 2019). To alleviate this problem, we deepen the encoder network architecture by designing the affiliated convolution blocks as a residual unit structure illustrated in Figure 3(b), while keeping the decoder convolutional structure unchanged as in Figure 1 & 3(c). In Figure 3(b), the shortcut connection is added after the first convolution layer to maintain the unity of all convolution blocks (Fang et al., 2021), and the reason why the simple “Conv+GN+ReLU” structure is retained in each decoding stage is to avoid introducing too many parameters to SiUNet3+-CD.

The siamese structure of the encoder side is similar to the structure of SNUNet-CD, but two things work differently: (1) SNUNet-CD uses the concatenation operation to integrate the feature representations of bi-temporal images at different scales (Fang et al., 2021), whereas we use the subtraction operation. The parameter number of the former is twice as many as the latter. (2) In SNUNet-CD, only eight convolution blocks \( (X^a_t \) to \( X^4_t \) and \( X^b_t \) to \( X^4_t ) \) are involved in the siamese structure, whereas our model has 10 blocks \( (X^a_t \) to \( X^5_t \) and \( X^b_t \) to \( X^5_t ) \) involved, as displayed in
Figure 3(a). That is to say, our model is able to capture additional semantic features. The rationality of these two modifications will be discussed in Section 4.4.

(3) We could not train a complicated model like SiUNet3+ with larger batch size as depicted in the literature due to the limited memory of our GPU (in this paper, all experimental setups were conducted using the PyTorch deep learning library on a single NVIDIA GeForce RTX 2080 with 8 GB of GPU memory). This could significantly degrade the CD performance since the batch normalization (BN), which is widely used in the original UNet3+ network, depends heavily on the batch size (Wu & He, 2018). The proposed model has 27.00 M parameters which is much more than UNet3+, SNUNet-CD (Fang et al., 2021), etc., and the batch size can only be set to 1 or 2. Inasmuch, we replace all the BN layers in UNet3+ with group normalization (GN) layers in SiUNet3+ (the number of groups is set to 32). GN is the counterpart of BN, and it can divide the feature-map channels into groups and computes within each group the mean and variance for normalization (Wu & He, 2018). GN’s computation is independent of batch sizes, and its accuracy is stable in a wide range of batch sizes.

(4) The ReLU activation functions are replaced by Leaky ReLU. Problems with ReLU: 1, ReLU is not continuously differentiable, and sometimes the gradient cannot be computed. 2, ReLU sets all values < 0 to zero, and the gradient of 0 is 0. As a result, neurons arriving at large negative values cannot recover from being stuck at 0 (John & Walters, 2019), especially when neurons are not initialized properly or when the data is not normalized very well, causing significant weight swings. Leaky ReLU helps resolve these problems in theory (John & Walters, 2019), so it is incorporated into our model.

Note that in Figure 3, the backbone network, the convolution blocks, the skip-layer connections, the GN layer, deep supervision, and so on are all based on existing technologies and designs, whereas by recombining them in an appropriate manner, we
demonstrated that it is possible to develop a more powerful model for CD tasks. This is the major innovation of the proposed model.

**Details of loss function**

During the practical CD, the number of unchanged pixels may be much more than the number of changed ones. The best way to alleviate the impact of sample imbalance is to introduce a hybrid and weighted loss function. As defined blow, here we consider the combination of focal loss (Lin et al., 2017) and dice loss (Li et al., 2019).

\[ L = L_{focal} + L_{dice} \]

Therein, \( L_{focal} \) is defined as:

\[ L_{focal}(p_t) = -\alpha_t (1 - p_t)^\gamma \log(p_t) \]

Therein, \( p_t \) is the probability that the model predicts for the ground truth. To deal with class imbalance is to introduce weights. Give high weights to the rare class and small weights to the dominating or common class. These weights are referred to as \( \alpha_t \). As for \( \gamma \), the higher the value of \( \gamma \), the lower the loss for well-classified examples, then we could turn the attention of the model more towards “hard-to-classify” examples, and vice versa. When \( \gamma = 0 \), the equation is equivalent to Cross Entropy Loss. Here in this study, \( \alpha_t \) is set as 0.4, and \( \gamma \) is set as 2.

\( L_{dice} \) is formulated as follows:

\[ L_{dice} = 1 - \frac{I + \varepsilon}{U - I + \varepsilon} \]

\[ I = \sum_{i=1}^{N} t_i y_i \quad \text{and} \quad U = \sum_{i=1}^{N} (t_i^2 + y_i^2) \]

Where the smoothing factor \( \varepsilon \) is an extremely small number; \( y_i \) is the predicted value processed using the sigmoid or softmax activation function, and \( 0 \leq y_i \leq 1 \); \( t_i \) is the label/target value, being either 0 or 1. Here \( \varepsilon \) is set as 1e-7.

**The outputting module**

In Figure 3(a), the backbone of SiUNet3+-CD finally has four outputs that are up-sampled to the same size as the original images using the bilinear method, and each of them has 320 channels. Therefore, for the first submodule, we conduct the convolution operation to reduce the dimensionality of these feature maps from 320 to 1 (here 1 represents the number of class).

In Figure 3(a), instead of producing prediction, i.e., score maps, from only the output of the final stage (namely \( X_{De} \)), the proposed network architecture also produces dense predictions from the outputs of intermediate stages which are \( X_{De}^2 \), \( X_{De}^3 \), and \( X_{De}^4 \). And then, we fuse these multi-resolution predictions which have different semantic levels and spatial position representations. The fusion method, namely the second submodule, is formulated as follows:

\[ S_c = w_c^0 \odot S_c^1 + w_c^2 \odot U(S_c^2, 2) + w_c^3 \odot U(S_c^3, 4) + w_c^4 \odot U(S_c^4, 8) \]

Where \( S_c^i \) represents the score map of class \( c \) obtained from stage \( i \) (i.e., 1, 2, 3, 4); \( U(S_c^i, \text{number}) \) means an up-sampling operation with rate “number”; \( w_c^i \) denotes the weight of prediction produced from stage \( i \) for the class \( c \); \( \odot \) denotes the convolution operation, which can be easily implemented by the \( 1 \times 1 \) depth-wise convolution.

The third submodule is the activation function layer. Change detection can be seen as a binary classification problem that decides whether the change is present, and it seems that the Sigmoid function, which maps any input to a probability value ranging from 0 to 1, is the recommended activation function in the outputting module (John & Walters, 2019; Shi et al., 2020). However, our experiments have demonstrated that vanishing gradients usually happen while using Sigmoid both in the hidden layer and the outputting layer. Inasmuch, here we use Leaky ReLU as an alternative.

The fourth submodule is the GN layer. As the Leaky ReLU function cannot convert the input tensors to probability values ranging from 0 to 1, so the GN layer is adapted here to normalize the Leaky ReLU-activated tensor data.

**A new deep supervision strategy**

In order to learn hierarchical representations from the full-scale aggregated feature maps, the full-scale deep supervision is further adopted in the SiUNet3+-CD.

First, as shown in Figure 3, SiUNet3+-CD yields a side output from each decoder stage, which is supervised by the ground truth. Deep supervision for these intermediate layers is employed as follows:

\[ O^i = GN(UP(f^{3 \times 3}(X_{De}^i))) \quad (i = 1, 2, \ldots, 4) \]

where \( O^i \) denotes the \( i \)th side output; \( f^{3 \times 3} \) denotes a convolution layer with kernel size of 3 \( \times \) 3; \( UP \) means the bilinear up-sampling operation; \( GN \) means the GN-normalization operation; \( X_{De}^i \) represents the \( i \)th decoder block.

Likewise, deep supervision for the outputting layer is employed as follows:

\[ O^{out} = GN(LeakyReLU(S_c)) \]

During the model-training process, the loss of each deep supervision is computed independently and is directly back-propagated to intermediate layers (In total, there are five losses in the network). In this way, the intermediate layers are effectively trained.
and weights of them can be finely updated, and thus, it alleviates the presence of vanishing gradient. By introducing multiple deep supervisions in the network, the performance of difference discrimination network might be improved as well (Zhang et al. 2020b).

In the original UNet3+ network (H. Huang et al., 2020), to realize deep supervision, the last layer of each decoder stage is fed into a plain 3 × 3 convolution layer followed by a bilinear up-sampling and a sigmoid function. In our model, however, the sigmoid function is replaced by the GN layer. This is feasible because the sigmoid/softmax function has already been embedded in the loss functions L_{focal} and L_{dice}.

Experiments

Data set and evaluation metrics

To evaluate our method, a series of comparative experiments were designed on CDD (Change Detection Dataset) and BCDD (Building Change Detection Dataset), which are the most common evaluation datasets in the field of RS change detection. CDD data set contains 11 pairs of RGB images taken in different seasons obtained by Google Earth, and the spatial resolution of them is 3 cm/px to 100 cm/px. There are 10,000 image pairs for the training set, and 3000 pairs for validation and testing sets respectively (Fang et al., 2021). The BCDD dataset covers the area of Christchurch, New Zealand where an earthquake of magnitude 6.3 occurred in February of 2011. It includes Google image pairs from the same location from 2012 to 2016, as well as detection tags of changed buildings. The image pair with a size of 32,507 × 15,354 pixels is divided into non-overlapping pairs of 256 × 256 pixels. Then, it is split into 5748 pairs for training, 744 pairs for validation, and 744 pairs for testing, in a ratio of 8:1:1. By the way, the definition of change in BCDD is different from that of CDD. It only needs to detect changes in buildings, which results in non-uniform distribution of changes in the BCDD image pairs (Tang et al., 2021).

In addition to the Google datasets, we also concern about the performance of SiUNet 3+-CD on other CD datasets. Here the SECOND dataset created by K. Yang et al. (2021) is considered. This is because the sample images in this dataset were collected from several platforms and sensors. There are 2968 pairs of aerial images annotated at the pixel level and each image has size 512 × 512. The ratio of the training, validation, and testing set is still 8:1:1. As for the change type in the SECONd dataset, there are 6 main land-cover classes, namely non-vegetated ground surface, water, buildings, trees, low vegetation, and playgrounds, while in this paper we only focus on the change of buildings.

In this article, three indicators were used to validate effectiveness of the method: Precision, Recall, and F1-Score. The formulas of these metrics are given as follows:

\[
\text{precision} = \frac{TP}{TP + FP} \\
\text{recall} = \frac{TP}{TP + FN} \\
F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}
\]

where TP, FP and FN denote all classes’ total number of true positive, false positive, and false negative, respectively.

Implementation details

We implement SiUNet3+-CD as well as the relevant comparative experiments using the Pytorch framework which is open source. We train the model from scratch, and the batch size is set to 2, and the AdamW optimizer is applied. The original learning rate is set to 0.001 and decays by 0.5 every eight epochs. The KaiMing normalization algorithm was used to initialize the weights of each convolutional layer. As mentioned, we conduct these experiments on a single NVIDIA GeForce RTX 2080 with 8 GB of GPU memory and train for 60 epochs to make the model converge.

For a better illustration, Figure 4 gives the training accuracy versus loss curves of SiUNet3+-CD. It can be seen from Figure 4 that after completing 50,000 iterations, the training precision increases very rapidly, and meanwhile the training loss decreases sharply. And then, the growth of precision becomes slow, eventually stabilizing at 0.9 or so. Meanwhile, the decreasing trend of loss become slow and oscillating, eventually being convergent to 0.36 or so. Such an experimental effect is satisfactory.

Comparison and analysis

The performance of the proposed model

Table 1 gives the performance of our models comparing to other state-of-the-art algorithms on CDD. Therein, FC-EF, FC-Siam-conc, and FC-Siam-diff act as the baseline models for CD. These models are simple combinations of siamese network and UNet (Rodrigo et al., 2018).
DASNet (J. Chen et al., 2020) introduces the dual-attention mechanism to traditional convolution neural network. STANet (Hao & Shi, 2020) introduces the self-attention mechanism to the convolution network. SNUNet-CD could be regarded as a combination of siamese network, NestedUNet, and ECAM. In the deeply supervised image fusion network (IFN) put forward by Zhang et al. (2020), highly representative deep features of bi-temporal input images are firstly extracted through a fully convolutional two-stream architecture, then the extracted features are fed into a deeply supervised difference discrimination network for CD. Finally, multi-level deep features of raw images are fused with image difference features by means of attention modules for change map reconstruction. Note that the proposed pre-processing operation was not conducted for these compared methods in Table 1.

There are several useful observations from Table 1:

(1) Several comparative experiments were conducted and SNUNet-CD gets the best performance with the highest precision, recall, and F1-score in change detection of complex scenarios. FC-EF, FC-Siam-conc, and FC-Siam-diff get the worst performance.

(2) Our model (SiUNet3+-CD) provides competitive CD accuracies in terms of precision and F1-score among all the compared methods, namely the second-best precision and the third-best F1-score. The recall rate of our model is relatively low, but is still competitive when compared with FC-EF, FC-Siam-conc, FC-Siam-diff, IFN, UNet++_MSOF, etc.

(3) SiUNet3+-CD presents worse performance than SNUNet-CD according to the data copied from the original paper of Fang et al. (2021), but our re-implemented results using the public code of SNUNet-CD with minor modifications (To make a fair comparison, all the BN layers were substituted with the GN layers, and ReLU with Leaky ReLU, and an additional GN layer was added to the output side, being consistent with the SiUNet3+-CD network architecture) indicated a much lower precision and recall rate (i.e., 0.914 & 0.842) than SiUNet3+-CD. That is to say, the potential of our model might be underestimated.

(4) In Table 1, we designed a new CD model: SiUNet3+-CA-CD for comparison. As its name implies, we add the channel attention (CA) to SiUNet3+-CD for deep supervision,
however the proposed outputting module is not included in the model. The deep supervision is achieved as follows:

\[ O^t = GN(UP(f^{3\times3}(CA(X^t_{Da}) \ast X^t_{De}))) (i = 1, 2, \ldots, 4) \]

where \( CA \) denotes the channel attention operator; and \( CA(X^t_{Da}) \ast X^t_{De} \) denotes the channel attention map. The implementation method of \( CA \) is detailed in Zhang et al. (2020).

In theory, by utilizing channel attention maps, the channels of \( X^t_{Da} \) contain actual change information will be emphasized, while other channels will be suppressed, since they are redundant and are helpless for the generation of the change map. However, the results reported do not show a significant increase in CD accuracy over SiUNet3+-CD as expected.

In addition, as illustrated in Figure 5, comparing with the visualization results of SiUNet3+-I and STANet, the proposed approach successfully returns the changed areas with relatively complete boundaries and high object internal compactness. However, as the recall rate of our model is relatively low, there are some important details still missing from the predicted maps. On the other hand, although the recall rate of STANet is higher, we cannot say the changed pixels are accurately extracted because of the dilated ROIs shown in Figure 5(d).

The necessity of pre-processing

Table 2 gives the performance of our model comparing to other state-of-the-art algorithms on CDD’1, CDD’2 and BCDD’. In order to validate the necessity of the proposed pre-processing algorithm, we first suppose that there is no misregistration error between the two input images in CDD and BCDD, and then, in CDD’1 the spatial alignment of them was made to contain a relatively small error of up to \( \pm 2 \) pixels, CDD’2 up to \( \pm 5 \) pixels, and BCDD’ up to 2 pixels. Figure 6 illustrates how the CDD’ or BCDD’ dataset was produced. All the mentioned methods in Table 2 were validated using these newly generated datasets.

There are three useful observations from Table 2:

1. The CDD’1 dataset our Pre-SiUNet3+-CD model yields the best precision. The recall rate is relatively low but still competitive. This suggests that the performance of RCVA-based pre-processing operation is robust and effective, and the local reorganization of Image \( t_2 \) has not resulted in the fuzzification of the detailed change information. SiUNet3+-CD yields the second-best precision and a competitive recall and F1-score. FC-Siam-Diff has the worst performance among the six methods.

Figure 5. Visualization results on the CDD testing dataset: (a) and (b) are original bi-temporal images; (c) is the ground truth; (d) is the result of STANet; (e) is the result of SiUNet3+-I; and (f) is the result of SiUNet3+-CD. The changed parts are shown in white, while the unchanged parts are shown in black.
Table 2. Performance comparison on CDD’1, CDD’2, and BCDD’ datasets.

| Dataset | Method                | Precision | Recall | F1-Score |
|---------|-----------------------|-----------|--------|----------|
| CDD’1   | FC-Siam-Diff*         | 0.764     | 0.583  | 0.662    |
|         | STANet*               | 0.840     | 0.871  | 0.855    |
|         | SNUNet3+CD*          | 0.895     | 0.824  | 0.858    |
|         | Pre-SiuNet3+CD*      | 0.943     | 0.868  | 0.904    |
| CDD’2   | FC-Siam-Diff*         | 0.694     | 0.569  | 0.625    |
|         | STANet*               | 0.842     | 0.527  | 0.635    |
|         | SNUNet3+CD*          | 0.886     | 0.810  | 0.846    |
|         | Pre-SiuNet3+CD*      | 0.911     | 0.836  | 0.872    |
| BCDD’   | FC-Siam-Diff*         | 0.398     | 0.540  | 0.453    |
|         | STANet*               | 0.797     | 0.626  | 0.696    |
|         | SNUNet3+CD*          | 0.783     | 0.743  | 0.761    |
|         | Pre-SiuNet3+CD*      | 0.822     | 0.731  | 0.772    |

Best results are highlighted in bold. Pre-SiuNet3+CD denotes the SiUNet3+CD model with pre-processing. Other notes in this table are consistent with those in Table 1. * means our re-implemented results. To ensure the fairness, we adopt the same batch size (2), epochs (60) and original learning rate (0.001) for all the re-implemented algorithms, and the relevant BN layers are all replaced by GN layers, and ReLU by Leaky ReLU.

Method (2) for the CDD’2 dataset, Pre-SiuNet3+CD’s advantages became more apparent: it provides the best precision, recall and F1-score, which implies that the pre-processing phase has great importance in CD tasks to enhance the efficiency and effectiveness of Pre-SiuNet3+CD. SiUNet3+CD yields the second-best results, indicating a good generalization ability and stability in this case.

Comparing to Table 1, the prediction accuracies obtained with different models represent a general declining trend which could be caused by the errors in the image co-registration. In particular, there is an unexpectedly sharp drop of the accuracies of SNUNet-CD. Note that compared with CDD, the BCDD dataset is more challenging due to: 1) large building size and shape changes, where the buildings are more diverse and complex, ranging from large industry and residence houses to small portable dwellings and 2) there are quite a few non-changed image pairs in the dataset, in other words, there are much more negative samples than positive ones. Thereby, the declining trend of the predicting accuracies obtained with different algorithms becomes more significant. Even so, for the BCDD’ dataset, Pre-SiuNet3+CD and SiUNet3+CD still give the best results in terms of precision, recall and F1-score.

It can be seen from Figure 7 that: after pre-processing, pixels in Image t2 with misregistration error were rearranged to match with Image t1, and the most visible sign is that many, if not most or all, of the pixels in the black edges that represent the misregistration error were scattered throughout their neighbourhoods. Unlike traditional image registration methods, our algorithm attempts to locally rearrange the pixels in Image t2, rather than wholly moving it to match with Image t1. Therefore, it is difficult to evaluate the registration accuracy between Image t1 and t2 using traditional metrics. However, the improved performance of Pre-SiuNet3+CD relative to other models in Table 2 has proved its feasibility and effectiveness.

Figure 8 shows the change maps obtained by different deep learning methods for the CDD’2 and BCDD’ datasets. One can observe that there are many false alarms and missed detections in the comparative methods, whereas the proposed SiUNet3+CD and Pre-SiuNet3+CD achieve the best visual performance, as their change maps are more consistent with the ground truth. In addition, due to the poor co-registration, the object boundaries generated by the compared methods do not always coincide with the real changed objects in the ground-truth images, whereas our model can generate change maps with more accurate boundaries. In particular, compared with the baseline method of FC-Siam-Diff, missed detections and false positives, such as missed buildings and cars, and false building change, are largely reduced by Pre-SiuNet3+CD, as shown in Row 1, 2 and 4 of Figure 8. For BCDD’ dataset, FC-Siam-Diff produced quite a few false positives, while STANet produced too many false negatives. There are still some false alarms and missed detections derived by our proposed models, but they obtained the best performance on both quantitative metrics evaluation and visual comparison.

The performance of our model on other CD datasets

As shown in Table 3 and Figure 9, SiUNet3+CD still outperforms other state-of-the-art CD algorithms except ChangeSTAR with bitemporal supervision proposed by Zheng et al. (2021) – the feature-extracting network of it is FarSeg with pre-trained ResNeXt-101 32x4d (Zheng et al., 2020). These imply that: although SiUNet3+CD is not the most optimal solution for CD, it provides competitive accuracies and shows better visual performance (than others including ChangeSTAR) in practical application, thus establishing a new benchmark model for developing more sophisticated change detection algorithms.

The ablation study – testing on the CDD dataset

Note that in Table 4, SiUNet3+-0 denotes the SiUNet3+CD model not using deep supervision, the residual unit structure, as well as the proposed outputting module; SiUNet3+-1 denotes the SiUNet3+CD model not using deep supervision and the residual unit structure shown in Figure 3(b) in the encoder processing; SiUNet3+-II denotes the SiUNet3+CD...
model without the proposed outputting module and deep supervision. SiUNet3+-III denotes the SiUNet3+-CD model without deep supervision.

First, we implemented the basic SiUNet3+-CD without using deep supervision, the residual unit structure, and the outputting module, and took it as the baseline, namely SiUNet3+-0. Our baseline achieved 0.913 precision on the CDD testing dataset with 216.67 G FLOPs (floating point operations).

Next, we added the proposed outputting module to the baseline, namely SiUNet3+-I, and it achieves the performance gain of 0.913 to 0.94 (precision) and 0.800 to 0.858 (recall) with 216.72 G FLOPs.

Then, we added the residual unit structure instead of the "Conv+GN+ReLU" structure to the baseline model, namely SiUNet3+-II, and the precision increased by 3.0%. The accuracy differences of recall and F1-score are significant as well. Obviously, such a residual unit structure contributes most to the performance gain.

After that, we added both the residual unit structure and the outputting module to the baseline – SiUNet3+-III, and the precision improved by 3.5%.

Finally, we added the residual unit structure, the proposed outputting module, and deep supervision to the baseline, namely SiUNet3+-CD, and the precision, recall and F1-score improved by 3.70%, 7.00%, and 5.50%, respectively. At the same time, the number of parameters and FLOPs did not increase very much. The deep supervision, which is able to enhance the representation and discrimination capabilities of shallow or intermediate features, contributes a lot to the performance gain.

As shown in Figure 10, visual comparison confirmed the above observations. The visual performance is gradually improved from SiUNet3+-0 to SiUNet3+-II and SiUNet3+-CD. To be specific, Model SiUNet3+-0 often generates change masks with holes and fragmentized boundaries, while the masks generated by SiUNet3+-CD are very close to the ground truth. The performance of SiUNet3+-II is

![Figure 6. From CDD to CDD'.](image)
Figure 7. Automatic registration effect of our pre-processing model.

Figure 8. Visualization results on the CDD’2 and BCDD’ testing dataset. The changed parts are shown in white, while the unchanged parts are shown in black.
in between. Note that as the accuracy difference between SiUNet3+-I, SiUNet3+-III and SiUNet3+-II is small, so they are not displayed in Figure 10.

Our ablation study has also revealed that: (1) the Leaky ReLU layers have little influence on the model performance. (2) the performance of SiUNet3+-CD using the concatenation operation in the encoder stage is slightly better than that of SiUNet3+-CD with "FC-Siam-diff"-like architecture, but the parameter number of the former is twice as many as the latter. (3) the precision, recall, and IOU of SiUNet3+-CD using BN layers (batch size = 10) are 0.925, 0.780, and 0.734, respectively, which is worse than the performance of SiUNet3+-CD using GN layers.

The efficiency evaluation

As listed in Table 1, our model has 27.00 M parameters, which is much more than the compared models (smaller than 17.00 M) except IFN, see, Figure 11(a). In Figure 11(b), the FLOPs (the number of floating-point operations) of our model come up to 216.72 G, while according to Fang et al. (2021), the FLOPs of IFN is between 150 and 175 G, UNet++_MSOF between 75 and 100 G, DASNet is between 100 and 125 G, FC-Siam-diff is 4.70784 G, and SNUNet-CD (with 32 channels) is 54.76720 G. Such inefficiency is also reflected in the runtime, for example, if setting batch size = 2, it costs 4 mins and 29s for our model to run a forward iteration 200 times during the training phase; while for SNUNet-CD it costs 1 min and 30s; for FC-Siam-diff it only costs 49s. Obviously, it is the full-scale skip connections that give SiUNet3+-CD an expensive computational complexity while producing a relatively good result.

We accept that these may be the main disadvantage of the proposed model, but fortunately, its inference time for each bi-temporal testing image of size 256 × 256 is acceptable (<0.30s on average), and one way to balance the running efficiency and prediction accuracy is to develop a transfer learning model based on pre-trained SiUNet3+-CD. To be specific, freeze the initial weights of blocks in the encoder part of SiUNet3+-CD, and the rest of the weights in the decoder part will be used to compute loss and be updated by the optimizer. This requires less resource than normal training and allows for faster training time, although it may also result in minor reductions to final prediction accuracy. This will be done in our future work.

Discussion

About the pre-processing module

Actually, if the image co-registration is not guaranteed, traditional state-of-the-art algorithms would have worse performance since the CD algorithms are usually very sensitive to this kind of related issues (Di
Figure 10. Visual comparison of SiUNet3+-0, SiUNet3+-II, and SiUNet3+-CD on the CDD testing dataset. Note that the changed parts are shown in white, while the unchanged ones are shown in black.

Figure 11. (a) the number of parameters versus precision; (b) FLOPs versus F1-Score (Modified after Fang et al. (2021)).
Pilato et al., 2021). The proposed pre-processing algorithm, on the other hand, can mitigate the effects of poor co-registration between bitemporal images, thus giving a significant accuracy boost of SiUNet3+–CD.

However, people may question that for high-resolution images with high intra-class difference, illustration and/or seasonal changes, the difference image Xdiffb may not be reliable, and it is difficult to determine the moving window size of Xdiffb either. We accept these shortcomings, but one thing is certain: our co-registering module proceeds under a basic assumption that a pixel \( x_2(j \pm w, k \pm w) \) in the post-event image shows the least spectral variance to \( x_1(j, k) \) in the pre-event image is the pixel containing most of the corresponding ground information of \( x_1(j, k) \). If it is true, differences caused by geometric distortions, image misregistration, and shifts of the instantaneous field of view can thus be minimized by the consideration of the pixel neighborhood (Thonfeld et al., 2016) in the co-registering process. Thereby, the only question is whether this assumption holds true in the complex scenarios as aforementioned.

Our conclusion is that for CDD and BCDD datasets, it holds true because (1) the changed objects in these datasets are mainly man-made features such as cars, buildings, roads, or other impervious surfaces, and the intra-class spectral variation of them is relatively homogeneous. (2) The spectral differences between man-made features and the background are usually significant, even though there are seasonal changes, and the RCVA algorithm is good at collecting and distinguishing such differences. (3) As there is a moderate height (below 15 m) of most buildings in both datasets, so the illustration changes have limited impact on Xdiffa and Xdiffb. On the other hand, the assumption may be incorrect if a pixel \( x_1(j, k) \) in the pre-event image shows the least spectral variance to several pixels within \( (j \pm w, k \pm w) \) in the post-event image. This happens from time to time in practical application, and we have to randomly choose a pixel \( x_2 \) to match with \( x_1 \). In general, it is not possible to eliminate the effects introduced by illumination variations and misregistration errors completely (Thonfeld et al., 2016), and the residual black edges in Figure 7 (row 2 and column 4) is such an example. Our goal is just to minimize, rather than eliminate, these effects. If the changing objects of interest are high-rise buildings, farmlands, forests, grasslands, and so on, which have high intra-class difference or are prone to be affected by seasonal changes and shifts of the instantaneous field of view, our proposed method might fail to provide meaningful co-registering products.

In addition, it is both appropriate and practical to choose the window size considering the misregistration error. For example, if the misregistration error is 5 (w) pixels, the moving window size should be 11 \((2 '* w + 1)\); if the error is 2, the size should be 5. However, in real-world scenarios, visual inspecting or trial-and-error is often the only way to determine the misregistration error for setting the window size.

**About other co-registering methods**

We also considered the availability of other co-registering methods such as SIFT, Template Matching, etc. These methods attempt to locate the local features in an image, commonly known as the “key points” of the image. These points are scale & rotation invariant that can be used for image matching (Bisht et al., 2014). However, sometimes it may be difficult to extract and locate the key points (such as building corners) from two changing images with and without the ground features of interest. As shown in Figure 12, once the extraction of the keypoints (or say corresponding-image-points) fails, the results can be severely distorted, especially when there are fewer corresponding-image-points. That is why these methods were not taken into consideration in this research. On the contrary, our RCVA-based co-registering method is always robust to the changing environments.

Even so, we cannot deny that in many cases, due to the large intra-class spectral difference, one or several pixel pair(s) that has/have the minimum Xdiffb value(s) may not be the key points. However, our proposed algorithm is still useful in this situation because (1): for the reorganization of the time-t2 image occurs in a small-size moving window, the semantic features of the relevant ground objects cannot be destroyed, and the subsequent deep learning analysis will distinguish them; (2): from the macroscopic and probabilistic perspective, relative to other candidate image pairs, the image pairs having the minimum Xdiffb are more likely to be or be spatially close to real key points (Thonfeld et al., 2016), and this is helpful for dynamic image co-registration. These explained why our model performs better in Table 2 and Figure 12.

**About the recall rate of SiUNet3+–CD**

One of the biggest disadvantages of our method is the relatively worse performance of recall. Recall is the measure of how often the actual positive class is predicted as such. Lower recall occurs if many of the positive changes are never predicted, which might be attributed to the less discriminative feature maps. One possible solution for this issue is to alter the probability threshold at which we classify the changed vs. non-changed pixels. In other words, we can reduce the probability threshold, thereby predicting the positive changes more often. Setting a probability threshold at 0.5 in the above experiments means that all the pixels above such value will be rounded to 1, or to 0
Fewer Keypoints:

Moderate Keypoints (found on the building corners):

More Keypoints (found on the building corners, road intersection points, etc.):

Pre-Event Image (Fixed Image)  Post-Event Image with a 5-pixel error (Moving Image)  Template Matching  SIFT  Ours

Figure 12. A comparison of visualization results of different co-registering methods the SIFT and template matching methods are re-implemented based on public code.

otherwise, but it does not necessarily represent the optimal choice if we want to maximize the recall metric. Taking the CDD dataset as example, after changing the probability threshold of the outputting feature map from 0.5 to 0.0 (As in Figure 3, we use the GN layer to normalize the outputting probability values to have a mean zero and standard deviation one, rather than to 0 ~ 1, so it is better to set up the threshold as 0), the recall rate of SiUNet3+-CD in Table 1 increased from 0.870 to 0.877, while the precision and F1-score became 0.948 and 0.911. Although these accuracy values are still inferior to the results derived by SiUNet-CD in Table 1, we provide a new perspective to explain, analyze and understand the RS CD information, and this is the first time that such a network architecture was proposed and applied to high-resolution satellite imagery.

About the two modifications comparing to SiUNet-CD

As mentioned in Section 2.2, we use the subtraction operation in the Siamese model, rather than concatenation, but what will happen if we add a 1x1xN convolution layer after the concatenation operation to reduce the number of feature maps? The experimental result indicated that the precision, recall, F1 score, and IOU (Intersection over Union) derived from the “concatenation + 1x1xN convolution” operation are 0.9401, 0.8695, 0.9034, and 0.8240, respectively, which is slightly lower than the accuracies derived from the subtraction operation. This might be because the 1x1xN convolution does not take into account the contextual information of the previous feature maps.

Another modification based on SiUNet-CD is that our encoder has ten convolution blocks, while SiUNet-CD has eight. The precision, recall, F1 score, and IOU derived from the latter are 0.9307, 0.8142, 0.8685, and 0.7676, respectively, which is much lower than the accuracies derived from the former (see, Table 1). This is because deeper and wider networks can capture richer and more complex features, and generalize well on new tasks (Tan & Le, 2019). However, our model exhibits a relatively poor recall because it failed to detect the small changed objects in many cases, and this can be attributed, to some extent, to the introduction of the additional two convolution blocks \(X_A^5 \) and \(X_B^5\). In the context of the full-scale skip connection, the very-low-resolution feature maps additionally extracted by \(|X_A^5 - X_B^5|\) may hinder the detection of small changed objects.
Other problems

In addition, training the proposed model heavily relies on large labeled datasets. However, it is time-consuming and labor-intensive to collect large-scale bitemporal images that contain various changes, due to both its rarity and sparsity. We have trained our model on a modified BCDD dataset. As mentioned, the image pair of BCDD with a size of 32,507 × 15,354 pixels can be divided into non-overlapping pairs of 256 × 256 pixels, and it is split into 7434 pairs. We delete the non-changed image pairs, and then there are only 1887 pairs left. We trained our model on these residual image pairs, and the precision, recall rate, and F1-score obtained after 70 epochs are only 0.686, 0.774, and 0.725, respectively. Findings showed that when only a small amount of training data available, our model is prone to overfitting or presenting poor performance on CD.

Even worse, we have to train our model from scratch, which is very time-consuming and costly. One possible solution to these problems could be the use of transfer learning (C. Cao et al., 2019). In the follow-up study, we are planning to further modify the proposed architecture and transfer pre-trained parameters to the double-branch encoder of it, so as to provide a good balance between training efficiency/time, sample size and CD accuracy. At least, the trained model file (in *.pt or *.pth format) using the CCD or BCDD dataset could directly provide useful model-initialization parameters for other CD tasks, which is expected to improve the prediction accuracy further.

As mentioned, although our model training is relatively time-consuming, its inference efficiency is acceptable. Pre-processing bi-temporal RGB images of size 256 × 256 takes 0.62s on average, while delineating the changed pixels between them takes 0.28s averagely. In total, it takes about 0.90s for our model to make one prediction.

Conclusion

In this article, we design a densely connected siamese network for CD of high-resolution images, namely Pre-SiUNet3+-CD. Structurally, Pre-SiUNet3+-CD can be regarded as the combination of pre-processing, siamese network, UNet3+, and deep supervision. It is a step forward in designing an effective end-to-end CD technique for multi-temporal high-resolution image analysis.

Using CDD and BCDD dataset as benchmark, a series of comparative experiments were conducted, and our findings showed that: (1) the proposed SiUNet3+-CD model provides very competitive CD accuracies in terms of precision and F1-Score when compared with other state-of-the-art algorithms such as STANet, DASNet, IFN, etc. This is because SiUNet3+-CD inherits most of the advantages of UNet3+ in object location and boundary production. (2) Our RCVA-based pre-processing model was proved to be capable of mitigating, rather than eliminating, the effects introduced by illumination variations and misregistration, thus giving a further accuracy boost of SiUNet3+-CD.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China (under Grant No. 41601465). Helpful review comments and constructive criticism on this manuscript were provided by several anonymous reviewers, and the authors would like to thank them here.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

Bisht, S. S., Gupta, B., & Rahi, P. (2014). Image registration concept and techniques: A review. Int. J. Eng. Res. Appl., 4(4), 30–35. http://ijera.com/papers/Vol4_issue4/Version_207/F44083035.pdf
Cao, C., Dragičević, S., & Li, S. (2019). Land-use change detection with convolutional neural network methods. Environments, 6(2), 1–15. https://doi.org/10.3390/environments6020025
Cao, Z., Wu, M., Yan, R., Zhang, F., & Wan, X. (2020). Detection of small changed regions in remote sensing imagery using convolutional neural network. IOP Conference Series: Earth and Environmental Science, 502(1), 1–11. https://doi.org/10.1088/1755-1315/502/1/012017
Celik, T. (2009). Unsupervised change detection in satellite images using principal component analysis and K-means clustering. IEEE Geoscience and Remote Sensing Letters, 6(4), 772–776. https://doi.org/10.1109/LGRS.2009.2025059
Chen, H., Li, W., & Shi, Z. (2021). Adversarial instance augmentation for building change detection in remote sensing images. IEEE Transactions on Geoscience and Remote Sensing, 60, 1–16. https://doi.org/10.1109/TGRS.2021.3066802
Chen, H., Qi, Z., & Shi, Z. (2022). Remote sensing image change detection with transformers. IEEE Transactions on Geoscience and Remote Sensing, 60, 1–14. https://doi.org/10.1109/TGRS.2021.3095166
Chen, J., Yuan, Z., Peng, J., Chen, L., Li, H., Zhu, J., Liu, Y., & Li, H. (2020). DASNET: Dual attentive fully convolutional siamese networks for change detection of high resolution satellite images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 1194–1206. https://doi.org/10.1109/JSTARS.2020.3037893
Di Pilato, A., Taggio, N., Pompili, A., Iacobellis, M., Di Florio, A., Passarelli, D., & Samarelli, S. (2021). Deep learning approaches to earth observation change detection. Remote Sensing, 13(20), 4083. https://doi.org/10.3390/rs13204083
Peng, S., Li, K., Shao, J., & Li, Z. (2021). SNUNet-CD: A densely connected siamese network for change detection of VHR Images. IEEE Geoscience and Remote Sensing Letters, 19, 1–5. https://doi.org/10.1109/LGRS.2021.3056416

Hao, C., & Shi, Z. (2020). A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. Remote Sensing, 12(10), 1662. https://doi.org/10.3390/rs12101662

Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, T. W., & Wu, J. (2020). UNET 3+: A full-scale connected UNET for medical image segmentation. In ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE. (pp. 1055–1059). Barcelona, Spain.

Huang, F., Yu, Y., & Feng, T. (2019). Automatic building change image quality assessment in high resolution remote sensing based on deep learning. Journal of Visual Communication and Image Representation, 63, 1–10. https://doi.org/10.1016/j.jvcir.2019.102585

Huang, X., Zhu, T., Zhang, L., & Tang, Y. (2014). A novel building change index for automatic building change detection from high-resolution remote sensing imagery. Remote Sensing Letters, 5(8), 713–722. https://doi.org/10.1080/2150704X.2014.963732

Hussain, M., Chen, D., Cheng, A., Wei, H., & Stanley, D. (2013). Change detection from remotely sensed images: From pixel-based to object-based approaches – ScienceDirect. ISPRS Journal of Photogrammetry and Remote Sensing, 80(2), 91–106. https://doi.org/10.1016/j.isprsjprs.2013.03.006

Jiang, F., Gong, M., Zhan, T., & Fan, X. (2019). A semi-supervised GAN-based multiple change detection framework in multi-spectral images. IEEE Geoscience and Remote Sensing Letters, 17(7), 1223–1227. https://doi.org/10.1109/LGRS.2019.2941318

John, H., & Walters, G. (2019). Hands-on generative adversarial networks with PyTorch 1. x: Implement next-generation neural networks to build powerful GAN models using Python. Packt Publishing Ltd.

Li, X., Sun, X., Meng, Y., Liang, J., Wu, F., & Li, J. (2019). Dice loss for data-imbalanced NLP tasks. Cornell University Library. https://doi.org/10.48550/arXiv.1911.02855

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). Venice, Italy.

Liu, Y., Soichi, N., & Tomohisa, Y. (2004). Analysis of four change detection algorithms in bi-temporal space with a case study. International Journal of Remote Sensing, 25 (11), 2121–2139. https://doi.org/10.1080/01431160310001606647

Onur, I., Maktaf, D., Sari, M., & Sönmez, N. K. (2009). Change detection of land cover and land use using remote sensing and GIS: A case study in Kemer, Turkey. International Journal of Remote Sensing, 30(7), 1749–1757. https://doi.org/10.1080/01431160802363966

Peng, D., Bruzzone, L., Zhang, Y., Guan, H., Ding, H., & Huang, X. (2021). SemiCDNet: A semi-supervised convolutional neural network for change detection in high resolution remote-sensing images. IEEE Transactions on Geoscience and Remote Sensing, 59(7), 5891–5906. https://doi.org/10.1109/TGRS.2020.3011913

Peng, D., Zhang, Y., & Guan, H. (2019). End-to-end change detection for high resolution satellite images using improved UNet++. Remote Sensing, 11(11), 1–23. https://doi.org/10.3390/rs11111382

Rodrigo, C. D., Saux, B. L., & Alexandre, B. (2018). Fully convolutional siamese networks for change detection. In 25th IEEE International Conference on Image Processing (ICIP), IEEE. (pp. 4063–4067). Athens, Greece. https://doi.org/10.1109/ICIP.2018.8451652

Shi, W., Zhang, M., Zhang, R., Chen, S., & Zhan, Z. (2020). Change detection based on artificial intelligence: State-of-the-art and challenges. Remote Sensing, 12(10), 1–35. https://doi.org/10.3390/rs12101688

Sun, K., Xiao, B., Liu, D., & Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (pp. 1–12). Long Beach, USA.

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, (pp. 6105–6114). Long Beach, USA. PMLR.

Tang, P., Li, J., Ding, F., Chen, W., & Li, X. (2021). PSNet: Change detection with prototype similarity. The Visual Computer, 1–10. https://doi.org/10.1007/s00371-021-02177-4

Thonfeld, F., Feilhauer, H., Braun, M., & Menz, G. (2016). Robust Change Vector Analysis (RCVA) for multi-sensor very high resolution optical satellite data. International Journal of Applied Earth Observation and Geoinformation, 50, 131–140. https://doi.org/10.1016/j.ijag.2016.03.009

Wang, Q., Yuan, Z., Du, Q., & Li, X. (2019). GETNET: A general end-to-end two-dimensional CNN framework for hyperspectral image change detection. IEEE Transactions on Geoscience and Remote Sensing, 57(1), 3–13. https://doi.org/10.1109/TGRS.2018.2849692

Wang, M., Zhang, H., Sun, W., Li, S., Wang, F., & Yang, G. (2020). A coarse-to-fine deep learning based land use change detection method for high-resolution remote sensing images. Remote Sensing, 12(12), 19–33. https://doi.org/10.3390/rs12121933

Wiratama, W., Lee, J., Park, S. E., & Sim, D. (2018). Dual-dense convolution network for change detection of high-resolution panchromatic imagery. Applied Sciences, 8(10), 1–13. https://doi.org/10.3390/app8101785

Wiratama, W., & Sim, D. (2019). Fusion network for change detection of high-resolution panchromatic imagery. Applied Sciences, 9(7), 1–17. https://doi.org/10.3390/app9071441

Wu, C., Du, B., & Zhang, L. (2014). Slow feature analysis for change detection in multi-spectral imagery. IEEE Transactions on Geoscience and Remote Sensing, 52(5), 2858–2874. https://doi.org/10.1109/TGRS.2013.2266673

Wu, Y., & He, K. (2018). Group normalization. In Proceedings of the European conference on computer vision (ECCV), (pp. 3–19). Munich, Germany. https://doi.org/10.1007/s11263-019-01198-w

Yang, M., Jiao, L., Liu, F., Hou, B., & Yang, S. (2019). Transferred deep learning-based change detection in remote sensing images. IEEE Transactions on Geoscience and Remote Sensing, 57(9), 6960–6973. https://doi.org/10.1109/TGRS.2019.2909781
Yang, K., Xia, G. S., Liu, Z., Du, B., Yang, W., & Pelillo, M. (2021). Asymmetric Siamese Networks for Semantic Change Detection in Aerial Images. IEEE Transactions on Geoscience and Remote Sensing, 60, 1–18. https://doi.org/10.1109/TGRS.2021.3113912

Zhang, W., & Lu, X. (2019). The spectral-spatial joint learning for change detection in multispectral imagery. Remote Sensing, 11(3), 1–17. https://doi.org/10.3390/rs11030240

Zhang, M., Shi, W., Chen, S., Zhan, Z., & Shi, Z. (2020a). Deep multiple instance learning for landslide mapping. IEEE Geoscience and Remote Sensing Letters, 18(10), 1711–1715. https://doi.org/10.1109/LGRS.2020.3007183

Zhang, C., Yue, P., Tapete, D., Jiang, L., Shangguan, B., Huang, L., & Liu, G. (2020b). A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images. ISPRS Journal of Photogrammetry and Remote Sensing, 166, 183–200. https://doi.org/10.1016/j.isprsjprs.2020.06.003

Zheng, Z., Ma, A., Zhang, L., & Zhong, Y. (2021). Change is everywhere: Single-temporal supervised object change detection in remote sensing imagery. In Proceedings of the IEEE/CVF International Conference on Computer Vision, (pp. 15193–15202). Montreal, Canada.

Zheng, Z., Zhong, Y., Wang, J., & Ma, A. (2020). Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (pp. 4096–4105). Seattle, USA.

Zhou, Z., Siddiquee, M., Tajbakhsh, N., & Liang, J. (2018). UNet++: A nested U-Net architecture for medical image segmentation. In Stoyanov, D. et al. (eds). Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. DLMIA 2018, ML-CDS 2018. Lecture Notes in Computer Science, vol 11045. Springer, Cham. https://doi.org/10.1007/978-3-030-00889-5_1