IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 May 2023, accepted 26 June 2023, date of publication 5 July 2023, date of current version 12 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3292531

RESEARCH ARTICLE

Change Detection of High-Resolution Remote Sensing Images Through Adaptive Focal Modulation on Hierarchical Feature Maps

LHUQITA FAZRY, MGS M. LUTHFI RAMADHAN, AND WISNU JATMIKO, (Senior Member, IEEE)
Faculty of Computer Science, University of Indonesia, Depok City 16424, Indonesia
Corresponding author: Lhuqita Fazry (lhuqita.fazry@ui.ac.id)

This work was supported in part by the Tokopedia-Universitas Indonesia Artificial Intelligence (UI AI) Center of Excellence (NVIDIA DGX-1 computing facilities), and in part by the Publikasi Terindeks Internasional (PUTI) Q1 from Universitas Indonesia under Project NKB-297/UN2.RST/HKP.05.00/2023.

ABSTRACT One of the major challenges in the change detection (CD) of high-resolution remote sensing images is the high requirement for computational resources. Besides, to get the best change detection result, it must spot only the important changes while omitting unimportant ones, which requires learning complex interactions between multi-scale objects on the images. Despite Convolution Neural Network (CNN) efficiently extracting features from such images, it has a limited receptive field resulting in sub-optimal representation. On the other hand, Vision Transformer (ViT) can capture long-range dependencies. Still, it suffers from quadratic complexity concerning the number of image patches, especially for high-resolution images. Furthermore, both approach can not model the interactions among multi-scale image patches, which is essential for a model to fully understand the natural images. We propose FocalCD, a CD method based on a recently proposed focal modulation architecture capable of learning short and long dependencies to solve this problem. It is attention-free and does not suffer from quadratic complexity. Also, it supports learning multi-scale interaction by adaptively selecting the discriminator regions from multi-scale levels. Besides the efficient yet powerful encoder, FocalCD has an effective multi-scale feature fusion and pyramidal decoder network. FocalCD achieves strong empirical results on various CD datasets, including CDD, LEVIR-CD, and WHU-CD. It reaches F1 scores of 0.9851, 0.952, and 0.9616 on datasets CDD, LEVIR-CD, and WHU-CD outperforming state-of-the-art CD methods while having comparable or even lower computation complexity.

INDEX TERMS Change detection, FocalCD, focal modulation, multi-scale feature fusion, vision transformer.

I. INTRODUCTION

Change Detection (CD) is a remote sensing task for identifying changes from two satellite images taken at different times [1], [2]. The images consist of pre-change and post-change, which refer to the image before and after the change. Both images must be registered spatially before the detection process takes place. This pre-processing step is carried out to ensure those images cover the same area and from the same perspective, including the position, rotation, and width.

A typical CD method processes the co-registered bi-temporal remote sensing images and returns a binary-change map. This map is a binary image containing only two-pixel values representing the changed and unchanged area. Other properties of the change map are it has identical spatial resolution with pre-change and post-change images and covers the same place. Technically, a CD can be seen as a segmentation task because its goal is to separate the image into two parts: changed and unchanged areas. However, unlike regular segmentation tasks that segment a single image based on object instances or semantic meaning, CD needs two input images to segment the changed areas.

The associate editor coordinating the review of this manuscript and approving it for publication was Gerardo Di Martino.

FIGURE 1. Illustration of fundamental changes and pseudo changes due to various conditions. From left to right: seasonal changes, vegetation differences, and variational object shadows. Green and orange rectangles show the fundamental and pseudo changes, respectively.

The CD is a crucial task widely adopted in various applications, such as urban planning [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], disaster management [5], [13], [14], [15], [16], [17], and environmental monitoring [18], [19], [20]. These applications must capture the changes quickly and accurately to make proper decisions. The low-cost operation also becomes an important consideration that hinders manual detection from being selected as an appropriate method. Manual change detection by humans is expensive, prolonged, and error-prone. The wider area, represented by higher resolution on the remote sensing images, makes these restrictions more difficult to occupy. To meets these strict requirements, an automatic CD method is needed.

Despite its importance, CD is challenging due to the complex and diverse image variations, including seasonal changes, vegetation differences, and variational object shadows. Fig. 1 illustrates the variations between pre-change and post-change images based on various conditions. In the left image, snows create pseudo changes, as portrayed in the orange rectangle, while those changes are ignored in the ground truth. In the middle images, different shape of the same vegetation creates a pseudo-change which is also ignored in the ground truth image. In the right images, the pseudo changes created are due to the shadow of the vegetation. These variations contribute to forming unintended differences between pre-change and post-change images. Those differences can be considered as noises and can cause a significant reduction in the detection accuracy. Not all changes are essential and relevant to the task. To support this idea, we define two terms: fundamental changes and pseudo changes. Fundamental changes are the changes that need to be detected, while pseudo changes are the ones which irrelevant to the task and should be ignored. A suitable CD method should be able to distinguish between fundamental changes and pseudo changes. We need a way to extract salient features from each input image while suppressing the noises and redundant information to achieve this goal. Technically, it should focus on selected image regions which are the most discriminative representation of the changes. The method should address local and global interactions between image regions to improve the representation power. It is worth mentioning that the interaction at a multi-scale level can also improve the representation, as has been shown in [21]. The multi-scale interaction is motivated by the fact that natural images, including remote sensing images, can contain many objects at different scales. So this particular type of interaction is essential to understand the images entirely.

There have been many previous works on the development of the CD method. The advance of deep learning techniques also pushes forward innovations in this field. Convolution Neural Network (CNN) is one of the most used architectures for extracting image features in the CD method. The main reason is that CNN is designed to process image data type, which matches the CD input data type. Another reason is it's efficient because it utilizes parameter sharing to learn the features of the whole image. Typically, features extracted from pre-change and post-change images are combined and fed into the pixel classifier to build the change map. Despite its efficiency, the receptive field of CNN is limited, hindering it from capturing long-range dependencies between image regions. This limitation makes the CD method sub-optimal at learning image features, especially for high-resolution input like remote sensing images.

Recent advances in Transformer architecture on Natural Language Processing (NLP) are quickly adopted by a broad range of scientific fields, including remote sensing. Transformer was initially developed to process text data. However, recent works proved it could also be used to process image data. Dosovitskiy et al. demonstrated this possibility by treating 16×16 image patches as visual tokens and then fed the embedding of the visual token into the vanilla Transformer [22]. Surprisingly, it achieves state-of-the-art accuracy in ImageNet classification. This result confirmed the Transformer architecture's ability to model long-range dependencies between image patches, making it better at extracting image features than CNN.

This variant of the Transformer emerges as a promising vision backbone, dubbed Vision Transformer (ViT), replacing the CNN after over two decades has been a standard vision model. Despite its success, it can not be directly adopted to fine-grained image tasks such as segmentation and object detection. It is because those tasks need a smaller patch size to perform a classification at a pixel level. A smaller patch size will increase the number of visual tokens. However, the self-attention (SA) mechanism, which is the critical component of the Transformer architecture, has quadratic computation complexity with respect to the number of visual

tokens, making it impossible to process smaller patch sizes. SA consists of two primary operations: context calculation and context aggregation. Both are expensive operations due to the need for token-to-token calculations. Many works have been proposed to address the computation complexity of SA through constrained attention [23], [24], [25], token down-sampling [26], dynamic token [27], [28], or hybrid approaches [29], [30], [31]. These variants of SA enable the adoption of ViT for fine-grained tasks. There were some works on the development of CD method utilizing ViT such as PSTNet [32], BIT-CD [33], ChangeFormer [34], UVACD [35], Hybrid-TransCD [36], SiamixFormer [8], SwinSUNet [11], and TransUNetCD [37].

The aforementioned CD methods can spot fundamental changes by focusing on the discriminative regions on the images, thanks to the SA mechanism. However, none of these methods pay attention to the interactions between regions at multi-scale levels. As mentioned above, this condition may constrain their performance as multi-scale interactions are essential for natural images. Inspired by [30], we propose a novel CD method, dubbed FocalCD, to solve this problem. FocalCD can capture local, global, and multi-scale interactions by adaptively selecting the discriminative regions from multi-scale levels. Furthermore, it is attention-free and doesn't suffer from quadratic computation complexity like SA.

The main component of FocalCD is the Focal Modulation (FM). The functionality of FM is to refine the representation of feature maps by modulating the features in hierarchical settings that are adaptively selected based on their focal level. This modulation strategy guarantees that the resulting refined feature maps attend local and global interaction from multi-scale levels. FocalCD consists of 3 main modules: the encoder, multi-scale feature fusion (MSFF), and the decoder. The encoder is a two-stream network because it needs to process two input images. The encoder network incorporates the FM inside it. MSFF is responsible for fusing feature maps from the two-stream encoder. Each feature map is combined with another feature map at the corresponding scale. The fusion process forces the network to learn the discriminative features between two input features. The decoder then reconstructs the combined features into the binary change map.

Finally, it is worth to mention the contribution of our work. Here, we summarize our contribution:

1) To overcome the high computational cost of detecting change on high-resolution remote sensing images, we propose a novel CD method by leveraging an efficient focal modulation capable of modeling short and long interactions between visual tokens. Furthermore, we append the network with multi-scale feature fusion and a pyramidal decoder network to support the multi-scale feature interaction imposed by the encoder.
2) We relate the ability of the CD method to distinguish between fundamental changes and pseudo changes with the method's capability to model multi-scale object interactions. Based on this observation, we use aggregated hierarchical contexts to support learning multi-scale interaction by adaptively selecting the discriminator regions from multi-scale levels.
3) Our proposed method achieved solid empirical results on various CD datasets, including CDD, LEVIR-CD, and WHU-CD, resulting in a new state-of-the-art CD method.

The rest of this paper is organized as follows: Section II presents existing works on CD method development. Section III describes our proposed method. Section IV describes the experiments, including the experimental setup. Section V provides the result and discussion. Section VI concludes our work.

II. RELATED WORK

Despite its challenges, CD is one of the most popular remote sensing tasks that attracts much attention. Many exciting methods have been proposed to solve it. This section briefly explains some notable works in this field and describes the main differences between our proposed method and the existing works.

A. CD METHODS ON HIGH-RESOLUTION REMOTE SENSING IMAGES

Remote sensing images are known to have a high resolution both on channel and spatial dimensions. A common problem when dealing with high-resolution images is the high computational and memory that limits the number of parameters of the model's architecture. This limitation can bring a significant reduction to the model's performance. There are existing methods that focus on solving CD of high-resolution remote sensing images, such as UNetCD [38], IFN [39], SCDNet [40], and SNUNet [10].

UNetCD [38] used an efficient UNet++ [41] backbone as a feature extractor composed of a stack of convolutions using 3×3 kernel. To save the network's parameters further, UNetCD incorporates an early fusion of the input images. IFN [39] proposed a novel feature fusion consisting of efficient channel-wise attention followed by spatial attention to mixing information from both channel and spatial dimensions. These efficient attentions incorporate average and max pooling strategies to reduce the feature map dimensions. On the other hand, SCDNet [40] proposed a lightweight attention module to combine the feature maps returning from the encoder. This attention is an improved version of the attention module used in IFN. This attention module first creates two reduced versions of feature maps and then merges them through element-wise addition. SCDNet also used an efficient atrous convolution to enlarge the receptive field of the encoder while keeping the network's parameters small. Unlike IFN and SCDNet, SNUNet-CD [10] proposed ECAM (Ensemble Channel Attention Module), a natural expansion of the attention module used in IFN. This attention module inherits the efficiency of IFN's attention module while having

better performance and a smaller number of the models' parameters.

On the other hand, our proposed method uses an attention-free architecture composed of efficient depth-wise convolutions. Depth-wise convolution uses smaller parameters than regular convolution, which is more efficient. We use a stack of depth-wise convolutions to increase the receptive field. Also, we add additional global context to expand the receptive field further.

B. CNN-BASED CD METHODS

As a standard architecture for processing image data, CNN has been extensively explored to develop CD methods. FC-EF, FC-Siam-Conc, and FC-Siam-Diff [42] are some early adopters using CNN to solve the CD task. These methods comprise stacks of convolution and max pooling operations that create a bottleneck that return a compact representation of the input images. The architecture is then followed by upsampling operations to generate the change map from the intermediate features. Instead of a stack of convolution and pooling combinations, Ji et al. [5] proposed the CD method by utilizing UNet architecture [43]. This better vision backbone has been proved empirically to have a high-performance feature extraction capability. FDCNN [44] expands the use of CNN in CD by designing a feature difference network consisting of a CNN backbone. This network is forced to learn the difference between the features of two input images. A feature fusion network then appends the network to enhance the performance of the backbone network.

Instead of simply utilizing a standalone CNN backbone as a feature extractor which may be suboptimal for solving the CD tasks, DASNet [9] took a different approach. It combined the power of CNN with an attention mechanism suitable for CD tasks. Specifically, it comes with a unique spatial and channel attention combination, enhanced using the novel weighted double-margin contrastive (WDMC) loss. However, DSANet [45] took another strategy. DSANet improved the DASNet further by replacing the channel attention using atrous convolution and atrous spatial pyramid pooling (ASPP) combination while keeping the spatial attention module.

Unlike existing approaches, our proposed method uses the recently proposed focal modulation architecture to extract bi-temporal input. This architecture aims to refine feature maps by modulating them using adaptively aggregated multi-scale contexts. These multi-scale contexts provide rich information to refine the feature maps.

C. TRANSFORMER-BASED CD METHODS

Despite being relatively new, Vision Transformer (ViT) is widely used for solving vision tasks, including fine-grained tasks such as image segmentation. In the field of CD, there are some existing works that ViT influences. Some of the works are PSTNet [32], Hybrid-TransCD [36], TransUNetCD [37], SiamixFormer [8], and SwinSUNet [11].

PSTNet [32] is a CD method that uses a hybrid approach. Instead of directly treating image patches as tokens, it uses the feature maps produced by a CNN backbone. PSTNet consists of ResNet as a feature extractor followed by a progressive sampling transformer (PST) module to make the token from the feature maps in a progressive manner. Similar to PSTNet, Hybrid-TransCD [36] also employs a CNN backbone to create hierarchical feature maps before the transformer processes. However, unlike PSTNet, which uses vanilla ViT, Hybrid-TransCD uses a mix of pyramid vision transformer (PVT) [46], and Swin Transformer [23]. Furthermore, Hybrid-TransCD only uses the latest stage of the feature maps representing the global context to be fed into the Transformer. Like PSTNet and Hybrid-TransCD, TransUNetCD [37] also uses a hybrid approach. However, TransUNetCD employs a slightly different approach. TransUNetCD has a U-shaped consisting of stacks of convolution like vanilla UNet [43]. However, it attaches a Swin Transformer in the bottleneck stage to further increase the feature maps' representation power.

Unlike previous hybrid methods, SiamixFormer [8] and SwinSUNet [11] use the Transformer architecture without being preceded by CNN. However, there are two significant differences between SiamixFormer and SwinSUNet. SiamixFormer used the Transformer only in the encoder and used a lightweight Multi-layer Perceptron (MLP) for the decoder. In contrast, SwinSUNet used the Transformer both in the encoder and decoder. Another difference is SiamixFormer used SegFormer [47] as the backbone, while SwinSUNet used Swin Transformer [23] architecture.

On the other hand, our proposed method is not directly using the Transformer. However, it uses the focal modulation architecture inspired by the Transformer, which can model short and long interactions within images. Compared to SA, focal modulation is much more efficient because it adopts the pre-calculated context aggregation mechanism, which prevents it from attending token-to-token interaction. Despite its efficiency, FocalCD performs comparable or even better than SwinSUNet and SiamixFormer.

III. PROPOSED METHOD

This section presents detailed information about our proposed method, FocalCD.

A. ARCHITECTURE OVERVIEW

FocalCD accepts a bi-temporal of high-resolution remote sensing images as input. The images consist of pre-change and post-change in the same spatial area while being taken at different times. Given pair of images $\mathbf{X}_{\text{pre}}, \mathbf{X}_{\text{post}} \in \mathbb{R}^{H \times W \times 3}$, FocalCD processes them and produce the binary change map $\mathbf{M} \in \mathbb{R}^{H \times W}$ where H, W, and 3 denote the height, width, and channel of the input images respectively. The map $\mathbf{M}$ represents the changed and unchanged area between the images. The changed area is depicted by white pixels, whereas the unchanged area is by black pixels.

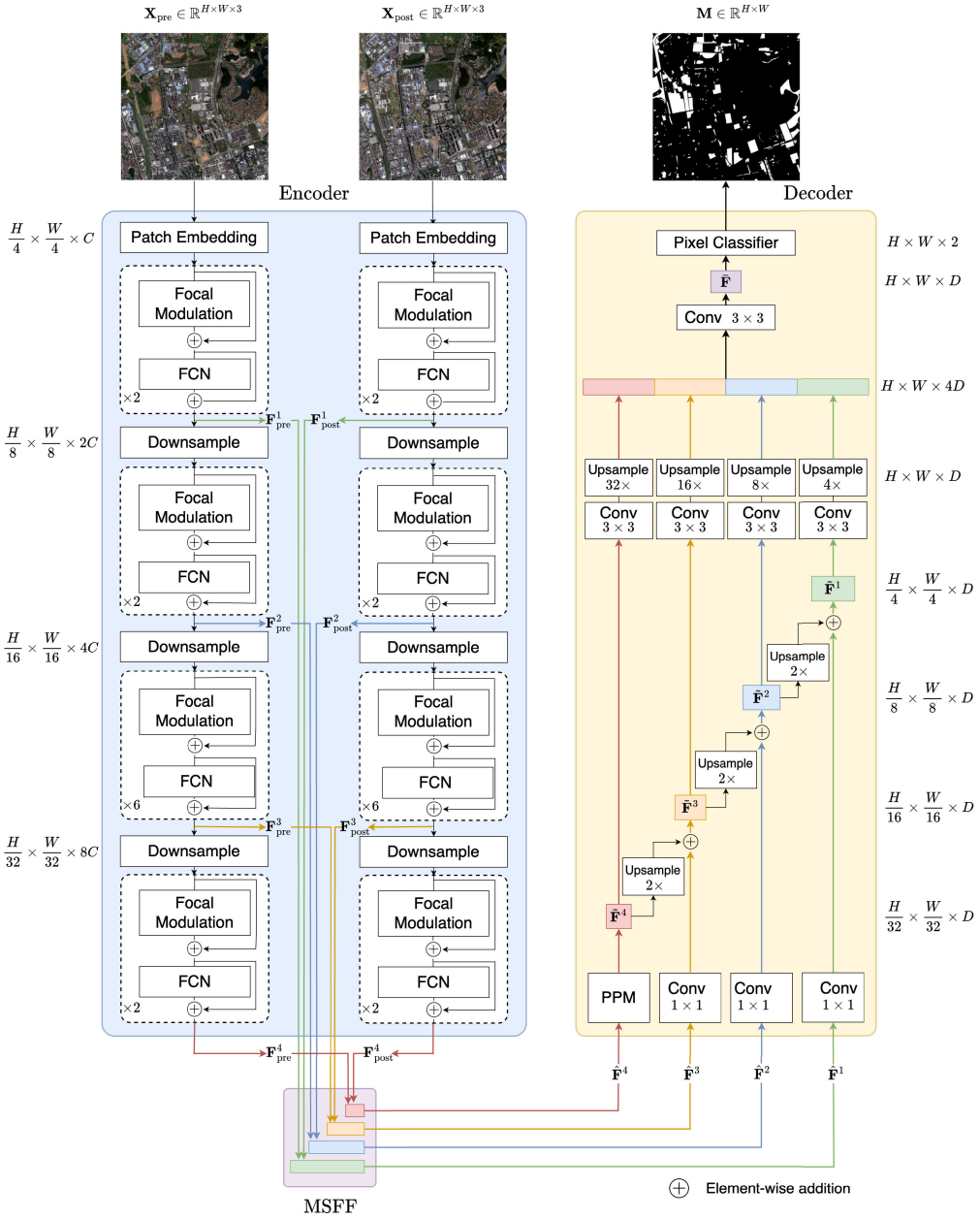

FIGURE 2. The overall architecture of FocalCD. It consists of three main modules: the encoder, multi-scale feature fusion (MSFF), and the decoder.

FocalCD consists of 3 main modules: encoder, multi-scale feature fusion, and decoder. The encoder is responsible for extracting multi-scale salient features from pair of input images. The outcome of the encoder is multi-scale features: $(\mathbf{F}^1_{pre}, \mathbf{F}^2_{pre}, \mathbf{F}^3_{pre}, \mathbf{F}^4_{pre})$ and $(\mathbf{F}^1_{post}, \mathbf{F}^2_{post}, \mathbf{F}^3_{post}, \mathbf{F}^4_{post})$ for $\mathbf{X}_{pre}$ and $\mathbf{X}_{post}$ respectively. The multi-scale feature fusion then combines the extracted features of each image at the corresponding scale into $\hat{\mathbf{F}}^1, \hat{\mathbf{F}}^2, \hat{\mathbf{F}}^3$ and $\hat{\mathbf{F}}^4$. The decoder transforms the combined features into a binary change map. Fig. 2 shows the general architecture of FocalCD.

B. FOCAL MODULATION

SA [22], [48] can model global and local interactions between visual tokens [29], which are essential to identify unique features of natural images. Nevertheless, covering both interactions on high-resolution images like remote sensing images suffers from quadratic computational cost with respect to the number of visual tokens. On the other hand, CNN has efficient computation due to using a shared kernel but has a narrow receptive field. Inspired by the success of [30], we introduce focal modulation (FM) architecture utilization for solving CD tasks. It is an efficient and attention-free

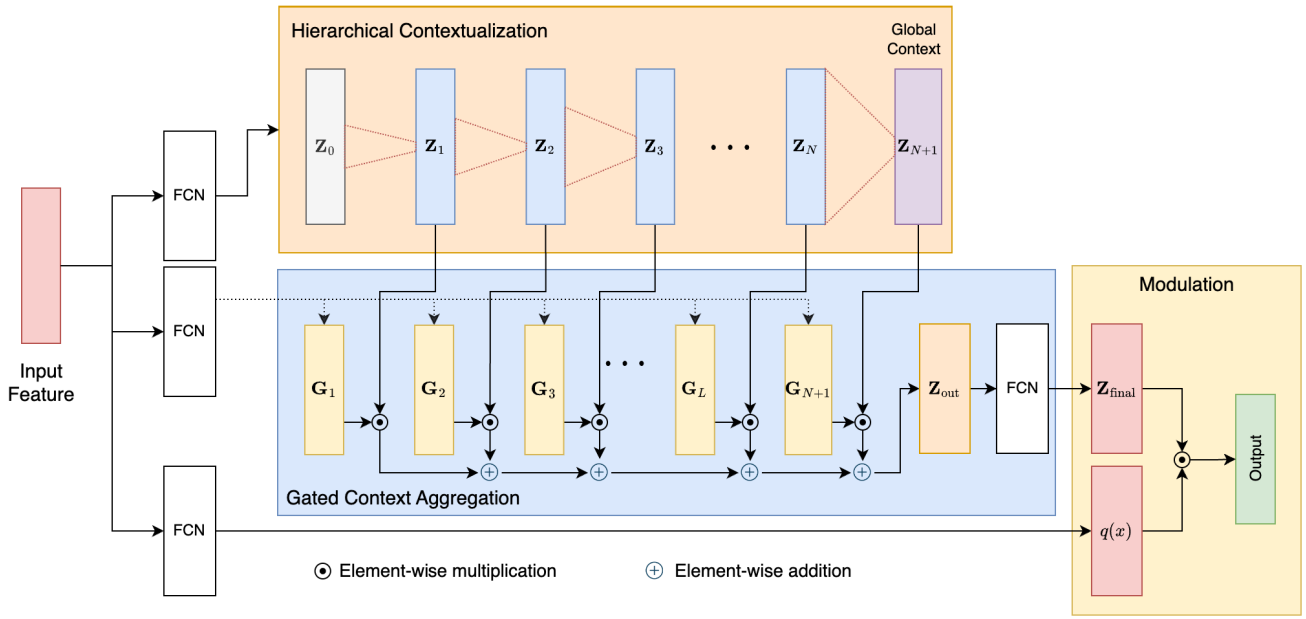

FIGURE 3. Focal Modulation (FM) consists of 3 steps. First, hierarchical contextualization for extracting context features at hierarchical levels. Second, gated context aggregation for adaptive combining of the hierarchical contexts. Third, query modulation using the aggregated contexts.

architecture that covers both interactions to refine the visual features extracted from high-resolution images. FM is a hybrid approach that incorporates the capability of SA and the efficiency of CNN. We use the FM as a component to assemble the encoder of our FocalCD method.

SA calculates the context scores before the context aggregation to produce an output. This scenario requires a query-dependent operation which leads to heavy operations such as token-to-token calculation. FM takes a different approach by pre-calculate context aggregation focally around each query which is query-independent. This approach has an advantage, i.e., it can be calculated efficiently using a shared kernel like CNN or even a more parameter-efficient alternative like depth-wise convolution [49], [50]. Furthermore, it can also calculate the context aggregation at different granularity levels by utilizing a larger kernel size to improve the representation power further.

Given a visual token $\mathbf{x}_i \in \mathbb{Z}^{H\times W\times d}$ as query, FM can be formulated as

$$\mathbf{y}_i = q(\mathbf{x}_i) \odot \alpha(i, \mathbf{Z}), \tag{1}$$

where $\mathbb{Z}$ is a feature map and $\mathbf{y}_i$ is a refined representation of $\mathbf{x}_i$. Equation (1) is a Hadamard product between a query projection function $q(\mathbf{x}_i)$ and a context aggregation function $\alpha(i, \mathbf{Z})$. The function $q(\mathbf{x}_i)$ preserves the finest information for each visual token while $\alpha(\cdot)$ extracts the coarser context. Thus (1) combines fine and coarse information. The query projection is simply a linear layer that projects the feature dimension of $\mathbf{x}_i$ from d into an arbitrary number C. The function $\alpha(\cdot)$ is a modulator that modulates the context of feature maps $\mathbf{Z}$ at each location i for some pre-defined scales.

FM formulation in (1) can be split into three steps: hierarchical contextualization of feature maps, gated context aggregation, and modulation. The first two steps elucidate the context aggregation function $\alpha(\cdot)$. The hierarchical contextualization step is responsible for extracting contexts at different levels of granularity by incorporating multi-scale feature maps. Each piece of information from the hierarchical context will go through the gated entrance to be modulated in the modulation step. Fig. 3 illustrates the process inside the focal modulation module.

1) HIERARCHICAL CONTEXTUALIZATION OF FEATURE MAPS

In this step, we extract a coarse context representation at multi-level granularity. We construct hierarchical contextualization by stacking N depth-wise convolutions to achieve this. Given a feature map $\mathbf{X}$, we first project it using a linear transformation into $\mathbf{Z}_0 = f^Z(\mathbf{X}) \in \mathbb{R}^{H\times W\times C}$. This transformation can help enhance the input feature's discriminative power and capture relevant patterns or correlations between different input features. A contextualization function $f_n^c(\cdot)$ is then applied to the input feature map at each level $n \in \{1, \ldots, N\}$. The output of the contextualization function is defined as:

$$\mathbf{Z}_n = f_n^c(\mathbf{Z}_{n-1}) := \text{GeLU}(\text{DWConv}(\mathbf{Z}_{n-1})), \tag{2}$$

where $\mathbf{Z}_n \in \mathbb{R}^{H\times W\times C}$. The contextualization output $\mathbf{Z}_n$ is obtained by applying function $f_n^c(\cdot)$ at level n, which is implemented through depth-wise convolution (DWConv) over previous output $\mathbf{Z}_{n-1}$ using the kernel size k_n followed by a GeLU (Gaussian Error Linear Units) non-linearity activation [51]. The reason to use depth-wise convolution is that it has a lower computational cost than standard convolution

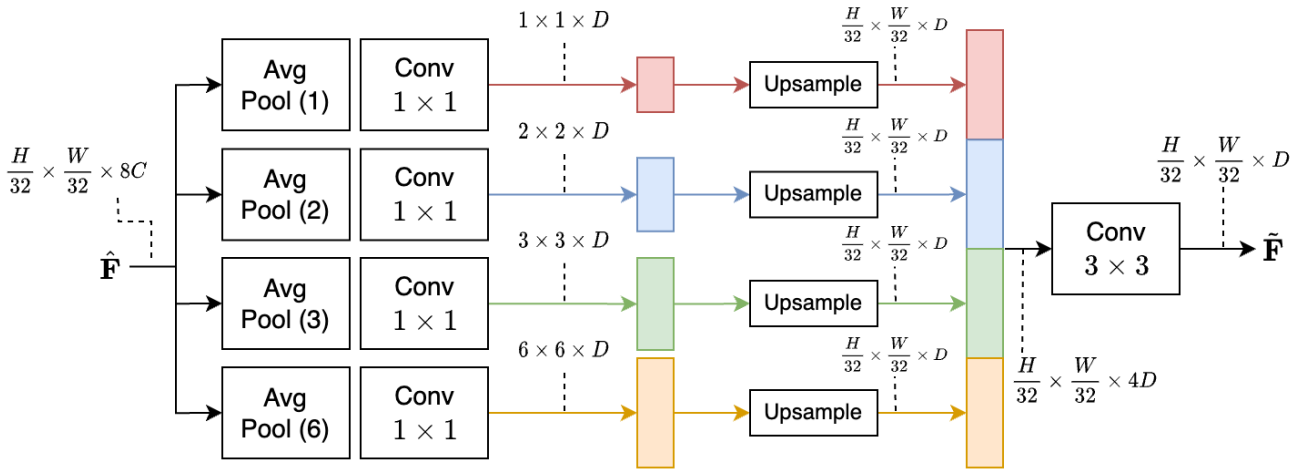

FIGURE 4. PPM provides global-context prior by creating four feature maps of different scales. The maps are then concatenated to produce a single output. The PPM is only applied to the feature maps of the encoder's last stage.

as the process is performed channel-wise. We use GeLU activation following the prior art [52]. Furthermore, GeLU performs better than the well-known ReLU (Rectified Linear Unit) activation due to a non-monotonicity property that allows it to model complex interactions between input features.

The hierarchical structure is accomplished by employing $k_n < k_{n+1}$. Under such a setting, the receptive field of the contextualization function at level n is

$$r_n = 1 + \Sigma_{i=1}^{n}(k_n - 1). \tag{3}$$

This receptive field is larger than the kernel size k_n itself, specifically for $n > 1$.

There are N contexts obtained in total from the above procedure. However, they may not cover the global context of the whole input feature, especially for a small number of N. We add another context level to solve this problem by applying global average pooling to the last context $\mathbf{Z}_N$, i.e., $\mathbf{Z}_{N+1} = \text{AvgPool}(\mathbf{Z}_N)$. Therefore, we have $N + 1$ feature maps that represent short and long contexts at various levels of granularity.

2) GATED CONTEXT AGGREGATION

Recall that the primary goal of the CD method is to distinguish between fundamental and pseudo changes. This implies that some changes are relevant and others are not. We associate this condition with relative dependencies of an object in an image to other parts. A visual token (query) may highly depend on the fine-grained local features in natural images, especially for remote sensing images. However, it can also hold a small portion of dependencies on the global coarse-grained features. To cover these circumstances, we incorporate a gating mechanism to take control of how much information to aggregate from various levels of contexts for each query.

We enforce this relative dependency by adding adaptive weight to each gating entrance. This adaptive weighting serves the same functionality as the attention score in SA. However, it's free from attending token-to-token interactions, so it doesn't suffer from quadratic complexity like in SA. To increase flexibility, we enable per-token weight. Specifically, we implement this adaptive weight through learnable parameters by incorporating a linear layer. Given $N + 1$ context from prior step, $\mathbf{Z} = \{\mathbf{Z}\}_{n=1}^{N+1}$, we define the weight-scoring function as $\mathbf{G} = f^g(\mathbf{Z}) \in \mathbb{R}^{H \times W \times (N+1)}$. Identical to SA, the final output $\mathbf{Z}_{out}$ is obtained by performing a sum to the weighted output acquired through an element-wise multiplication operation. We choose the element-wise multiplication over other operations to align with its role as a weighting score. Formally, it is defined as

$$\mathbf{Z}_{\text{out}} = \sum_{n=1}^{N+1} \mathbf{G}_n \odot \mathbf{Z}_n, \tag{4}$$

where $\mathbf{Z}_{out} \in \mathbb{R}^{H \times W \times C}$ identical to the individual dimension of the context $\mathbf{Z}_n$. Note that $\mathbf{G}_n$ is a slice of $\mathbf{G}$ at level n having dimension $H \times W \times 1$. We can easily observe that the sum operation in (4) is performed element-wise. This imposes the spatial correlation between various levels of context. However, it does not support the correlation between channels, whereas it is required to strengthen the representation power of the final context. To tackle this problem, we project it by utilizing the linear layer to impose a correlation between channels. Therefore, the final output can be written as

$$\mathbf{Z}_{\text{final}} = f^h(\mathbf{Z}_{\text{out}}). \tag{5}$$

3) MODULATION

The steps described so far are part of the context aggregation function $\alpha(\cdot)$. One step left to complete FM formulation, as depicted in (1), is focal modulation. This final step modulates the aggregated context into an individual projected

query $q(\cdot)$ through an element-wise multiplication. Formally it can be written as

$$\mathbf{y}_i = q(\mathbf{x}_i) \odot f^h(\sum_{n=1}^{N+1} \mathbf{g_n^i} \cdot \mathbf{z}_n^i) \tag{6}$$

where $\mathbf{g_n^i}$ and $\mathbf{z}_n^i$ represent the individual gating value and visual feature at location i. Algorithm 1 summarizes the FM process.

Algorithm 1 Focal Modulation

Require: Input feature maps $\mathbf{X} \in \mathbb{Z}^{B\times H\times W\times d}$
Ensure: Refined feature maps $\mathbf{Z} \in \mathbb{Z}^{B\times H\times W\times d}$
$\mathbf{Q} \leftarrow q(\mathbf{X})$
$\mathbf{Z} \leftarrow f^Z(\mathbf{X})$
$\mathbf{G} \leftarrow f^g(\mathbf{X})$
for $n = 1, \cdots, N$ **do**
  $\mathbf{Z} = \text{GELU}(\text{DWConv}(\mathbf{Z}))$
  $\mathbf{A} = \mathbf{A} + \mathbf{G}[n] * \mathbf{Z}$
end for
$\mathbf{Z} = \mathbf{A} + \text{GELU}(\text{DWConv}(\text{AvgPool}(\mathbf{Z}))) * \mathbf{G}[N+1]$
$\mathbf{Z} = \mathbf{Q} * f^h(\mathbf{Z})$

C. ENCODER

The encoder is a two-stream network with parameter sharing. It accepts bi-temporal remote sensing images containing pre-change and post-change images, denoted by $\mathbf{X}_{\text{pre}}$ and $\mathbf{X}_{\text{post}}$ respectively where a different stream of the network will process the pre-change and post-image separately. The encoder is constructed in 4 stages. Each stage takes the output feature map from the previous stage and returns a refined representation of the feature except for stage 1. Instead of receiving feature maps, stage 1 gets the whole input image. Each stage $i \in \{1, 2, 3, 4\}$ comprises a stack of M_i FMs followed by two consecutive feed-forward networks (FFN) with `GELU` non-linearity between them. We apply layer normalization [53] before FMs and FFN to stabilize the training. We create a skip connection between FM and these FFN, following [22], [48]. These FFNs induce the non-linearity into the model to help increase its expressive power, enabling it to model complex relationships between the input and output sequences. Furthermore, to avoid the gradient vanishing problem, we incorporate stochastic depth both on FM and FFN by applying a drop path strategy [54].

The encoder employs a hierarchical structure where each stage receives an input feature map with a spatial dimension that shrinks in half as the stage goes deeper. On the other hand, its channel dimension doubled. Specifically, the dimensions of the feature maps are $\frac{H}{4} \times \frac{W}{4} \times C$, $\frac{H}{8} \times \frac{W}{8} \times 2C$, $\frac{H}{16} \times \frac{W}{16} \times 4C$, and $\frac{H}{32} \times \frac{W}{32} \times 8C$ for stage 1, 2, 3, and 4 respectively. This hierarchical structure has been proven empirically effective in the recognition and downstream tasks [26], [55]. To achieve this hierarchical structure, we add a patch embedding before the first stage and downsample layer before layers 2, 3, and 4.

1) PATCH EMBEDDING

This layer is responsible for projecting an input image into a new feature space. Given an input image $\mathbf{X} \in \mathbb{R}^{H\times W\times 3}$, the patch embedding layer first split it into non-overlapping image patches with size $4 \times 4 \times 3$, resulting $\frac{H}{4} \times \frac{W}{4} \times 3$ patches in total. Each patch serves as a visual token that will be fed into the encoder. Each visual token is then projected into an embedding space in an arbitrary channel dimension C, which is commonly much larger than the channel dimension of the original token, to increase its representation capacity. Following [22], [56], we implement this projection using CNN with kernel size and stride equal to the patch size.

2) DOWNSAMPLE LAYER

This layer is added before stages 2, 3, and 4. It does two things: reduce the spatial dimension of each visual token in half and then double the channel dimension to increase its representation capacity further. Like the patch embedding layer, we also utilize CNN to implement the spatial reduction and channel expansion with kernel size and stride equal to the patch size. However, unlike the patch embedding layer, we use patch size 2 instead of 4 to generate fine-grained features.

D. MULTI-SCALE FEATURE FUSION (MSFF)

Multi-scale Feature Fusion (MSFF) combines the multi-scale feature maps produced by a two-streams encoder at the corresponding stage-i into a single feature map. Given feature maps produced by the encoder $\mathbf{F}_{\text{pre}} = \{\mathbf{F}_{\text{pre}}^i\}_{i=1}^4$ and $\mathbf{F}_{\text{post}} = \{\mathbf{F}_{\text{post}}^i\}_{i=1}^4$ where $\mathbf{F}_{\text{pre}}^i$ and $\mathbf{F}_{\text{post}}^i$ denote a feature map extracted at stage-i from input images $\mathbf{X}_{\text{pre}}$ and $\mathbf{X}_{\text{post}}$ respectively. We use element-wise L_p distance for the feature map fusion. The MSFF is defined formally as

$$\hat{\mathbf{F}}^i = |\mathbf{F}_{\text{pre}}^i - \mathbf{F}_{\text{post}}^i| \tag{7}$$

$$= \{|\mathbf{z}_{\text{pre}}^{i,1} - \mathbf{z}_{\text{post}}^{i,1}|, \ldots, |\mathbf{z}_{\text{pre}}^{i,k} - \mathbf{z}_{\text{post}}^{i,k}|, \ldots, |\mathbf{z}_{\text{pre}}^{i,K} - \mathbf{z}_{\text{post}}^{i,K}|\}, \tag{8}$$

where $K \in \mathbb{N}$ is the number of visual tokens and $\mathbf{z}_{\text{pre}}^{i,k}, \mathbf{z}_{\text{post}}^{i,k} \in \mathbb{R}^{h\times w\times c}$ represent the k-th individual visual token produced by the encoder from the input images $\mathbf{X}_{\text{pre}}$ and $\mathbf{X}_{\text{post}}$ at specified stage-i.

E. DECODER

The decoder is responsible for constructing a binary change map from multi-scale fused features returned by MSFF where the change map has the exact spatial resolution as $\mathbf{X}_{\text{pre}}$ and $\mathbf{X}_{\text{post}}$. A simple approach to building such a map is to upsample all the feature maps into the exact spatial resolution as $\mathbf{X}_{\text{pre}}$ and $\mathbf{X}_{\text{post}}$ then concatenate them channel-wise followed by a pixel classifier. However, this approach does not blend spatial information across different scales of the feature maps. Whereas it is needed to support the multi-scale feature interaction imposed by FM in the encoder. Despite the individual scale of the feature maps having a strong semantic representation, merging the coarser features into

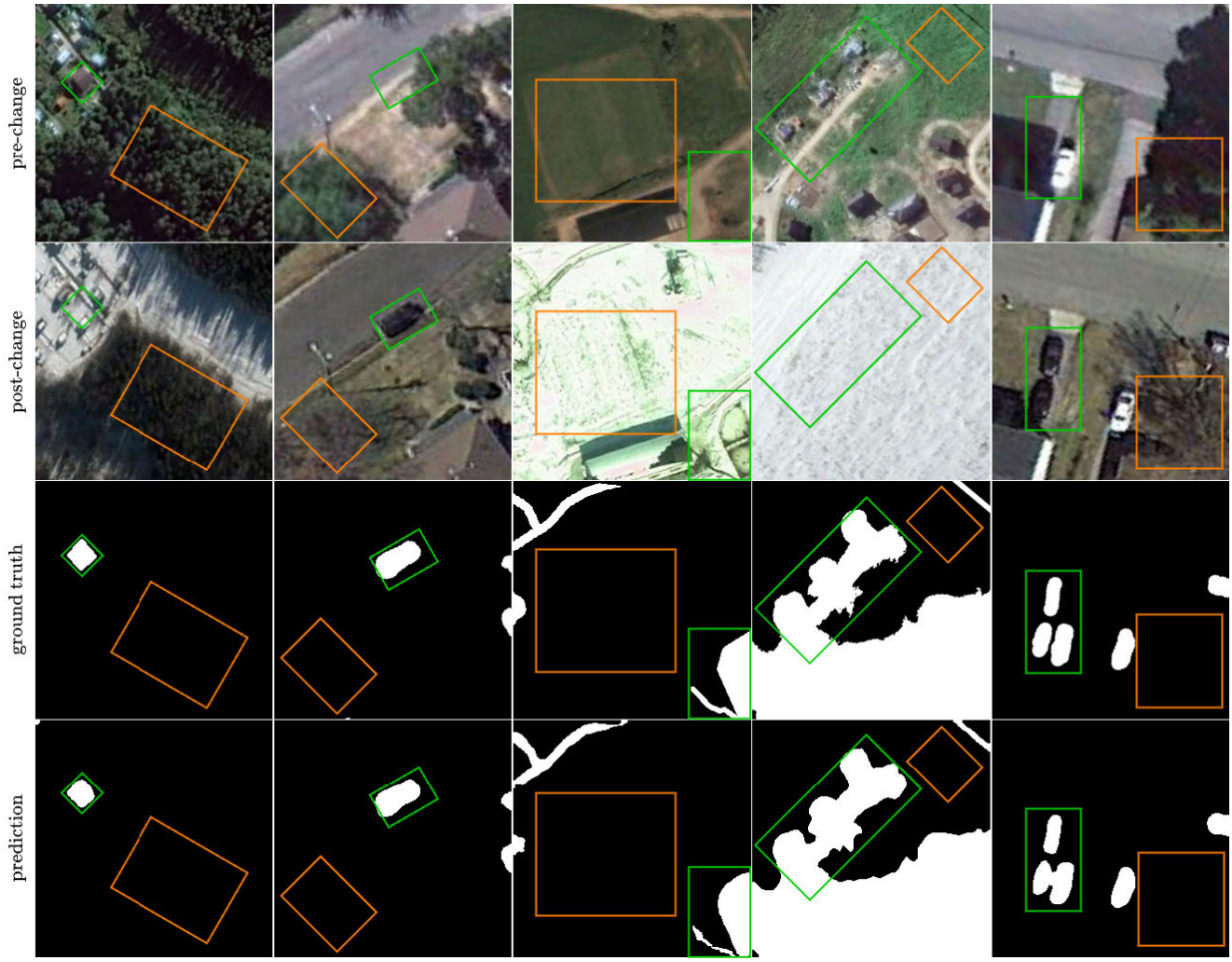

FIGURE 5. Prediction results of FocalCD on CDD test set. Green and orange rectangles represent samples of fundamental and pseudo changes.

higher resolution can further enhance the representation [57]. We also provide the global context prior to further increasing the encoder's receptive field by adding the pyramid pooling module (PPM) [58] to the feature maps of the encoder's latest stage.

Specifically, given the fused features $\hat{\mathbf{F}}^1$, $\hat{\mathbf{F}}^2$, $\hat{\mathbf{F}}^3$, and $\hat{\mathbf{F}}^4$, we apply the PPM only to $\hat{\mathbf{F}}^4$ following [59]. We create four feature maps from $\hat{\mathbf{F}}^4$, each with different scales. Technically, we perform adaptive average pooling followed by convolution with 1×1 kernel for each scale. The results are feature maps with spatial resolution 1×1, 2×2, 3×3, and 6×6, all having channel dimension D. The features are then concatenated channel-wise resulting single feature with channel dimension $4D$. The channel dimension is then reduced into D using convolution with 3×3 kernel resulting in a feature map $\tilde{\mathbf{F}}^4$. Fig. 4 illustrates the process inside the PPM.

The feature map $\tilde{\mathbf{F}}^4$ is then upsampled into the same spatial resolution as $\hat{\mathbf{F}}^3$ using linear interpolation. $\tilde{\mathbf{F}}^4$ and $\hat{\mathbf{F}}^3$ are then merged using element-wise addition. However, we first apply 1×1 convolution to $\hat{\mathbf{F}}^3$ to match the channel dimension into D. We then apply convolution with 3×3 kernel to the merged feature to reduce the aliasing effect of upsampling process resulting in $\tilde{\mathbf{F}}^3$. The process is then continued until we get four feature maps $\tilde{\mathbf{F}}^1$, $\tilde{\mathbf{F}}^2$, $\tilde{\mathbf{F}}^3$, and $\tilde{\mathbf{F}}^4$. $\tilde{\mathbf{F}}^2$, $\tilde{\mathbf{F}}^3$, and $\tilde{\mathbf{F}}^4$ are then upsampled to the same spatial resolution as $\tilde{\mathbf{F}}^1$ using linear interpolation. We then concatenate all those feature maps channel-wise then applied 3×3 convolution to reduce the channel dimension from $4D$ into D, resulting in a single feature map $\bar{\mathbf{F}} \in \mathbb{R}^{H \times W \times D}$ and then fed them into the pixel classifier. The pixel classifier is a convolution with 1×1 kernel and output channel dimension 2, representing the number of classes (changed and unchanged in this case). To build the final binary change map we apply a 0.5 threshold to all pixel probability values to build the final binary change map. We convert pixels whose value is greater than 0.5 into 255 (white), which represents the changed area, otherwise 0 (black), which represents the unchanged area.

IV. EXPERIMENTS

This section presents the details of our experiments to evaluate our proposed method, FocalCD.

A. DATASET

We use 3 CD datasets in the experiments: CDD [4], LEVIR-CD [5], and WHU-CD [5]. They are all open datasets widely

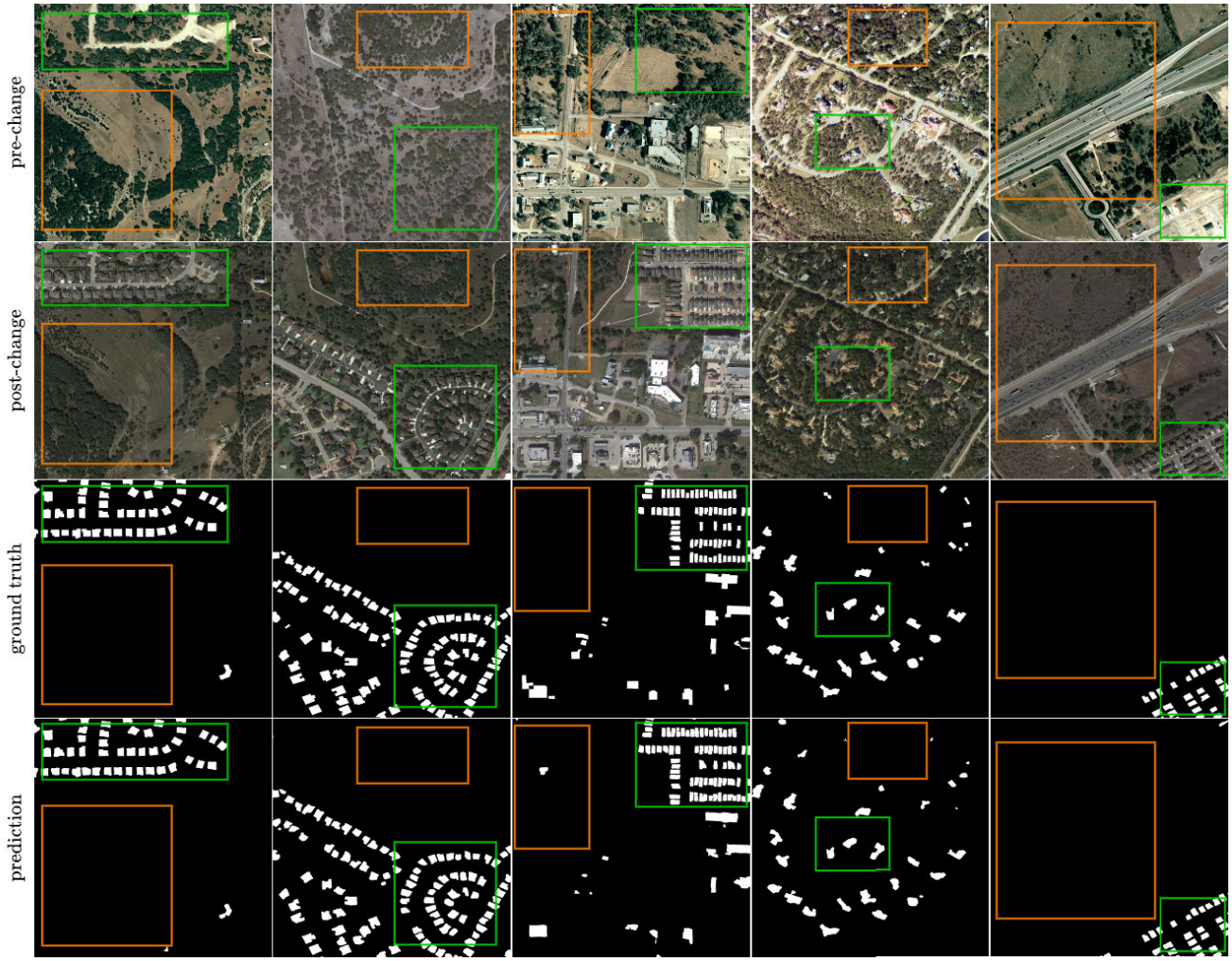

FIGURE 6. Prediction results of FocalCD on LEVIR-CD test set. Green and orange rectangles represent samples of fundamental and pseudo changes.

TABLE 1. Evaluation results on CDD, LEVIR-CD, and WHU-CD datasets (all values are in percentage (%)).

Method	CDD			LEVIR-CD			WHU-CD		
	F1	IoU	OA	F1	IoU	OA	F1	IoU	OA
FC-EF [42]	73.09	62.52	91.22	90.18	83.36	98.47	95.45	91.45	97.30
FC-Siam-Diff [42]	80.18	69.99	93.28	93.92	89.06	99.04	96.09	92.58	97.66
FC-Siam-Conc [42]	80.89	70.76	93.26	93.78	88.83	99.00	96.09	92.58	97.68
BIT [60]	71.37	61.03	91.28	90.43	83.69	98.34	91.92	85.44	95.03
ChangeFormer [34]	96.1	92.66	98.36	94.62	90.21	99.15	95.31	91.20	97.24
STANet [61]	93.62	44.00	88.00	87.26	77.40	98.66	82.32	69.95	**98.52**
SwinSUNet [11]	90.47	83.44	96.26	92.42	86.69	98.81	95.70	91.88	97.48
SiamixFormer [8]	93.21	87.73	97.25	93.09	87.73	98.92	95.96	92.35	97.62
FocalCD (ours)	**98.51**	**97.10**	**99.37**	**95.20**	**91.18**	**99.24**	**96.16**	**92.72**	97.75

used for training and evaluating the CD method. CDD is a CD dataset that contains pair of remote-sensing images within diverse seasonal conditions taken from Google Earth. The image pair consists of the registered pre-change and post-change images and the ground truth change map. Each of which is 256 × 256 pixels in size. The task is to detect fundamental changes defined in the ground truth that mainly focus on infrastructure development, such as buildings and

roads, while ignoring the seasonal changes. CDD provides three sets, each for training, validating, and testing. The total image pairs in each set are 10,000 for training, 3,000 for evaluating, and 3,000 for testing.

LEVIR-CD is a recently proposed remote sensing dataset designed to evaluate CD algorithms, particularly those leveraging deep learning techniques. The dataset comprises 637 pairs of very high resolution (VHR) Google Earth (GE) image patches, each of which is 1024 × 1024 pixels in size. These bitemporal images capture changes in land use over a period ranging from 5 to 14 years, with a particular focus on the growth of buildings. LEVIR-CD covers various facilities, including villa residences, tall apartments, small garages, and large warehouses. Specifically, the dataset emphasizes building-related changes, such as building growth (from soil/grass/hardened ground or a building under construction to new built-up areas) and building decline.

WHU-CD is a CD dataset encompassing an area struck by a 6.3-magnitude earthquake in February 2011 and subsequently rebuilt in the following years. The dataset comprises aerial images captured in April 2012, including 12,796 buildings across 20.5 square kilometers.

B. EXPERIMENTAL SETUP

We implemented FocalCD using the OpenCD [62], a CD toolkit built upon MMSegmentation [63], a widely adopted framework for image segmentation tasks. We used the patch embedding layer's dimension $C = 128$. The higher value of C will increase the model's performance in general. However, it also increases the computational cost. We used 128 because it has been used in the prior arts [23], [46] and already gave a good result. We used a stack of FMs (layer-depth) with $M = \{2, 2, 18, 2\}$. The depth in stage-3 is much bigger than in other stages to capture more fine-grained features like subtle patterns, including textures, shapes, and orientations, as these have been proved empirically in previous works [22], [23], [46], [55]. Also, we used the same drop rate of 0.5 on the drop path for FM and FFN.

We used focal level (the number of consecutive depth-wise convolutions in FM) 3 for each stage. We also used each stage's kernel size (k_1) 3 × 3. The motivation for why used these numbers is to keep the computational lightweight. Another reason is that the combination between the focal level and the kernel size already gave enough receptive field so the model can learn both short and long-range interaction. In the decoder, we use feature dimension $D = 128$.

We augmented the input images, including the ground truth change map using the standard data augmentation for images such as random crop, random rotation (at maximum 180°), and horizontal/vertical flip. All these augmentation strategies were used at 50% probability. We followed training best practices from [64] to stabilize the model's training by normalizing the image pixels using the standard normalization used for the ImageNet [65] dataset with [123.675, 116.28, 103.53] as mean and [58.395, 57.12, 57.375] as standard deviation, each for channel red, green, and blue respectively.

We trained the models using AdamW [66] optimizer with a learning rate 6×10^{-5}, β_1 0.9, β_2 0.999, and weight decay 0.01 on a single GPU V100. We used a polynomial learning rate scheduler [67] with a linear warmup for the first 1,500 iterations. We trained the models for 50,000 iterations using batch size 16. To speed up the model's convergencies, we initialized the model's weights using pre-trained weights of FocalNet [30], which has been trained on ImageNet 1K [65] dataset. The models are trained to minimize the binary cross-entropy loss.

V. RESULT & DISCUSSION

We put the evaluation results for all the experiments in Table 1. We compare our proposed FocalCD method with existing methods, including CNN-based and Transformer-based. We can see from these results that FocalCD outperforms existing CD methods on CDD, LEVIR-CD, and WHU-CD datasets according to various evaluation metrics, including F1 (F1 Score), IoU (Intersection over Union), and OA (Overall Accuracy), which are the most used metrics for evaluating the CD method. These results demonstrated the effectiveness of our proposed method.

We also present ablation studies of our proposed method, FocalCD. These ablation studies are performed to understand better the effectiveness of an individual component of FocalCD. These ablations include variations of the focal level (N), feature fusion, drop rate, layer depth, embedding size (C), and learning rate. Except for the training iteration, we used the same experimental setup for all ablation experiments as the main experiment presented in subsection IV-B. We only modify a single model component at a time for each ablation experiment while keeping up all other components. Table 4 summarizes the result of the ablation studies.

A. QUANTITATIVE EVALUATION

FocalCD outperforms SiamixFormer, a state-of-the-art CD method based on Transformer, on F1 and IoU by significant margins, 1.38 and 2.59. The margins are slightly larger when evaluated on the LEVIR-CD dataset, which is 3.62 and 5.8. On the WHU-CD dataset, FocalCD outperforms SwinSUNet, a Transformer-based CD method, on F1 by 1.65. However, SwinSUNet has more considerable accuracy on WHU-CD. After investigating the WHU-CD test set, we found enough samples having a high imbalance between changed and unchanged classes. Fig. 11 shows sample images from the WHU-CD test set that contains a high imbalance class. Class imbalances may affect accuracy as it is sensitive to class imbalances. This issue may arise as we do not focus on handling the class imbalance issue.

We report detailed evaluation results per class basis as presented in table 2. The table summarizes various evaluation metrics for unchanged and changed classes. For CDD datasets, we can see that the evaluation results for the unchanged class are better than the changed class in all

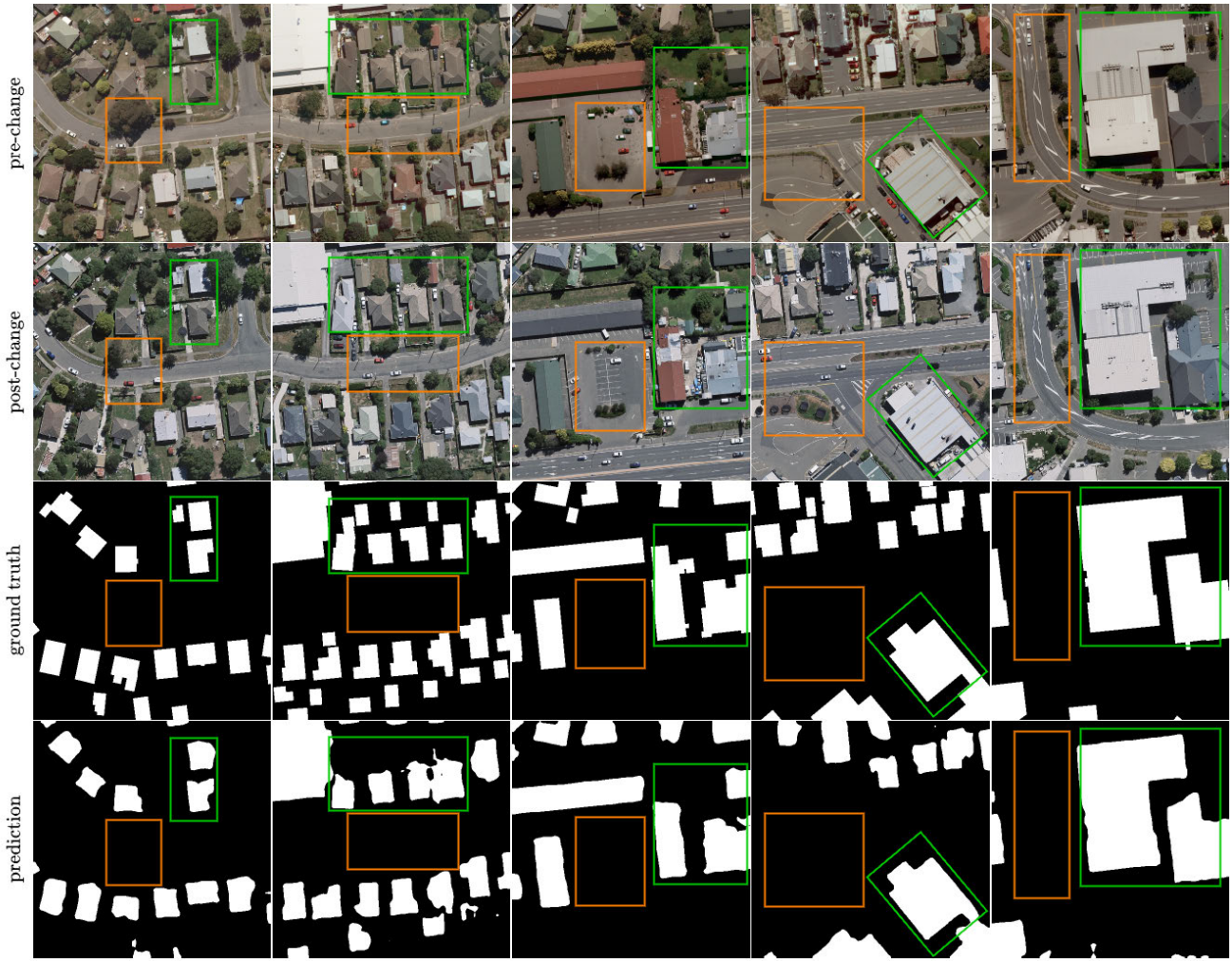

FIGURE 7. Prediction results of FocalCD on WHU-CD test set. Green and orange rectangles represent samples of fundamental and pseudo changes.

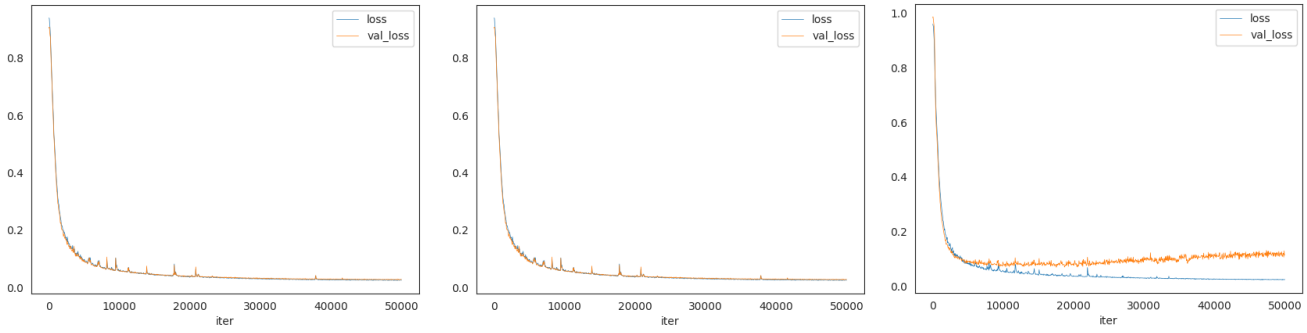

FIGURE 8. Plot of training and validation loss on (a) CDD (b) LEVIR-CD and (c) WHU CD datasets.

metrics. This also holds for two other datasets, LEVIR-CD and WHU-CD. These results conclude that detecting the changed area is more complicated than the unchanged area. However, we can see that the value of precision and recall are balanced in both classes across all datasets. This leads to a high score of F1 in both classes. Finally, the mean precision and mean recall are also balanced in all datasets.

We plot the training and validation loss for all datasets in Fig. 8. For CDD and LEVIR-CD datasets, the training and validation losses are reduced as the iteration increases.

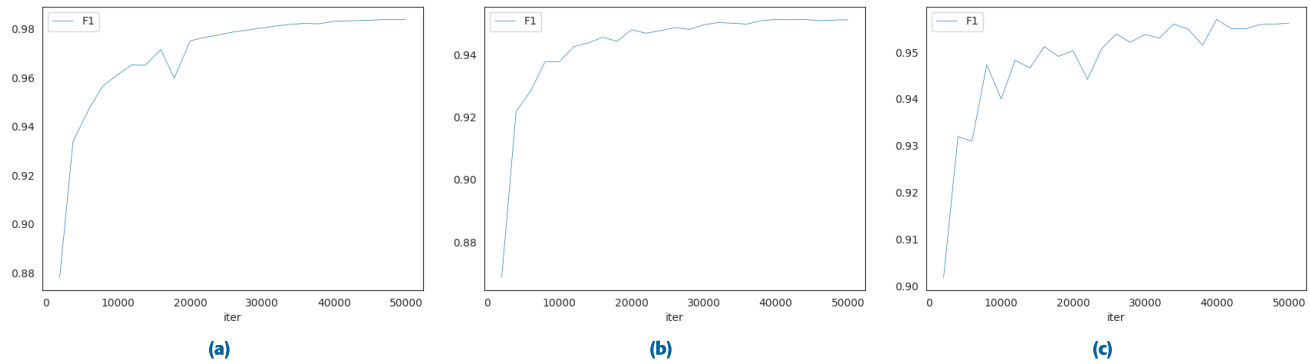

FIGURE 9. Plot of F1 score on (a) CDD, (b) LEVIR-CD, and (c) WHU CD datasets.

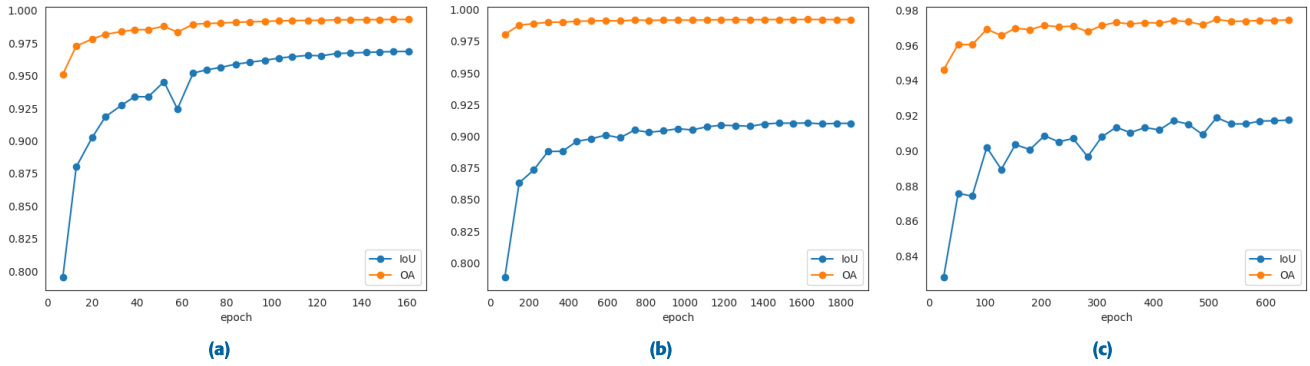

FIGURE 10. Plot of intersection over union (IoU) and overall accuracy (OA) on (a) CDD, (b) LEVIR-CD, and (c) WHU CD datasets.

TABLE 2. Detail evaluation results for unchanged and changed classes (all values are in percentage (%)). F1, P, R, mP, mR represent F1 score, precision, recall, mean precision, and mean recall, respectively.

Dataset	Unchanged					Changed					mP	mR
	F1	IoU	Acc	P	R	F1	IoU	Acc	P	R		
CDD	99.64	99.29	99.63	99.66	99.63	97.39	94.90	97.50	97.27	97.50	98.46	98.56
LEVIR-CD	99.59	99.18	99.65	99.52	99.65	90.42	82.52	89.03	91.86	89.03	95.69	94.34
WHU-CD	98.38	96.82	98.85	97.92	98.85	92.54	86.12	90.56	94.61	90.56	96.26	94.71

TABLE 3. Comparison of computation complexity.

Method	Encoder	Fusion	Decoder
SwinSUNet [11]	$\mathcal{O}\left(HW \times (3C^2 + 2Cw^2)\right)$	$\mathcal{O}\left(HW \times (C^2)\right)$	$\mathcal{O}\left(HW \times (5C^2 + C)\right)$
SiamixFormer [8]	$\mathcal{O}\left(HW \times \left((CR)^2 + \frac{2CHW}{R^2}\right)\right)$	-	$\mathcal{O}\left(HW \times (4C^2 + 3C)\right)$
FocalCD (ours)	$\mathcal{O}\left(HW \times (3C^2 + C(\Sigma_n(k_n)^2 + 2N + 3))\right)$	-	$\mathcal{O}\left(HW \times 28C\right)$

We can see that the validation loss follows the training loss smoothly without evidence of overfitting. However, the validation loss on WHU-CD started to increase at iteration 6,0000th, a signal of overfitting. This issue is still related to the class imbalances of the WHU-CD test set. However, it does not affect the final model too much, as we keep the best performance model during training.

We plot the F1 score on the validation set during training. Fig. 9 shows the plot of the F1 score for CDD, LEVIR-CD, and WHU datasets. The F1 score for CDD and LEVIR-CD shows stable F1 scores. It increases as the iteration goes further. In contrast, the F1 score on WHU-CD suffers from some perturbation. Nevertheless, the F1 score on WHU-CD still shows an increasing trend with the iteration. These conditions

FIGURE 11. Sample images show a high imbalance between changed and unchanged classes on the WHU-CD dataset. The ground-truth image contains 100% pixels from unchanged class.

also hold for IoU and OA, as shown in Fig. 10. Compared to IoU, OA has a higher value across all datasets.

B. QUALITATIVE EVALUATION

We also perform qualitative evaluations by saving the inference results on the test set across all datasets. We randomly select five samples from the test set and show them in the figures. Fig. 5 shows the inference results on the CDD dataset. We can see from this result that FocalCD is adequate to distinguish the fundamental and pseudo changes as represented in green and orange rectangles. FocalCD is also excellent at detecting the fundamental and pseudo changes on the LEVIR-CD dataset. Fig. 6 shows the inference results on the LEVIR-CD dataset. Despite the imbalances class issue, FocalCD still can spot the fundamental and pseudo changes on WHU-CD as demonstrated in Fig. 7.

We also compare the visualization result between FocalCD and existing CD methods for fairness. Figure 12 shows the visualization result on the CDD test set.

C. COMPLEXITY ANALYSIS

This section presents the computation complexity of FocalCD. We focus on the encoder's complexity and omit the decoder for simplicity, as it is much lower than the encoder. We can easily determine the complexity of FocalCD by examining (6). Significant operations come from the linear projections $q(\cdot)$ and f^h, which have joined complexity $\mathcal{O}(2C^2)$ where C is the chosen embedding dimension. Furthermore, another linear transformation $f^Z(\cdot)$ is used to project the $\mathbb{Z}_0$ resulting in total $\mathcal{O}(3C^2)$ combined with previous linear projections complexity. The hierarchical contextualization, which consists of N depth-wise convolution as defined in (2), has $\mathcal{O}\left(C\,\Sigma_n(k_n)^2\right)$ complexity. In most cases, N and $(k_n)^2$ are much smaller than C. Despite being lightweight, the linear transformation $f^g(\cdot)$ in the gated context aggregation contributes to the overall complexity. The complexity of gating mechanism is $\mathcal{O}\left(C(N+1)\right)$. Furthermore, dot product operation which is defined in (4) raises $\mathcal{O}\left(C(N+1)\right)$ complexity. Finally, the dot product operation in (6) also introduces $\mathcal{O}\left(C\right)$. Therefore, the total complexity of FocalCD is $\mathcal{O}\left(HW \times (3C^2 + C(2N+3) + C\,\Sigma_n(k_n)^2)\right)$. In contrast, SwinSUNet has complexity of $\mathcal{O}\left(HW \times (3C^2 + 2Cw^2)\right)$ where w is window size. On the other hand, SiamixFormer has $\mathcal{O}\left(HW \times (CR)^2 + 2C\left(\frac{HW}{R}\right)^2\right)$ complexity, where R is spatial reduction ratio. Table 3 summarizes the complete computation complexities for each module. To get a better understanding of these complexities, we plot the simplified version of these complexities of the methods concerning the number of inputs and embedding sizes. From Figure 13, we can see that the computation complexity of FocalCD is slightly lower than SiamixFormer and SwinSUNet.

D. IMPACT OF FOCAL LEVEL (N)

The focal level determines the receptive field size of the model's network. The receptive field size increases linearly

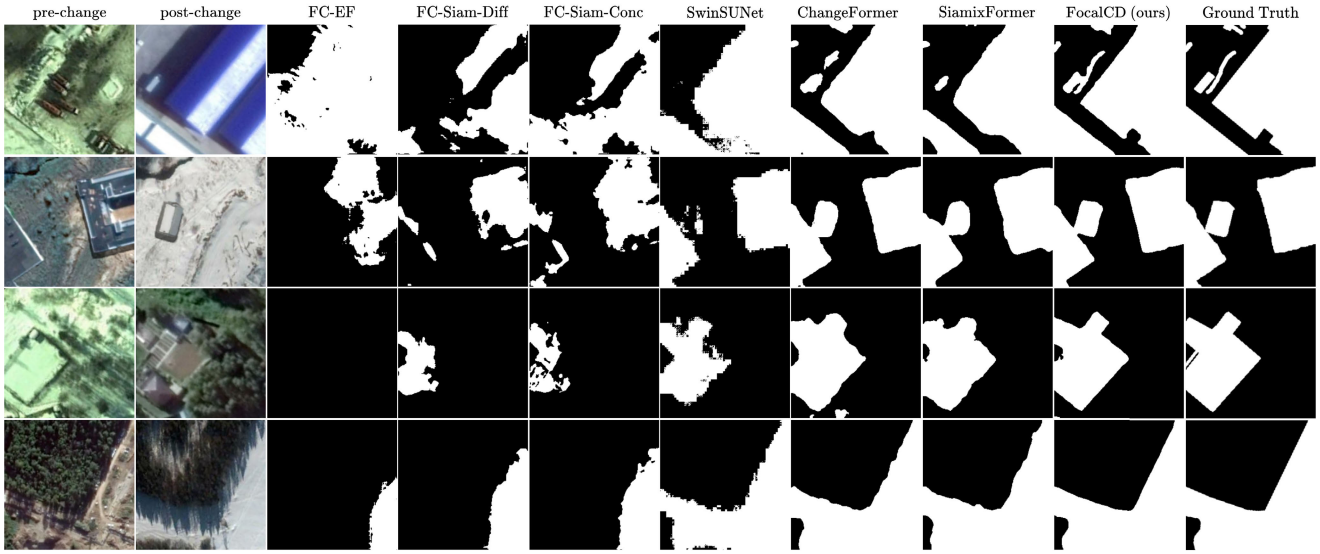

FIGURE 12. Visualization results of FocalCD compared to various CD methods on CDD test set.

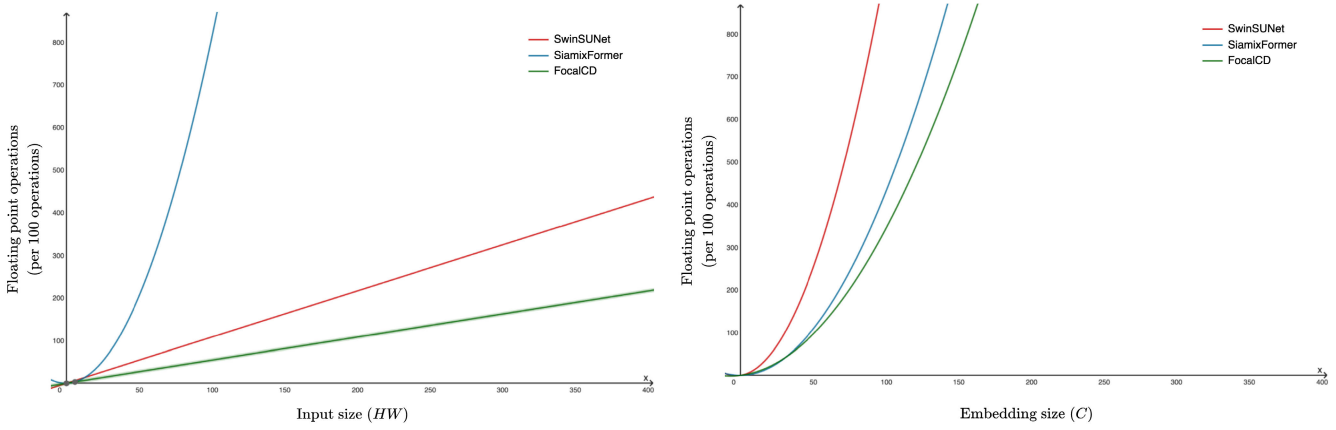

FIGURE 13. Plot of a simplified version of computation complexity for FocalCD, SwinSUNet, and SiamixFormer (a) Complexity concerning input size (*HW*) while keeping *C* constant and (b) Complexity concerning embedding size (*C*) while keeping *HW* constant.

with the focal level as depicted in (3). Theoretically, the higher value of the focal level corresponds to a longer receptive field resulting in better model performance. However, the actual receptive field can depend on various aspects. We tried to use different focal levels to find the best focal level that maximizes the model performance.

We perform experiments using the focal level values 2 and 4, which are smaller and higher than the default value 3. Tabel 4 (row 1) summarizes the experiment results. This table shows that the best results are achieved when the focal level is 3 for each stage. Despite focal level 4 resulting in better performance than focal level 2, it is still worst than focal level 3. It is weird because focal level 4 has a larger receptive field than focal level 3. It seems that the focal level 4 is not fully supported by the kernel size (k_1) 3×3. We perform a bit deeper experiment by increasing the kernel size (k_1) to 9×9, then re-run the experiment. The result is surprising. With a larger kernel size, focal level 4 outperforms focal level 3 as summarized in Tabel 4 (row 2).

E. IMPACT OF KERNEL SIZE (k_1)

Like the focal level, kernel size k_1 also determines the receptive field size of the overall network. Different from the focal level, it serves as the starting size of the receptive field, which is enlarged consecutively based on the focal level. We perform experiments to determine how kernel size k_1 impacts the overall model's performance. Specifically, we train the model using various kernel sizes. The results are shown in 4 (row 4). These results indicate that increasing the kernel size k_1 improves the model performance in all evaluation metrics.

F. IMPACT OF FEATURE FUSION

Feature fusion is essential in determining the accuracy of the generated change maps. Different fusion strategy gives

TABLE 4. Results of ablation studies (all values are in percentage (%)). The components in the bracket are for the stage 1, 2, 3, and 4, respectively.

No	Ablation	Component	F1	IoU	OA
1	Focal Level (N)	(2, 2, 2, 2)	96.62	93.57	98.57
		(3, 3, 3, 3)	**96.72**	**93.75**	**98.62**
		(4, 4, 4, 4)	96.63	93.59	98.58
2	Focal Level (N) with k_1 9×9	(2, 2, 2, 2)	97.05	94.37	98.76
		(3, 3, 3, 3)	97.08	94.42	98.77
		(4, 4, 4, 4)	**97.51**	**95.21**	**98.95**
3	Kernel Size (k_1)	3×3	96.72	93.75	98.62
		5×5	97.05	94.37	98.76
		7×7	97.08	94.42	98.77
		9×9	**97.51**	**95.21**	**98.95**
4	Feature Fusion	L_p Distance	**97.51**	**95.21**	**98.95**
		Addition	96.7	93.73	98.62
		Subtraction	97.12	94.49	98.79
		Concatenation	97.24	94.7	98.84
5	Drop Rate	0.0	**97.32**	**94.86**	**98.87**
		0.4	97.09	94.43	98.77
		0.5	97.28	94.78	98.85
		0.7	96.82	93.94	96.88
6	Layer Depth	(2, 2, 6, 2)	96.38	93.15	98.48
		(2, 2, 12, 2)	96.83	93.96	98.67
		(2, 2, 18, 2)	**97.51**	**95.21**	**98.95**
7	Embedding Size (C)	64	95.46	91.53	98.11
		96	96.34	93.07	98.47
		128	**97.51**	**95.21**	**98.95**
8	Learning Rate	6×10^{-3}	93.62	44.0	88.0
		6×10^{-4}	93.62	44.0	88.0
		6×10^{-5}	**97.27**	**94.76**	**98.85**

different result. Choosing the best fusion technique requires trial and error. We tried alternative operations other than the L_p distance for combining the feature maps, such as element-wise addition and subtraction or channel-wise concatenation. The operations are performed at individual scales same as the L_p distance. We can observe from the table 4 (row 4) that feature fusion ablation has consistent results for each operation in all evaluation metrics. The best-performed operation is L_p distance followed by concatenation, subtraction, and addition.

One of the reasons why L_p distance gives the best result is that the distance between features has a similar purpose to the actual CD task, which needs to detect changes between two images. While subtraction operation has also met this criterion, however, it can generate negative results, which usually be ignored by the network. On the other hand, concatenation operation is good for increasing feature expressiveness, but it is not suitable for CD tasks. This also holds for the additional operation because combining two features can dismiss the individual feature and yield bias that reduces CD task performance.

G. IMPACT OF DROP RATE

The main goal of the drop path is to fight overfitting on the model. It can also reduce training time significantly because the number of layers is reduced during training. Furthermore, it can also stabilize the training process. However, it can sometimes reduce the model's performance when used with a very high drop rate. Finding the best drop rate is also challenging; it requires trial and error. We perform additional experiments on various drop rates to determine the drop rate's impact on FocalCD. Table 4 (row 5) summarizes the experiment results. We found that the best performance is achieved when the drop rate is 0, which means no drop path. It seems that the higher drop rate can reduce the performance. However, we found something interesting here. Comparing the result from drop rates 0.4 and 0.5, we found that the drop rate of 0.5 resulted in better performance than that of 0.4.

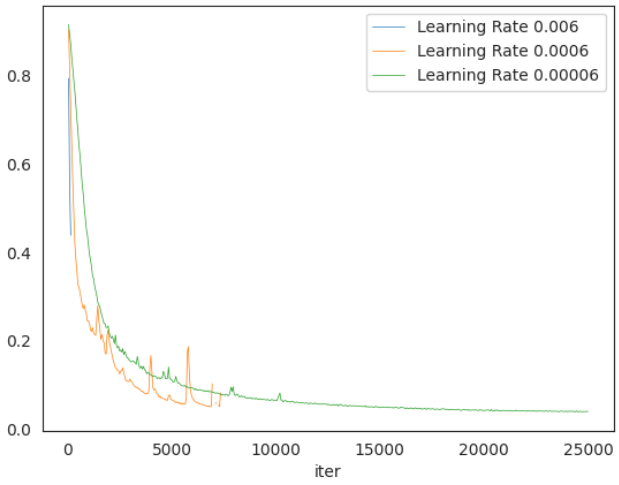

FIGURE 14. Plot of the training loss under various learning rates.

If we increase the drop rate to 0.7, the result is worse than 0.4. Likely, the drop rate does not follow any pattern. The best drop rate can only be found in the extensive trial.

H. IMPACT OF LAYER DEPTH

The next ablation is the layer depth variations. In this ablation, we tried to use different layer depths in the third stage while keeping the other stage the same. We chose only the third stage because it has important properties that return special feature maps, as mentioned before. Similar to the previous ablations, the results of this ablation are also consistent for each layer depth in all evaluation metrics. The deeper the layer will result in better performance. These results indicate that the network learns more unique features as the layer goes deeper.

I. IMPACT OF EMBEDDING SIZE (C)

The embedding size (C) can be treated as the learning capacity. The higher the size, the more pattern can be learned by the model, leading to better model performance. However, the embedding size can significantly increase the model parameter, which requires more training resources. Nevertheless, we perform experiments to see how the embedding size impacts the FocalCD performance. Table 4 (row 7) summarizes the experiment results. It is clear from the table that a higher embedding size yields better model performance.

J. IMPACT OF LEARNING RATE

A learning rate is an essential component that determines the success of deep model training. Failure to pick an appropriate learning rate can prevent the model from converging to the optimal solution. However, finding the best learning rate is not trivial; it requires much trial and error. Specifically, it depends on various aspects, including the complexity of the model, the loss function, and the optimizer. Furthermore, no rule of thumb specifies picking the best learning rate.

We perform experiments on various learning rates to see the impact of the learning rate on the FocalCD performance. Table 4 (row 8) summarizes the experiments' results. These results show that the effective learning rate for FocalCD is 6×10^{-5}. Any learning rates larger than 6×10^{-5} make the model fail to converge. This can lead to performance reduction of the model. In contrast, picking the learning rates smaller than 6×10^{-5} makes the model converge very slowly. Fig. 14 shows the training loss plot under these learning rate variations. The figure shows that the training loss in learning rate 6×10^{-5} converges as the iterations increase. In contrast, the training loss in learning rate 6×10^{-4} diminishes at around 7.5k iterations, preventing the model from converging. The training loss is getting worse for the learning rate 6×10^{-3}; it diminishes at the beginning of the iteration.

VI. CONCLUSION

We proposed FocalCD, a novel CD method for detecting changes in high-resolution remote sensing images. FocalCD consists of 3 main modules: encoder, multi-scale feature fusion, and decoder. We leverage a recently proposed focal modulation architecture capable of capturing local and global interaction to handle high-resolution bi-temporal input. FocalCD is attention-free and does not suffer from quadratic computation complexity.

FocalCD incorporates the efficiency of CNN and the capability of ViT. The power of FocalCD imposed by the focal modulation enables it to distinguish between fundamental and pseudo changes by performing adaptive multi-scale interactions. The multi-scale feature fusion combines the feature maps produced by the two-stream encoder. The decoder then processes the combined feature maps to build the binary change map. The decoder comprises the feature pyramid network and pyramid pooling module. Extensive experiments on CDD, LEVIR-CD, and WHU-CD demonstrated the effectiveness of our proposed method. It outperforms the state-of-the-art CD method while having comparable or lower computation complexity. Despite its advantages, FocalCD is not specifically designed to handle class imbalance. Therefore, class imbalance issues may affect the evaluation result, especially for metrics that are sensitive to class imbalance. Future work can improve FocalCD by adding the capability to handle class imbalance issues.

ACKNOWLEDGMENT

The authors would like to thank Laboratory 1231 Fasilkom Universitas Indonesia (UI) for their helpful discussion and feedback.

REFERENCES

[1] J. R. Jensen and D. K. Lulla, "Introductory digital image processing: A remote sensing perspective," *Geocarto Int.*, vol. 2, no. 1, p. 65, 1987, doi: 10.1080/10106048709354084.
[2] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *Int. J. Remote Sens.*, vol. 10, no. 6, pp. 989–1003, Jun. 1989.
[3] S. Tian, A. Ma, Z. Zheng, and Y. Zhong, "Hi-UCD: A large-scale dataset for urban semantic change detection in remote sensing imagery," 2020, *arXiv:2011.03247*.

[4] M. A. Lebedev, Y. V. Vizilter, O. V. Vygolov, V. A. Knyaz, and A. Y. Rubis, "Change detection in remote sensing images using conditional adversarial networks," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. XLII–2, pp. 565–571, May 2018. [Online]. Available: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2/565/2018/

[5] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8444434/

[6] R. C. Daudt, B. L. Saux, A. Boulch, and Y. Gousseau, "Urban change detection for multispectral Earth observation using convolutional neural networks," 2018, *arXiv:1810.08468*.

[7] T. Leichtle, C. Geiß, M. Wurm, T. Lakes, and H. Taubenböck, "Unsupervised change detection in VHR remote sensing imagery—An object-based clustering approach in a dynamic urban environment," *Int. J. Appl. Earth Observ. Geoinformation*, vol. 54, pp. 15–27, Feb. 2017. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0303243416301490

[8] A. Mohammadian and F. Ghaderi, "SiamixFormer: A Siamese transformer network for building detection and change detection from bi-temporal remote sensing images," 2022, *arXiv:2208.00657*.

[9] J. Chen, Z. Yuan, J. Peng, L. Chen, H. Huang, J. Zhu, Y. Liu, and H. Li, "DASNet: Dual attentive fully convolutional Siamese networks for change detection in high-resolution satellite images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1194–1206, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9259045/

[10] S. Fang, K. Li, J. Shao, and Z. Li, "SNUNet-CD: A densely connected Siamese network for change detection of VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9355573/

[11] C. Zhang, L. Wang, S. Cheng, and Y. Li, "SwinSUNet: Pure transformer network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5224713. [Online]. Available: https://ieeexplore.ieee.org/document/9736956/

[12] X. Zhang, P. Xiao, X. Feng, and M. Yuan, "Separate segmentation of multi-temporal high-resolution remote sensing images for object-based change detection in urban area," *Remote Sens. Environ.*, vol. 201, pp. 243–255, Nov. 2017. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0034425717304340

[13] Y. Li, S. Martinis, S. Plank, and R. Ludwig, "An automatic change detection approach for rapid flood mapping in Sentinel-1 SAR data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 73, pp. 123–135, Dec. 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0303243418302782

[14] A. Fujita, K. Sakurada, T. Imaizumi, R. Ito, S. Hikosaka, and R. Nakamura, "Damage detection from aerial images via convolutional neural networks," in *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2017, pp. 5–8. [Online]. Available: http://ieeexplore.ieee.org/document/7986759/

[15] R. Gupta, B. Goodman, N. N. Patel, R. Hosfelt, S. Sajeev, E. T. Heim, J. Doshi, K. Lucas, H. Choset, and M. E. Gaston, "xBD: A dataset for assessing building damage from satellite imagery," 2019, *arXiv:1911.09296*.

[16] V. Rážička, A. Vaughan, D. D. Martini, J. Fulton, V. Salvatelli, C. Bridges, G. Mateo-Garcia, and V. Zantedeschi, "Unsupervised change detection of extreme events using ML on-board," 2021, *arXiv:2111.02995*.

[17] F. Bovolo and L. Bruzzone, "A split-based approach to unsupervised change detection in large-size multitemporal images: Application to tsunami-damage assessment," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1658–1670, Jun. 2007. [Online]. Available: http://ieeexplore.ieee.org/document/4215033/

[18] W. A. Malila, "Change vector analysis: An approach for detecting forest changes with Landsat," in *Proc. 6th Annu. Symp. Mach. Process. Remotely Sensed Data*, West Lafayette, IN, USA, 1980, pp. 3–6.

[19] B. Desclée, P. Bogaert, and P. Defourny, "Forest change detection by statistical object-based method," *Remote Sens. Environ.*, vol. 102, nos. 1–2, pp. 1–11, May 2006.

[20] H. Lyu, H. Lu, and L. Mou, "Learning a transferable change rule from a recurrent neural network for land cover change detection," *Remote Sens.*, vol. 8, no. 6, p. 506, Jun. 2016. [Online]. Available: http://www.mdpi.com/2072-4292/8/6/506

[21] W. Wang, L. Yao, L. Chen, B. Lin, D. Cai, X. He, and W. Liu, "CrossFormer: A versatile vision transformer hinging on cross-scale attention," 2021, *arXiv:2108.00154*.

[22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[23] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021, *arXiv:2103.14030*.

[24] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, L. Yuan, D. Chen, and B. Guo, "CSWin transformer: A general vision transformer backbone with cross-shaped windows," 2021, *arXiv:2107.00652*.

[25] Y. Li, C. Wu, H. Fan, K. Mangalam, B. Xiong, J. Malik, and C. Feichtenhofer, "MViTv2: Improved multiscale vision transformers for classification and detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4794–4804.

[26] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "PVT v2: Improved baselines with pyramid vision transformer," 2021, *arXiv:2106.13797*.

[27] H. Yin, A. Vahdat, J. M. Alvarez, A. Mallya, J. Kautz, and P. Molchanov, "A-ViT: Adaptive tokens for efficient vision transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10799–10808. [Online]. Available: https://ieeexplore.ieee.org/document/9880220/

[28] P. Wang, X. Wang, F. Wang, M. Lin, S. Chang, H. Li, and R. Jin, "KVT: K-NN attention for boosting vision transformers," 2021, *arXiv:2106.00515*.

[29] J. Yang, C. Li, P. Zhang, X. Dai, B. Xiao, L. Yuan, and J. Gao, "Focal self-attention for local-global interactions in vision transformers," 2021, *arXiv:2107.00641*.

[30] J. Yang, C. Li, X. Dai, L. Yuan, and J. Gao, "Focal modulation networks," 2022, *arXiv:2203.11926*.

[31] Z. Wei, H. Pan, L. Li, M. Lu, X. Niu, P. Dong, and D. Li, "DMFormer: Closing the gap between CNN and vision transformers," 2022, *arXiv:2209.07738*.

[32] X. Song, Z. Hu, and J. Li, "PSTNet: Progressive sampling transformer network for remote sensing image change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 8442–8455, 2022.

[33] H. Chen, C. Wu, B. Du, L. Zhang, and L. Wang, "Change detection in multisource VHR images via deep Siamese convolutional multiple-layers recurrent neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2848–2864, Apr. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8937755/

[34] W. G. C. Bandara and V. M. Patel, "A transformer-based Siamese network for change detection," 2022, *arXiv:2201.01293*.

[35] G. Wang, B. Li, T. Zhang, and S. Zhang, "A network combining a transformer and a convolutional neural network for remote sensing image change detection," *Remote Sens.*, vol. 14, no. 9, p. 2228, May 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/9/2228

[36] Q. Ke and P. Zhang, "Hybrid-TransCD: A hybrid transformer remote sensing image change detection network via token aggregation," *ISPRS Int. J. Geo-Inf.*, vol. 11, no. 4, p. 263, Apr. 2022. [Online]. Available: https://www.mdpi.com/2220-9964/11/4/263

[37] Q. Li, R. Zhong, X. Du, and Y. Du, "TransUNetCD: A hybrid transformer network for change detection in optical remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5622519. [Online]. Available: https://ieeexplore.ieee.org/document/9761892/

[38] D. Peng, Y. Zhang, and H. Guan, "End-to-end change detection for high resolution satellite images using improved UNet++," *Remote Sens.*, vol. 11, no. 11, p. 1382, Jun. 2019. [Online]. Available: https://www.mdpi.com/2072-4292/11/11/1382

[39] C. Zhang, P. Yue, D. Tapete, L. Jiang, B. Shangguan, L. Huang, and G. Liu, "A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 183–200, Aug. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0924271620301532

[40] D. Peng, L. Bruzzone, Y. Zhang, H. Guan, and P. He, "SCDNET: A novel convolutional network for semantic change detection in high resolution optical remote sensing imagery," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 103, Dec. 2021, Art. no. 102465. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0303243421001720

[41] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Proc. Int. Workshop Multimodal Learn. Clin. Decis. Support* in Lecture Notes in Computer Science, vol. 11045, Granada, Spain: Springer, Jun. 2018, pp. 3–11, doi: 10.1007/978-3-030-00889-5_1.

[42] R. C. Daudt, B. L. Saux, and A. Boulch, "Fully convolutional Siamese networks for change detection," 2018, *arXiv:1810.08462*.
[43] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," 2015, *arXiv:1505.04597*.
[44] M. Zhang and W. Shi, "A feature difference convolutional neural network-based change detection method," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7232–7246, Oct. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9052762/
[45] Q. Ding, Z. Shao, X. Huang, and O. Altan, "DSA-Net: A novel deeply supervised attention-guided network for building change detection in high-resolution remote sensing images," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 105, Dec. 2021, Art. no. 102591. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0303243421002981
[46] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," 2021, *arXiv:2102.12122*.
[47] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," 2021, *arXiv:2105.15203*.
[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017, *arXiv:1706.03762*.
[49] Y. Guo, Y. Li, L. Wang, and T. Rosing, "Depthwise convolution is all you need for learning multiple visual domains," in *Proc. AAAI*, Jul. 2019, vol. 33, no. 1, pp. 8368–8375.
[50] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.
[51] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.
[52] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[53] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*.
[54] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 646–661.
[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[56] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. J'egou, "Training data-efficient image transformers & distillation through attention," 2020, *arXiv:2012.12877*.
[57] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[58] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239.
[59] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified perceptual parsing for scene understanding," 2018, *arXiv:1807.10221*.
[60] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5607514.
[61] H. Chen and Z. Shi, "A spatial–temporal attention-based method and a new dataset for remote sensing image change detection," *Remote Sens.*, vol. 12, no. 10, p. 1662, May 2020. [Online]. Available: https://www.mdpi.com/2072-4292/12/10/1662
[62] S. Fang, K. Li, and Z. Li, "Changer: Feature interaction is what you need for change detection," 2022, *arXiv:2209.08290*.
[63] M. Contributors. (2020). *MMSegmentation: Openmmlab Semantic Segmentation Toolbox and Benchmark*. [Online]. Available: https://github.com/open-mmlab/mmsegmentation
[64] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for image classification with convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 558–567.
[65] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[66] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv*, vol. abs/1711.05101, 2019.
[67] P. Mishra and K. Sarawadekar, "Polynomial learning rate policy with warm restart for deep neural network," in *Proc. TENCON IEEE Region 10 Conf. (TENCON)*, Oct. 2019, pp. 2087–2092.

LHUQITA FAZRY is currently pursuing the master's degree with the Faculty of Computer Science, University of Indonesia. He has a solid mathematical background. Furthermore, he has more than ten years of coding experience in software development. His current research interests include deep learning for computer vision, especially in vision transformers and remote sensing change detection.

MGS M. LUTHFI RAMADHAN received the master's degree from the Faculty of Computer Science, University of Indonesia. His current research interests include deep learning, pattern recognition, and computer vision.

WISNU JATMIKO (Senior Member, IEEE) received the D.Eng. degree from Nagoya University, Japan, in 2007. He is currently a Full Professor with the Faculty of Computer Science, Universitas Indonesia. His current research interests include autonomous robots, optimization, and real-time traffic monitoring systems.