remote sensing

MDPI

Article

CORN: An Alternative Way to Utilize Time-Series Data of SAR Images in Newly Built Construction Detection

Raveerat Jaturapitpornchai [1,*], Poompat Rattanasuwan [2] and Masashi Matsuoka [1] and Ryosuke Nakamura [3]

[1] Department of Architecture and Building Engineering, Tokyo Institute of Technology, Yokohama 226-8502, Japan; matsuoka.m.ab@m.titech.ac.jp

[2] Department of Information and Communication Engineering, Tokyo Institute of Technology, Yokohama 226-8503, Japan; rattanasuwan.p.aa@m.titech.ac.jp

[3] National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan; r.nakamura@aist.go.jp

* Correspondence: jaturapitpornchai.r.aa@m.titech.ac.jp; Tel.: +81-45-924-5605

Received: 8 March 2020; Accepted: 17 March 2020; Published: 19 March 2020

Abstract: The limitations in obtaining sufficient datasets for training deep learning networks is preventing many applications from achieving accurate results, especially when detecting new constructions using time-series satellite imagery, since this requires at least two images of the same scene and it must contain new constructions in it. To tackle this problem, we introduce Chronological Order Reverse Network (CORN)—an architecture for detecting newly built constructions in time-series SAR images that does not require a large quantity of training data. The network uses two U-net adaptations to learn the changes between images from both Time 1–Time 2 and Time 2–Time 1 formats, which allows it to learn double the amount of changes in different perspectives. We trained the network with 2028 pairs of 256 × 256 pixel SAR images from ALOS-PALSAR, totaling 4056 pairs for the network to learn from, since it learns from both Time 1–Time 2 and Time 2–Time 1. As a result, the network can detect new constructions more accurately, especially at the building boundary, compared to the original U-net trained by the same amount of training data. The experiment also shows that the model trained with CORN can be used with images from Sentinel-1. The source code is available at https://github.com/Raveerat-titech/CORN.

Keywords: satellite imagery; SAR; deep learning; U-net; urban change

1. Introduction

The popularity of deep learning approaches is continuously increasing, as these approaches have proven their potential by generating many state-of-the-art results for various fields of study [1,2]. For example, U-net [3] is one of the most used deep learning architectures, as it can accurately perform an image segmentation when trained with a sufficient amount of training data. Training data are the most important requirement when training of deep learning networks. The more data we use for training, the better the possibility that the model will make a more accurate prediction [4]. On the other hand, if there are not enough training data, the model tends to perform worse or may not be able to predict at all. This statement is applicable regardless of the field of study, including in remote sensing.

The use of satellite data opens doors to many possibilities. We can use it for various applications that involve viewing the Earth from above [5,6]. With this ability at our disposal, we can better

manage land use planning—for example, when planning the expansion of a city [7–9]. As time goes by, our living areas tend to grow bigger and bigger. Without good land use management, we may eventually cause complete deforestation as a result of building constructions for human beings [10,11]. To tackle this problem, we can use satellite images to create time-series data in order to observe how the city has changed—specifically, where new buildings have been built. The fundamental way to do so is to generate a difference image between two images using mathematical operations, and then perform segmentation to extract the area of changes. Some publications use threshold-based methods for segmentation, such as Y. Ban and O. Yousif [12], who applied several thresholding criteria on the difference image to obtain the urban change area. However, thresholding methods usually lead to false detections when the urban or non-urban area has an unordinary intensity in terms of its change behavior [13]. This can happen with any change detection approach that is based on the difference between images, even when using deep learning techniques such as in [14]. As a result, methods using deep learning have been widely researched, as they can generate more accurate results, while not having to depend on the difference image. For example, R.C. Daudt et al. [15] published an urban change-detection method based on convolutional neural networks [16] and Siamese networks [17] using bi-temporal multispectral images. Although the difference image was not used, the detection result was still not very accurate, especially at the construction boundaries. While Y. Xu et al. [18] were able to detect buildings with clear boundaries using U-net on very-high-resolution satellite imagery, these kinds of data using optical sensors unfortunately cannot capture Earth's surface when it is covered by clouds. The problem applies to some accurate building detection methods, which have a possibility to be extended for using in building change detection, that use the characteristic of the shape of building roofs from a very-high-resolution imagery [19–21]. As well as being hindered by cloud contamination, these images are also difficult to obtain especially when studying an event in the distant past. A similar reason can also be applied to methods such as [22,23] that require images from multiple sensors, since some data are not accessible to the public. We have previously addressed and tried to fix this problem [13] by obtaining the detection result directly from a pair of Synthetic aperture radar (SAR) images from two different time points without generating the difference image. Using U-net as the network contains the skip connection, which can help the decoder part to receive low-level features from the encoder, to allow it to generate a result with more solid boundaries. An SAR image is a kind of satellite image that is captured by the reflectance of microwaves emitted from the satellite to the earth's surface. With this property, a SAR image can be captured regardless of the weather condition of the specific area, which makes it a good substitution for an optical image in a cloudy area, such as tropical countries. However, as many satellites, especially satellites with SAR sensors, are orbiting around the earth and are unable to take images of the same area as frequently as we may want, the number of images we can use to make time-series data is not high enough to use at maximum potential. Especially with deep learning, in order to train a deep learning network efficiently, a large amount of data need to be used in the training process. As we stated earlier, the amount of time-series data of SAR images is limited, which makes it even harder to apply to deep learning. Not to mention, the term "training data" always includes ground truth data. This means not only are the time-series images required, the ground truths of the building of constructions are also needed in order to train a deep learning network. As the ground truths are usually created by humans, obtaining enough data for the training process is both costly and time-consuming.

The limited number of training data prevented our previous publication [13] regarding the detection of newly built constructions from two different time points using SAR images based on U-net architecture from being more accurate. In order to utilize the training data, in this paper, we introduce a new way to detect newly built constructions in SAR images by proposing a network architecture called "Chronological Order Reverse Network" (CORN), which can learn to detect constructions more efficiently when the same number of SAR time-series data and ground truths are used. CORN is based on the assumption that regardless of whether the changes are found from before–after (Time 1–Time 2) or after–before (Time 2–Time 1), even though the detection in Time 1–Time 2 result in the appearance of constructions and Time 2–Time 1 result in the disappearance of

constructions, the changes are still at the same spots with the same shape. This means that both types can be correctly associated with the same ground truth data. While normally, the detection of new buildings is supposed to use the data in Time 1–Time 2 format, our proposed architecture takes both Time 1–Time 2 and Time 2–Time 1 formats of data to allow learning based on both of the changing features to make it more viable. With this architecture, the amount of training data of the network appears to be doubled. This allows the network to be trained with a greater variation of data, and can result an increased detection accuracy without having to use more SAR data or create any additional ground truths. Moreover, CORN has the potential to use SAR images from other satellites and other environments because the training back and forth causes the model to be more robust.

In summary, the objective of this paper is to cope with the lack of training data when training deep learning networks, which leads to inaccurate results in newly built construction detection. We do so by proposing a network architecture called "CORN", which doubles the training set by reversing the chronological order of the dataset. CORN contains two U-net adaptations; one trains on Time 1–Time 2 images, and the other one trains on Time 2–Time 1 images. The proposed network not only increases the detection accuracy, but can also be used in a greater variety of settings of the data; specifically, images from other satellites and other acquisition conditions, including the terrain of the testing area.

2. Network Description

2.1. Architecture Detail

The proposed architecture mainly consists of two U-net networks. The first one (upper side of Figure 1) is for training the network to learn the features of change in the appearance of buildings from Time 1–Time 2 times-series SAR images. The another one (lower side of Figure 1) is for learning the change in the disappearance of buildings from Time 2–Time 1 image. Each encoder 1 receives pairs of training data in the form of Time 1–Time 2 for the upper side and Time 2–Time 1, which are generated by the reverse of the original data, for the lower side. By having two networks, the network acts like it has been trained with double the number of datasets than actually exist, allowing it to learn a greater variety of features of change in different perspectives, since it learns from different forms of the dataset. These two U-nets have exactly the same architecture as the original one, except our little modification at encoder 8 and the skip connection. Instead of using the ordinary encoder 8, which is obtained through seven repetitions of Zeropadding-convolution-BatchNorm-ReLU [24] layers from each encoder block (details shown in Table 1), we let encoder 8 from the Time 1–Time 2 side and from the Time 2–Time 1 side share the features they have learned by

$$Encoder8 = 0.7\left(Encoder8_{self}\right) + 0.3\left(Encoder8_{opposite}\right) \quad (1)$$

This means encoder 8 from the Time 2–Time 1 side consists of 70% of what it has learned by itself, and 30% of what the Time 2–Time 1 side has learned; in the case of encoder 8 from the Time 2–Time 1 side, this pattern would be reversed. The portion of 7:3 is the most suitable for our dataset, as it gives the encoder 8 on each side some features that cannot be learned by itself, while not too much of what it has learned is affected. We will discuss more on this matter in Section 4.1 of the paper.

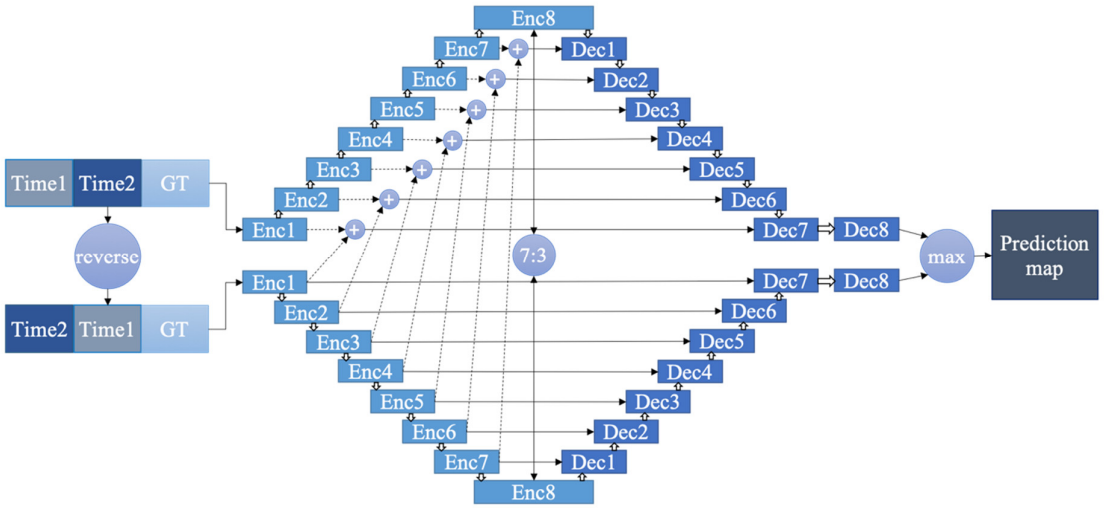


Figure 1. Architecture of the Chronological Order Reverse Network (CORN).

While the skip connection in the original U-net directly passes the features from each encoder to each corresponding decoder, which allows it to receive significant information regarding the edges and boundaries of the features, we do the same in our proposed architecture, but in a different way. As encoder 8, which is the starting point for the decoder, is influenced by the information of the opposite side, using such a straightforward skip connection would result in the decoder failing to generate an output that includes features from both sides. Thus, we solved this by adding the features from the encoders of both sides before passing it to the corresponding decoder. By following this approach, the decoder is able to generate an output with features learned by its own encoders, but with influence from the other side, while receiving all boundary information from both sides. However, we only applied this with the Time 1–Time 2 side, as the Time 2–Time 1 side used the traditional direct skip connection. The reason is that if we apply this on both sides (Figure 2), the boundary information shared between them will be too much, and will lead to a limited result within these boundaries. The evidence supporting this assumption is discussed in Section 4.2.

Table 1. Detail of the encoder and the decoder.

Encoder	Decoder
PCR (256,2,4,2)	CRD (1,512,2,2)
PCBR (128,64,4,2)	CRD (2,1024,2,2)
PCBR (64,128,4,2)	CRD (4,1024,2,2)
PCBR (32,256,4,2)	CRD (8,1024,2,2)
PCBR (16,512,4,2)	CRD (16,1024,2,2)
PCBR (8,512,4,2)	CRD (32,512,2,2)
PCBR (4,512,4,2)	CRD (64,256,2,2)
PCBR (2,512,4,2)	C (128,128,2,2)

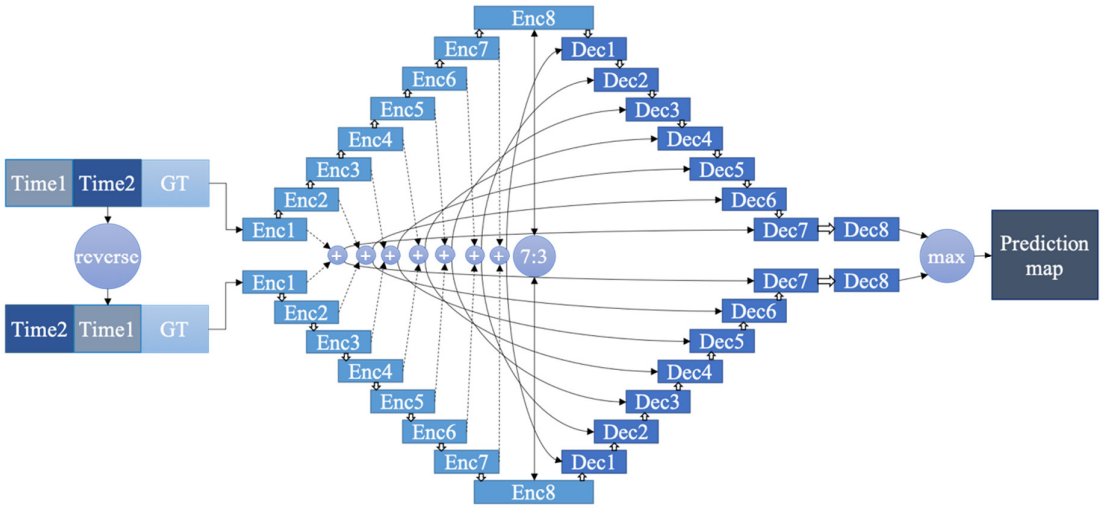


Figure 2. Architecture of CORN with an additional skip connection to both decoder sides.

Lastly, we then applied the maximum operation between these two results to draw the best result out of each one. In training, loss was calculated with the weighted binary cross entropy function [25], as our dataset contained a lot of negative class pixels (non-changed areas), while the number of positive class pixels (new construction areas) was small. The weight of loss function is the division of the percentage of negative pixels in the training set by the percentage of positive pixels in the training set, as per Ref. [13]. In our case, the weight was 181.5, which is the result of the rate of white pixels (positive class pixels) = 0.548% and the rate of black pixels (negative class pixels) = 99.452%.

In Table 1, P, C, B, R, and D represent the layers of zero padding (size 1,1), convolution (in the encoder) or deconvolution (in the decoder), batch normalization (0.2), ReLU, and dropout, respectively. From left to right, the numbers in parentheses indicate the input size^2, number of features, filter size, and stride amount of convolution filters, respectively. As we followed the method applied by Isola et al. [26], all of the ReLUs in the encoder were leaky with a slope of 0.2, while the ReLU functions in the decoder were not leaky. The dropout rate was 0.5.

2.2. Network Training

The training set in this paper was the same as in our previous work [13] to demonstrate the performance of our proposed architecture, since the goal of this paper was to increase the detection accuracy while not having to increase the number of datasets. We used three pairs of ALOS-PALSAR bitemporal data of Bangkok, Thailand, including 1 January 2008/15 January 2010, 12 January 2009/15 January 2010, and 1 January 2008/12 January 2009 in HH polarization at a 15 m/pixel resolution in ascending orbit mode. These SAR images were acquired in the right-looking direction with an off-nadir angle of 34.3°. The noise level of every image was suppressed using a 3 × 3 Lee filter [27]. The normalization was made for all image intensities to a range of [–1, 1] to avoid the problem of inconsistency between the data. Two dates from each set of Bangkok SAR images and the corresponding ground truths of the same area, which were created manually by drawing polygons of where building changes were detected in Google Earth software, were then stacked and cut into 256 × 256 pixel patches for a total number of 2028 pairs. Please note that a patch must contain at least one polygon to be included in the training set. A patch size of 256 × 256 was proven in our previous work to be the most suitable number, since a patch contains a suitable portion between positive and negative classes for training building change detection. Our number of training sets was considerably smaller than other well-known deep learning datasets such as CIFAR-10 [28], which includes 10 categories of natural image datasets containing 60,000 images.

We tried to train the network the same way as we trained U-net in the previous work so as to fairly observe the difference in performance between these two models. Thus, the number of epochs was 10 with a batch size of 16. The Adam optimizer was used at a learning rate of 0.001. The model finished training in approximately 71 min (10 epochs), while the original U-net model took approximately 55 min (10 epochs) on the same machine. The specification of the machine was Intel(R) Xeon(R) E5-2630 v4 @ 2.20 GHz CPU with 10 cores (dual thread) and an NVIDIA Tesla P100 GPU computing processor (Tesla P100 SXM2 16 GB).

3. Dataset

This section describes all the datasets we use in this study. As we stated earlier in Section 2.2, the three sets of Bangkok ALOS-PALSAR were used in training the network. In testing, two sets of Bangkok at the different time from those in training were used along with the image from Hanoi and Xiamen captured with the same satellite and acquisition conditions as the Bangkok sets used in training. Please note that while the training data were 256 × 256 pixels, the data prepared for testing were 400 × 400 pixels and chosen for ease of visual inspection. While these datasets are the same as in [13], in this paper, we add one more testing set which is Chiang Mai, Thailand, captured by Sentinel-1 to see if our proposed model trained with ALOS-PALSAR can detect new constructions in images from different satellites or not. All the datasets used in this paper are summarized in Table 2.

Table 2. Acquisition information of dataset. SAR—synthetic-aperture radar.

Purpose	Location	Acquisition Date of SAR Images (Time 1–Time 2)	Acquisition Satellite	Resolution (meters)
Training	Bangkok, Thailand	1 January 2008–15 January 2010	ALOS-PALSAR	15
		12 January 2009–15 January 2010	ALOS-PALSAR	15
		1 January 2008–12 January 2009	ALOS-PALSAR	15
Testing	Bangkok, Thailand	27 November 2008–15 January 2010	ALOS-PALSAR	15
		12 January 2009–21 November 2009	ALOS-PALSAR	15
	Hanoi, Vietnam	2 February 2007–13 February 2011	ALOS-PALSAR	15
	Xiamen, China	22 January 2007–2 November 2010	ALOS-PALSAR	15
	Chiang Mai, Thailand	9 December 2015–24 December 2017	Sentinel-1	10

The area used in the Bangkok dataset is located in the northern part of Bangkok in the city of Rangsit. From the whole dataset, two areas were selected as testing areas. The first area, as shown in Figure 3a,b, is mostly rice fields areas with several large groups of villages scattered around the area. The second area has a similar characteristic with a lower number of villages but has a continuously developing, large temple (Figure 3c,d) as a landmark at the middle of the image.

(a) **(b)**

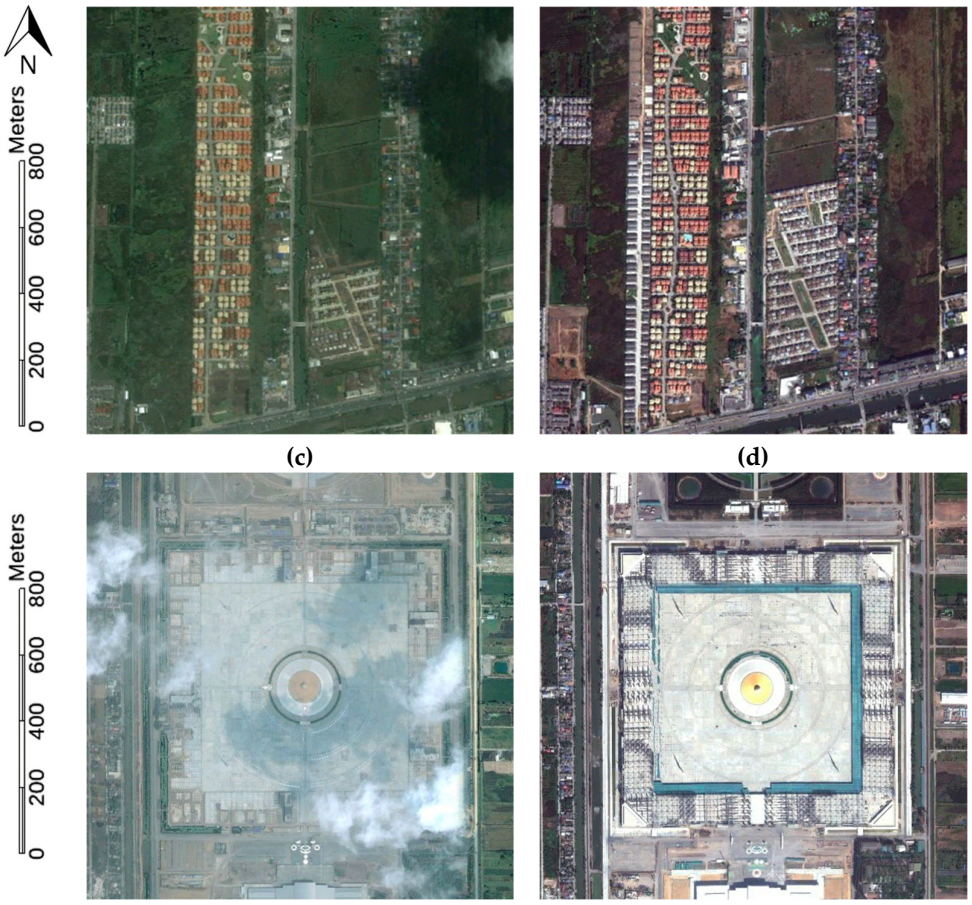


Figure 3. Optical image showing an example of new constructions of Bangkok testing area. The size of each image is 1.3 × 1.3 km. (a) Time 1 image of first testing area from 22 August 2008, (b) Time 2 image of first testing area from 18 December 2009, (c) Time 1 image of second testing area from 10 February 2005, (d) Time 2 image of second testing area from 18 December 2009.

Since the model was trained with the Bangkok area, we picked Hanoi and Xiamen for testing the applicability of the model to other areas. The southern part of Hanoi in the Văn Điển town (Figure 4) has a similar environment to the Bangkok area in terms of terrain and the density of resident area, but also has quite different shapes and sizes of building in the selected area, especially around the center of the image. Xiamen is also selected for testing, not only because it has been developing rapidly throughout the last decade, but because it also has water surrounding the area, which should reveal whether the model can work with areas of water or not since water areas were not included in the training data. Two areas were selected for the Xiamen area: one contains three bridges as a landmark of the area (Figure 5a,b), which the model is not supposed to detect, the another one contains the building changes on the water (Figure 5b), which the model should be able to detect.

(a) **(b)**

Figure 4. Optical image showing an example of new constructions of Hanoi testing area. The size of each image is 2 × 2 km. (a) Time 1 image from 15 November 2002, (b) Time 2 image from 9 February 2010.

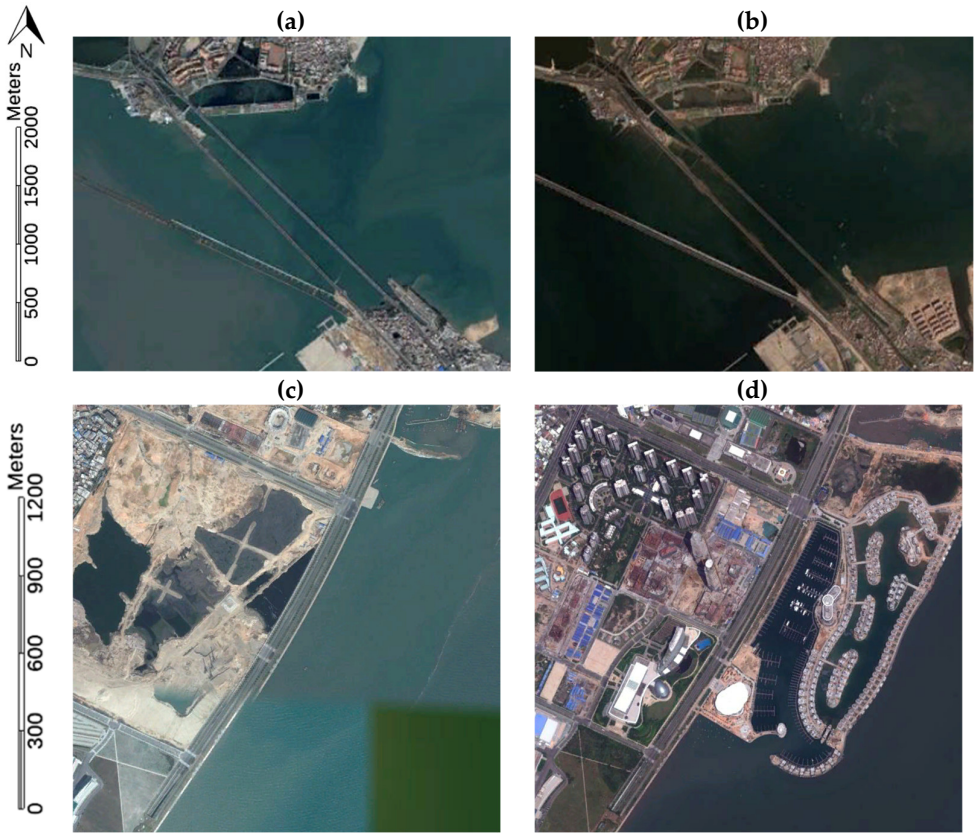


Figure 5. Optical image showing an example of new constructions of Xiamen testing area. The size of (a) and (b) image are 3.7 × 2.8 km and size of (c) and (d) image are 1.7 × 1.7 km. (a) Time 1 image of first testing area from 12 May 2006, (b) Time 2 image of first testing area from 29 October 2009, (c) Time 1 image of second testing area from 5 December 2006, (d) Time 2 image of second testing area from 17 September 2011.

Lastly, the Chiang Mai testing area viewed from the Sentinel-1 satellite was selected to test the model on images from other satellite with other acquisition conditions, and also to test the applicability of the model on mountain area. The area in question is at Doi Suthep mountain; as seen in Figure 6, half of the image is mountain and another half is mainly scattered with small houses.

(a) **(b)**

Figure 6. Optical image showing an example of the new constructions in the Chiang Mai testing area. The size of each image is 3.77 km × 3.97 km. (a) Time 1 image from 17 November 2015, (b) Time 2 image from 24 December 2017.

4. Experiment on Network Detail

In order to select the most appropriate setting for CORN, we conducted a number of experiments to ensure the architecture we made works properly, which will be explained in this section. First, in Section 4.1, to support our assumption on the ratio of the ordinary input set to the reverse input set to be calculated in (1), we conducted experiments to prove that 7:3 is the most suitable ratio for encoder 8 among the 6:4, 7:3, 8:2, and 9:1 ratios for the model to learn the shared features between the two input sets. In Section 4.2, an experiment on skip connection was conducted, where we tested the model trained with the architecture in Figure 1 and compared the result with the model trained with the architecture with an additional skip connection to both decoder sides, as shown in Figure 2.

Before calculating the accuracy, the final output prediction map in the range of [0, 1] from each model is turned into binary map with thresholding by 0.5, where white pixels indicate newly built construction and black pixels indicate non-change areas. The accuracy in these experiments, as well as the rest of the paper, was calculated in the form of overall accuracy, precision, recall, F measure, F1 measure, Kappa, intersect over union (IOU), false negative (FN) rate, and false positive (FP) rate. The false negative rate was obtained by the number of pixels that were in the ground truth, but not in our predicted result, multiplied by 100 and then divided by the total number of positive pixels in the ground truth. The false positive rate was the number of pixels that were not in the ground truth, but were in our predicted result, multiplied by 100 and then divided by the total number of negative pixels in the ground truth. The calculation of each validation method, excluding the false negative and false positive rates, is shown in Table 3. The TP in Table 3 stands for true positive, while TN stands for true negative. Please note that the β value of our F measure was 0.3.

Table 3. The calculation of each validation method. IOU—intersect over union; TP–true positive; TN–true negative.

Validation Method	Calculation
Overall accuracy	$\boldsymbol{Overall\ accuracy} = \frac{\boldsymbol{TP + TN}}{\boldsymbol{TP + TN + FP + FN}}$
Precision	$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP + FP}}$
Recall	$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP + FN}}$
F measure	$\boldsymbol{F_\beta} = (\mathbf{1} + \boldsymbol{\beta^2}) \cdot \frac{\boldsymbol{precision \cdot recall}}{(\boldsymbol{\beta^2 \cdot precision}) + \boldsymbol{recall}}$
F1 measure	$\boldsymbol{F_1} = \mathbf{2} \cdot \frac{\boldsymbol{precision \cdot recall}}{\boldsymbol{precision + recall}}$
Kappa	$\boldsymbol{Kappa} = \frac{\boldsymbol{Observed\ agreement - chance\ agreement}}{\mathbf{1} - \boldsymbol{chance\ agreement}}$
IOU	$\boldsymbol{IoU} = \frac{\boldsymbol{target} \cap \boldsymbol{prediction}}{\boldsymbol{target} \cup \boldsymbol{prediction}}$

4.1. Experiment on Encoder 8 Feature Sharing Ratio

To compare the encoder 8 ratio, we tested each model with the Bangkok site. The results of the first testing area from the SAR pair of 12 January 2009/21 November 2009 are displayed in Figure 7, as it is the easiest with which to notice the difference. The buildings tend to be detected less when the influence from the opposite side of encoder 8 is smaller, as seen in Figure 7d, at a ratio of 9:1. In contrast, the 6:4 ratio in Figure 7a includes too many, too-large buildings in the detection result, since it experiences more influence from the opposite encoder 8. The 7:3 and 8:2 ratios have similar detection results, but 7:3 was chosen for our work since it works significantly better in reducing the false positive rate, as shown in Table 4.

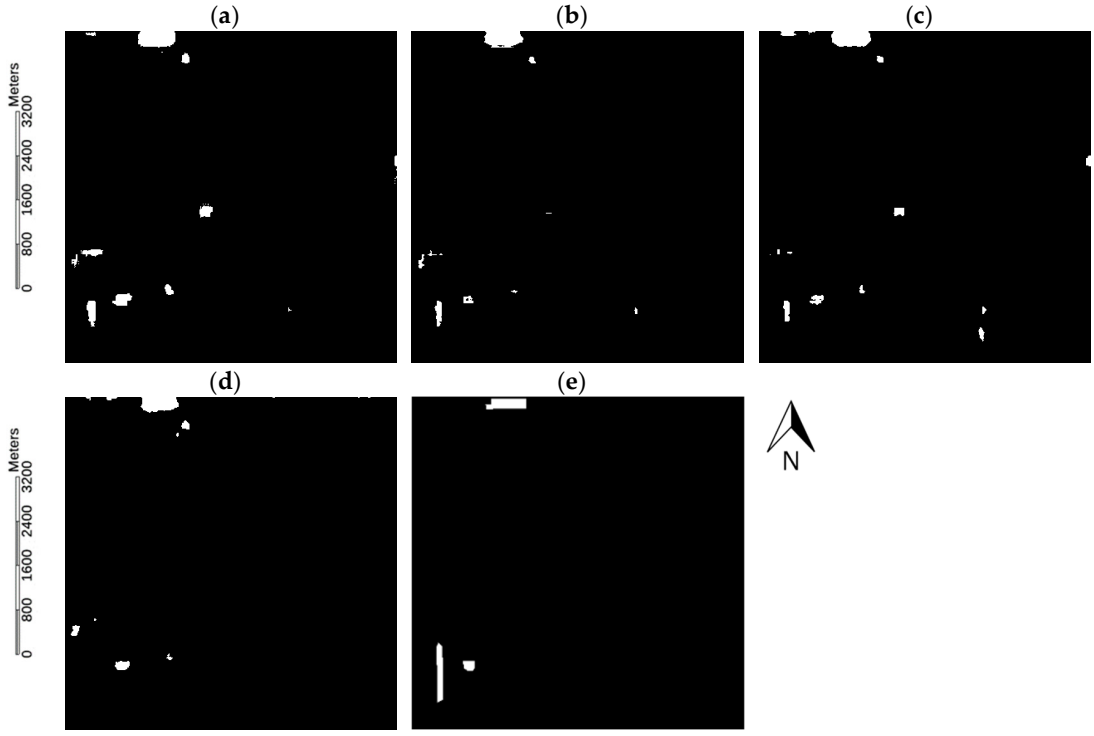


Figure 7. Results of the Bangkok site in the first area of the SAR pair of 12 January 2009/21 November 2009, where the size of each image is 6 × 6 km: (**a**) the encoder 8 portion is 6:4, (**b**) the encoder 8 portion is 7:3, (**c**) the encoder 8 portion is 8:2, (**d**) the encoder 8 portion is 9:1, (**e**) ground truth. 14°1′2.26″N 100°41′15.99″E.

Table 4. Accuracy of the model in the different encoder 8 portions at the Bangkok site.

Validation Method	6:4	7:3	8:2	9:1
False negative	47.368	54.667	55.739	59.621
False positive	0.601	0.210	0.391	0.285
Overall accuracy	98.928%	99.791%	99.614%	99.717%
Precision	0.471	0.687	0.535	0.590
Recall	0.526	0.453	0.442	0.404
F measure	0.475	0.659	0.526	0.568
F1 measure	0.497	0.546	0.484	0.479
Kappa	0.492	0.543	0.480	0.475
IOU	0.331	0.376	0.320	0.315

4.2. Experiment on the Addition of a Skip Connection in the Network

The second area of the Bangkok site from the SAR pair of 12 January 2009/21 November 2009 was used to display the difference between sending the added features with skip connection to one side of the decoder (Figure 1), and to both side of the decoders (Figure 2) in Figure 8.

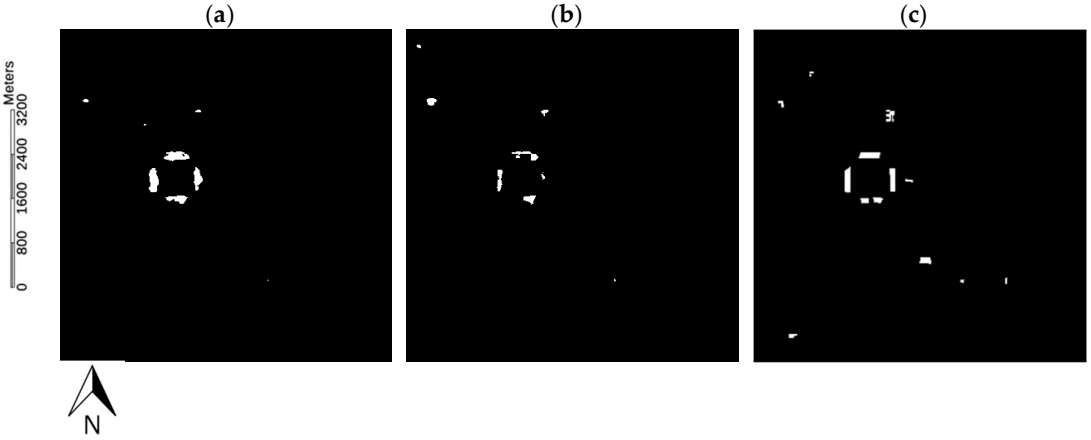


Figure 8. Results of the Bangkok site in the second area of the SAR pair of 12 January 2009/21 November 2009, where the size of each image is 6 × 6 km: (**a**) Additional skip connection on one side of the network; (**b**) Additional skip connection on both sides of the network, (**c**) ground truth.

As we stated earlier, the shape of a detected building is too limited to the boundary information sent by the addition of encoders when the addition of the skip connection is applied to both sides of the network, as can be observed by the square-shape-like building change at the center of Figure 8b. As a result, the false negative rate increases, as shown in Table 5.

Table 5. Accuracy of the model in the different skip connections in the architecture at the Bangkok site.

Validation Method	Proposed Network	Additional Skip Connection on Both Sides
False negative	54.667	66.936
False positive	0.210	0.180
Overall accuracy	99.791%	99.149%
Precision	0.687	0.652
Recall	0.453	0.331
F measure	0.659	0.603
F1 measure	0.546	0.439
Kappa	0.543	0.435
IOU	0.376	0.281

5. Experiment on Testing Sets and Discussion

In this section, we used the CORN-trained model to compare with our previous work, which was the newly built construction detection model trained with original U-net architecture. The reason we chose the comparison with U-net is because it is proven to be the most effective method compared to a number of conventional methods, such as the fully convolutional network, fuzzy c-mean, and Otsu thresholding [29–31]. The training and validation conditions between the CORN and U-net models were all the same in every aspect to make the comparison as fair as possible: the same training set (three Bangkok datasets), same patch size (256 × 256 pixels), same loss function (weighted binary cross entropy at ω_p = 181.5), same epoch number (10 epoch), and same testing images (Bangkok, Hanoi, and Xiamen) were used. The testing sites, which were Bangkok (same as in Section 4.1 and 4.2 experiments), Hanoi, and Xiamen, were from various time points between 22 January 2007–13 February 2011. As per the training set, a Lee filter sized 3 × 3 and normalization were applied to all images. In addition to the previous testing set, we added one more testing area—that is, an image from the Sentinel-1 satellite in Chiang Mai, Thailand—to demonstrate how CORN works against the different image setting.

5.1. Bangkok Testing Set

The same area as in [13] was selected for testing the model in Bangkok, which was the same city chosen in the training data. The results are shown in Figures 9 and 10. Although multiple testing areas were tested, we selected one area to show for ease of inspection. The pixel number of the testing area was 640,000, including 6439 positive pixels and 633,561 negative pixels in ground truths.

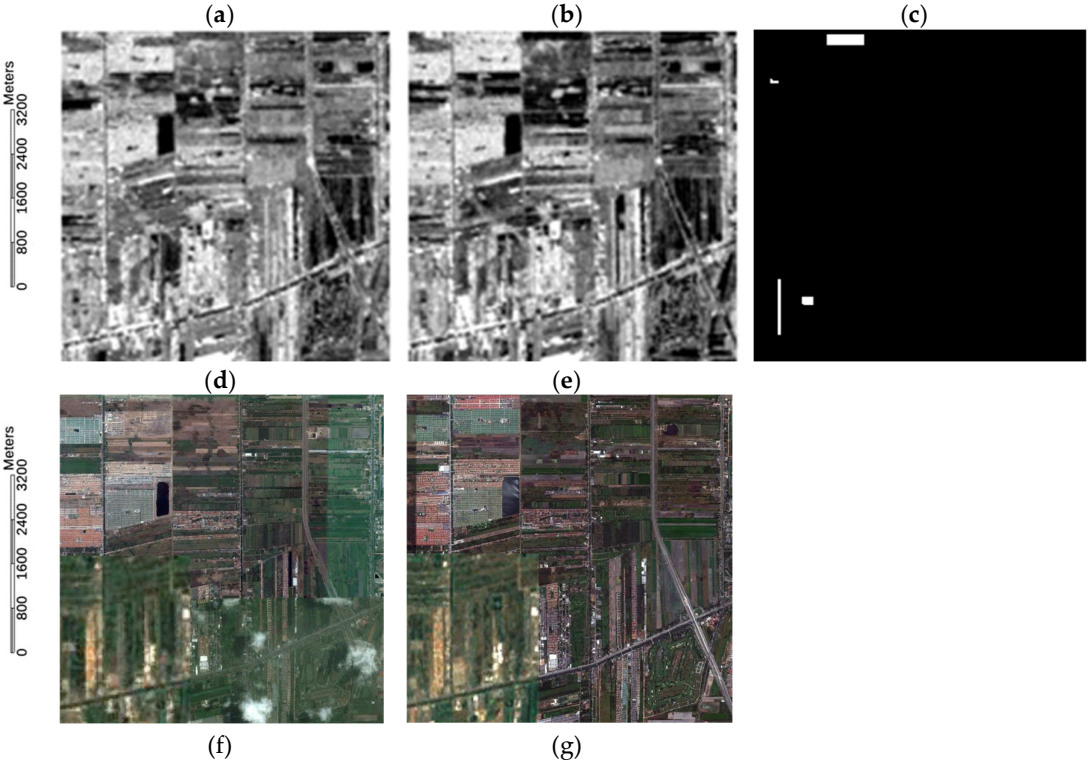


Figure 9. Results of the Bangkok site in the first area at 14°1′2.26″N 100°41′15.99″E. The size of each image is 6 × 6 km (for SAR pairs 27 November 2008/15 January 2010: (**a**) Time 1 SAR image; (**b**) Time 2 SAR image; (**c**) ground truth; (**d**) result of U-net; (**e**) proposed result).

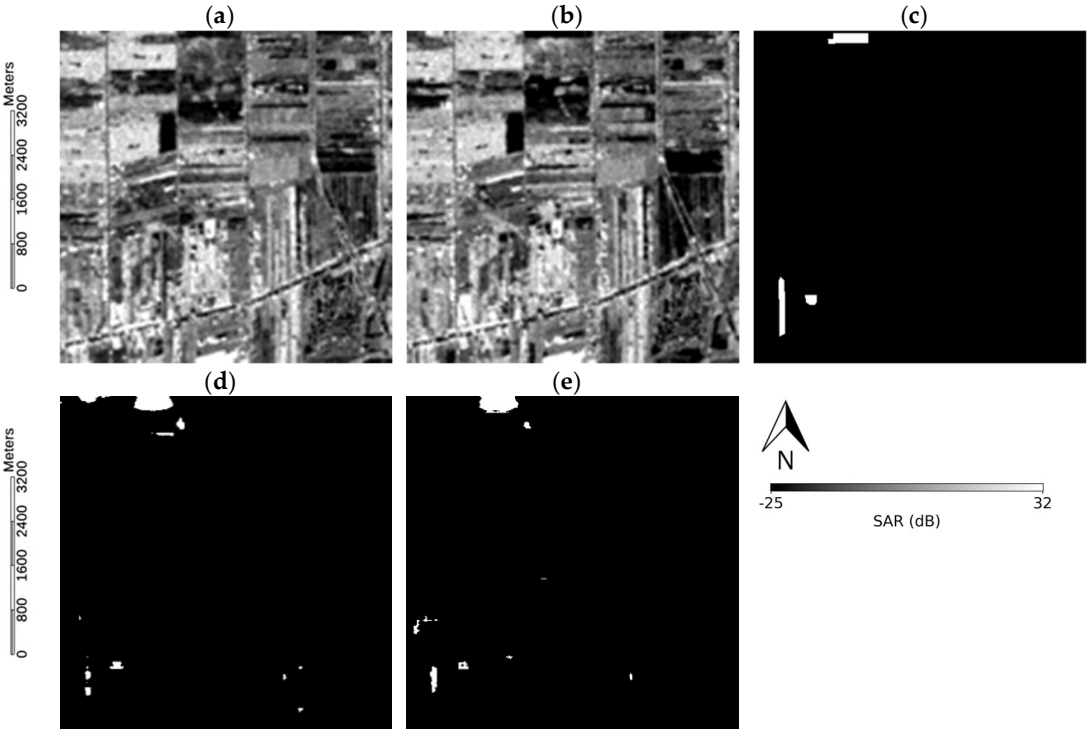


Figure 10. Results of the Bangkok site in the first area at 14°1′2.26″N 100°41′15.99″E. The size of each image is 6 × 6 km (for SAR pairs 12 January 2009/21 November 2009: (**a**) Time 1 SAR image; (**b**) Time 2 SAR image; (**c**) ground truth; (**d**) result of U-net; (**e**) proposed result).

The results were able to detect only the construction of buildings while avoiding the change caused by the season. However, the results from the proposed architecture are visually better than that of U-net, as it can detect more detailed buildings and provides more accurate shapes of buildings, thus reflecting lower false negative and false positive rate in Table 6. The new model can also detect rows of buildings at the lower left part of the image more accurately, even though it has a low intensity difference between Time 1 and Time 2 images.

Table 6. Accuracy of the model in the Bangkok area.

Validation Method	Proposed Network	U-net
False negative	54.667	55.801
False positive	0.210	0.403
Overall accuracy	99.791%	99.04%

Precision	0.687	0.527
Recall	0.453	0.442
F measure	0.659	0.519
F1 measure	0.546	0.481
Kappa	0.543	0.476
IOU	0.376	0.316

5.2. Hanoi and Xiamen Testing Sets

The comparison of the CORN and U-net results in the Hanoi testing site is shown in Figure 11 while that of the Xiamen site, which has two testing areas, is in Figures 12 and 13. The Hanoi testing set was, in total, 160,000 pixels, including 859 positive pixels and 159,141 negative pixels, and that of the Xiamen testing site was, in total, 320,000 pixels, including 4482 positive pixels and 325,518 negative pixels.

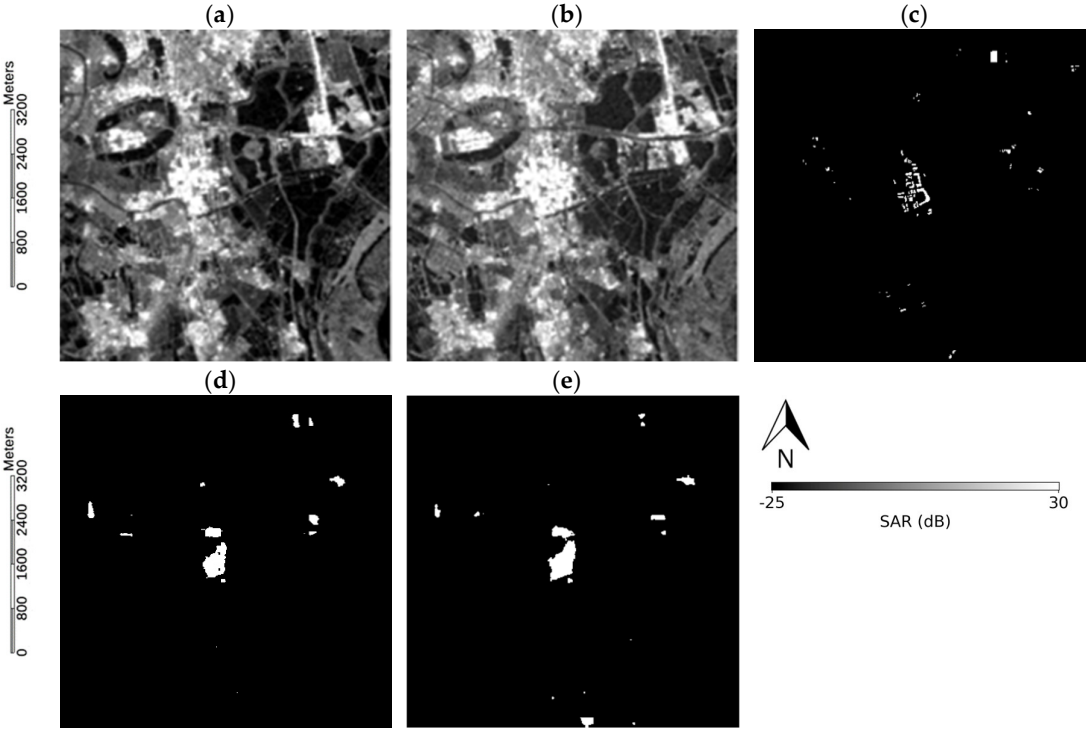
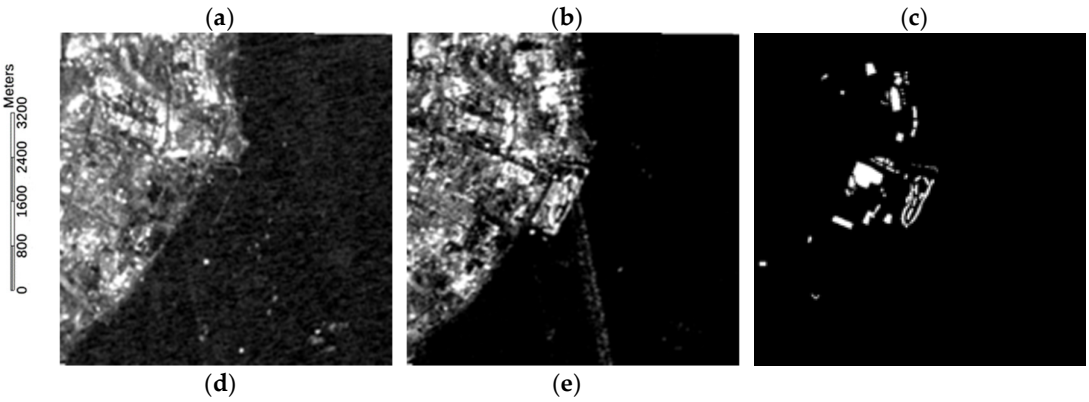


Figure 11. Result of the Hanoi site at 20°57′06.35″N 105°51′08.87″E. The size of each image is 6 × 6 km. (**a**) Time 1 SAR data; (**b**) Time 2 SAR data; (**c**) ground truth; (**d**) result of proposed model; (**e**) result of U-net.



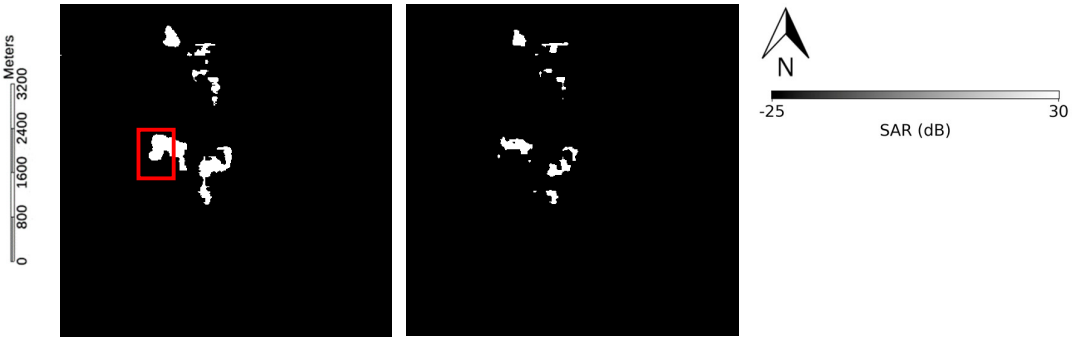


Figure 12. Result of the first Xiamen test site at 24°28′35.28″N 118°11′36.12″E. The size of each image is 6 × 6 km. (**a**) Time 1 SAR data; (**b**) Time 2 SAR data; (**c**) ground truth; (**d**) result of proposed model; (**e**) result of U-net.

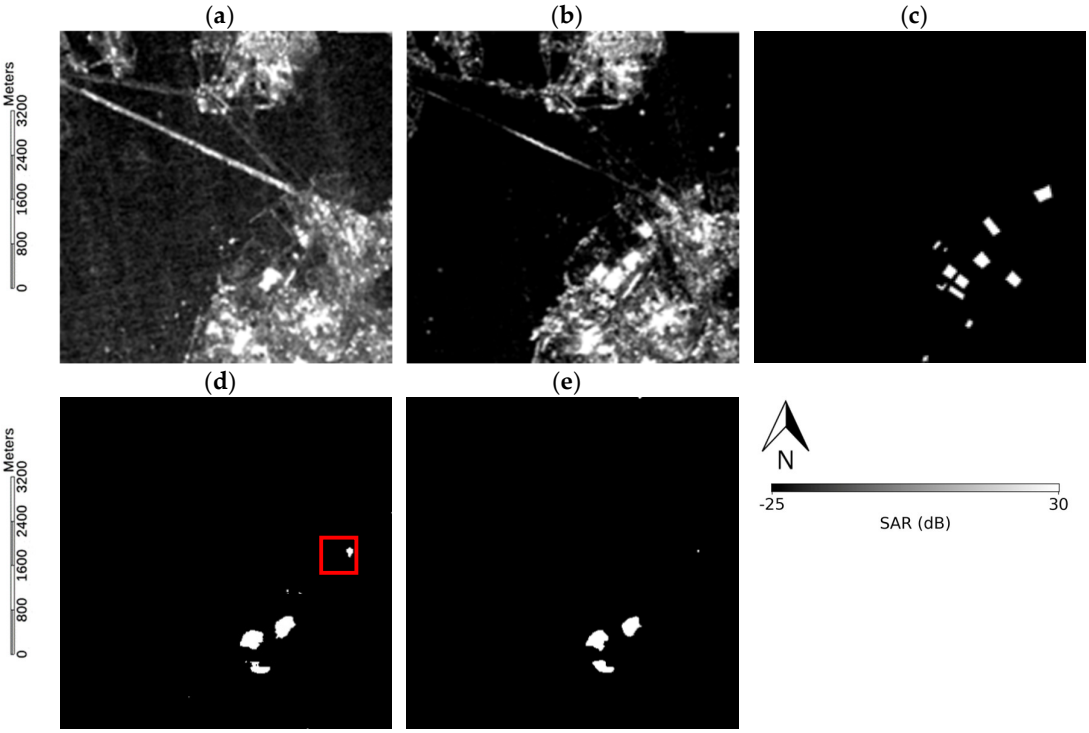


Figure 13. Result of the second Xiamen test site at 24°31′59.42″N 118°06′54.51″E. The size of each image is 6 × 6 km. (**a**) Time 1 SAR data; (**b**) Time 2 SAR data; (**c**) ground truth; (**d**) result of proposed model; (**e**) result of U-net.

For the Hanoi site, the accuracies of the results from the proposed network shown in Table 7 are very close to that of the U-net. Since the constructions that occurred in this dataset were mainly of small buildings, our model tried to generate the shapes of the changes as accurately as possible, which led to very small detection results in some areas—so much so that some detected objects appeared in very few pixels or were even omitted, as can be especially seen in the bottom half of Figure 11d. As a result, the false negative rate increased in our results, which made that of recall, F1 measure, Kappa, and IOU slightly lower than in U-net. In contrast, while several objects were detected by U-net in the bottom half of Figure 11e, they did not all correlate with those in the ground truths, causing U-net to have more false positive values than CORN. The lower false positive rate of our model resulted in higher overall accuracy, precision, and F measure rates than U-net.

Table 7. Accuracy of the model in the Hanoi area.

Validation Method	Proposed Network	U-net
False negative	62.980	58.324
False positive	0.782	0.922
Overall accuracy	99.522%	98.77%
Precision	0.204	0.196
Recall	0.370	0.417
F measure	0.211	0.205
F1 measure	0.263	0.267
Kappa	0.258	0.261
IOU	0.151	0.154

The accuracies of Xiamen are shown in Table 8.

Table 8. Accuracy of the model in the Xiamen area.

Validation Method	Proposed Network	U-net
False negative	68.652	77.577
False positive	0.861	0.508
Overall accuracy	98.189%	98.412%
Precision	0.341	0.385
Recall	0.313	0.224
F measure	0.338	0.364
F1 measure	0.327	0.283
Kappa	0.317	0.276
IOU	0.195	0.165

Xiamen is a city surrounded by water, which is an area type that the training data did not include. Unlike with Hanoi, the results for Xiamen from our model achieved better accuracies over U-net in recall, F1 measure, Kappa, and IOU, because of the reduction in the false negative rate. This reduction was a result of a better detection rate for constructions, especially in building boundaries, as highlighted in the red rectangles in Figures 12d and 13d where the U-net can only detect as a small group of pixels. Please note that although CORN used information from both Time 1–Time 2 and Time 2–Time 1 formats, it can also avoid detecting a noise in the SAR image, displayed as a faded line from the center to the bottom in Figure 12b, compared to U-net that only uses change information in the Time 1–Time 2 format.

5.3. Sentinel-1 SAR Image Testing Set

Past experiments show that the current model trained with images of Bangkok city can be used with other areas viewed from the same satellite. However, we wanted to show that it can also be used with SAR images from other satellites too. We tested the model with a C-band SAR image from the Sentinel-1 satellite, while an image from the ALOS-PALSAR training data was captured in L-band. Other properties were also different from those in the training data in many aspects; for instance, the resolution was 10 m/pixel and the polarization was VV. The selected area was Chiang Mai in the northern part of Thailand, where most of the area is mountainous, while Bangkok, the city used in the training of the model, comprises mostly plain areas. Some parts of the detection results were cropped and are shown in Figures 14–16. As this area was an additional area to the previous work, the ground truth of this area was created, and thus, the validation was done by visual comparison with optical images, since accuracies cannot be calculated and shown in terms of numbers. The date of the Time 1 optical images in Figures 14 and 15 was 7 January 2016, while in Figure 16, it was 17 November 2015, due to the cloud cover problem. The Time 2 optical images in Figures 14 and 15 were from 29 October 2017; while in Figure 16, they were from 24 December 2017 due to the availability of the existing data. Please note that all of the optical images in this experiment were selected from Google Earth software, where images were captured by a variety of satellites and aircraft, meaning

it is difficult to determine the image source. However, according to the rough data provided by the software, some images were captured with Landsat 7 at a 30 m/pixel resolution.

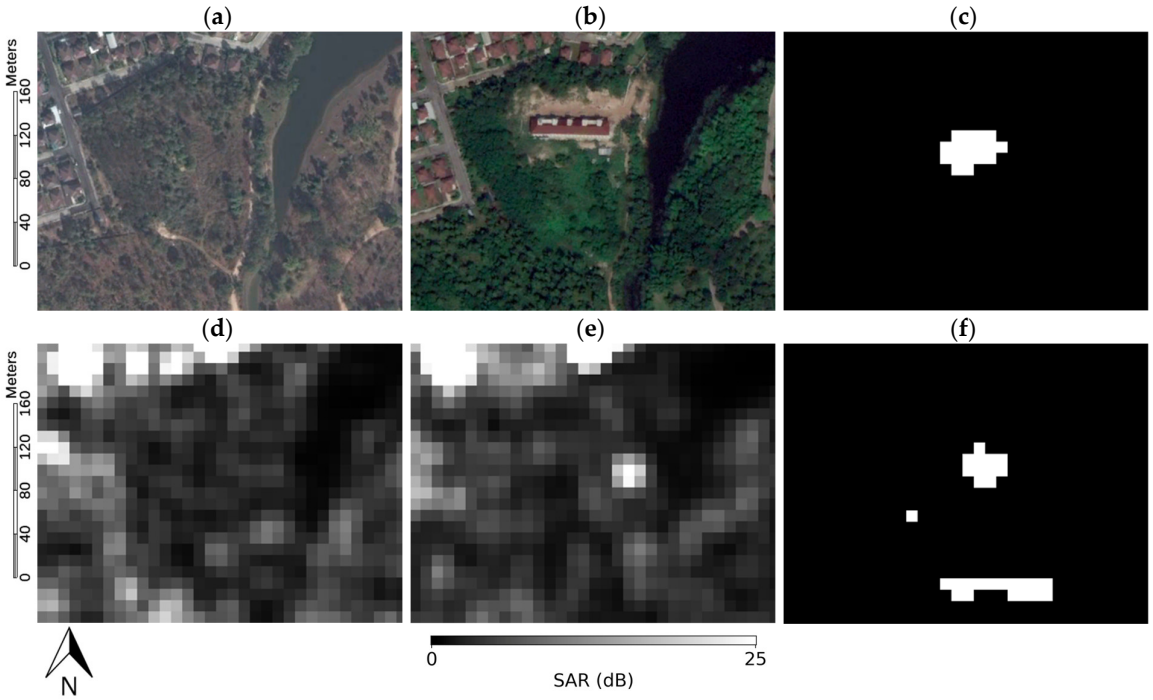


Figure 14. Detection results of the first area in Chiang Mai at 18°51′23.49″N 98°57′17.90″E. The size of each image is 0.32 × 0.23 km. (**a**) Time 1 optical data; (**b**) Time 2 optical data; (**c**) result of CORN; (**d**) Time 1 SAR data 'Copernicus Sentinel data [2015]'; (**e**) Time 2 SAR data 'Copernicus Sentinel data [2017]'; (**f**) result of U-net.

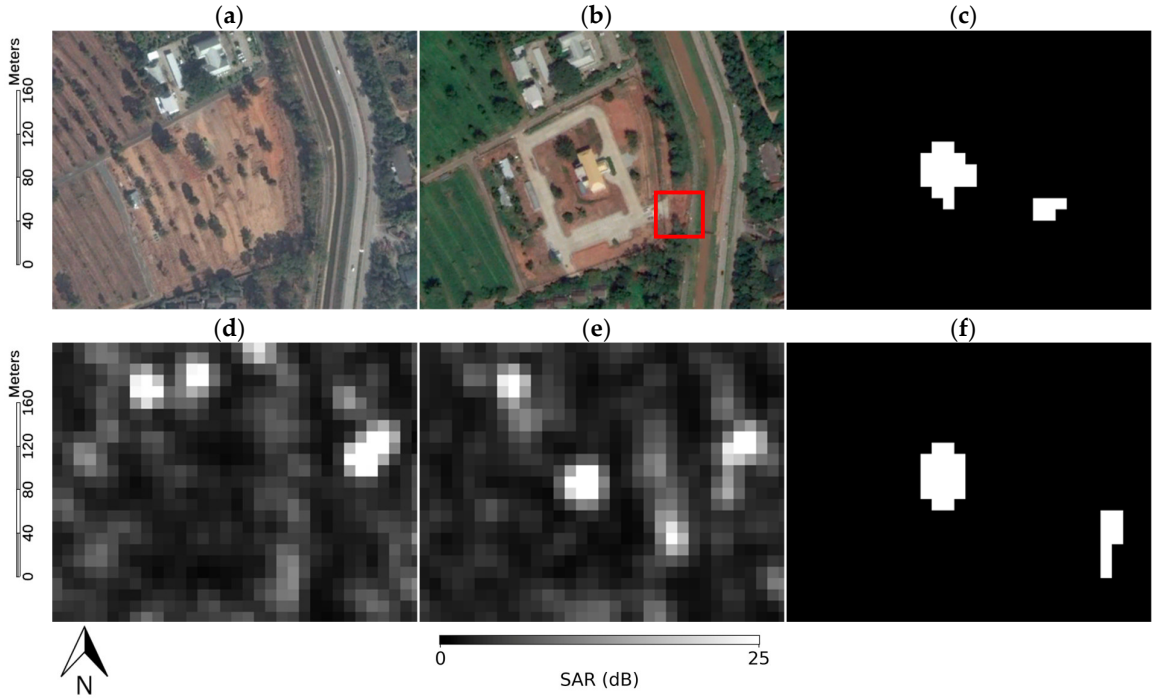


Figure 15. Detection results of the second area in Chiang Mai at 18°51′22.36″N 98°57′40.70″E. The size of each image is 0.32 × 0.23 km. (**a**) Time 1 optical data; (**b**) Time 2 optical data; (**c**) result of CORN; (**d**) Time 1 SAR data 'Copernicus Sentinel data [2015]'; (**e**) Time 2 SAR data 'Copernicus Sentinel data [2017]'; (**f**) result of U-net.

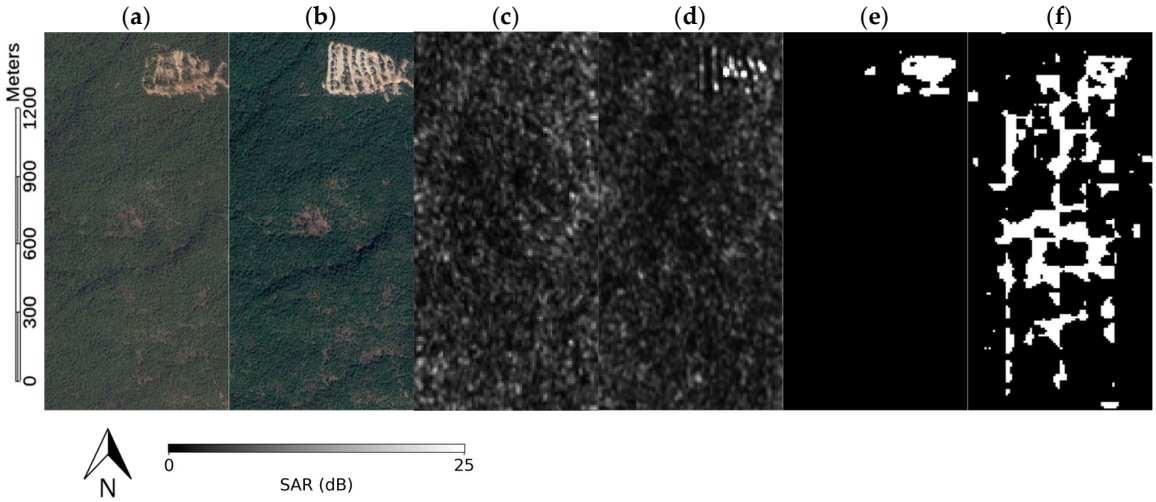


Figure 16. The result of the proposed model with an image from Sentinel-1 at 18°50′48.34″N 98°56′55.95″E. The size of each image is 0.8 × 1.75 km. (**a**) Time 1 optical data; (**b**) Time 2 optical data; (**c**) Time 1 SAR data 'Copernicus Sentinel data [2015]'; (**d**) Time 2 SAR data 'Copernicus Sentinel data [2017]'; (**e**) result of CORN; (**f**) result of U-net.

Although both our proposed network and U-net can detect new constructions, the results indicate that U-net generated more false positive results than CORN. In Figure 14, CORN correctly detected building change without any false detection, while U-net mistakenly detected forest area in the bottom part of Figure 14d. Please note that the high intensity spot in the middle of Figure 14e is not the building. In Figure 15, both CORN and U-net show two detected buildings in their results. They both correctly detected a building in the center of the image. However, another object that U-net detected was an existing road in the right side of Figure 15d, which was a false detection, while the false detection in CORN is the area in the red rectangle in Figure 15b. From the selected Time 2 optical image from 29 October 2017, it is difficult to see what CORN detected in the area, but in the optical image from 3 March 2018 (Figure 17), there is a bridge placed next to CORN's detected area. Thus, it can be assumed that the other object detected by CORN was a bridge under construction, since our Time 2 SAR image was taken on 24 December 2017, which was around the middle of the dates that these two optical images were taken.

Figure 17. Optical image of the second area in Chiang Mai on 3 March 2018 at 18°51′22.36″N 98°57′40.70″E.

In Figure 16, even though U-net was also able to detect construction in Sentinel-1 data, as seen in the top right corner area, it failed to handle data containing changes in mountain areas and ended up involving them in the detection result instead. On the other hand, CORN was able to avoid the

intensity change of mountain areas and detected only the building changes. This experiment suggests that by combining the training of ordinary and reverse time-series data, our model can eliminate a greater variety of false positives, which means it can be used even with images from other satellites.

5.4. Other Experiments

It is also worth mentioning that we tried to randomly reduce the number of training sets from 2028 pairs to 1500 pairs and 500 pairs, respectively, to observe the learning capability of both CORN and U-net in slightly lower training set situations and very low training set situations. For each number of training sets, the networks were trained four times with four different randomly selected training pairs, and then tested with the Bangkok testing site. The results of this experiment are shown in Table 9 as the averages of four times the testing results. As expected, in the case of 1500 training pairs, the accuracies of both CORN and U-net dropped from when trained with 2028 pairs, but CORN still surpassed U-net, except in false negatives and recall. The use of 500 training pairs indicates that U-net cannot be trained with a very small dataset, as is reflected in the very low accuracies. While the accuracies of CORN were relatively low, they were still in the acceptable range, which means that the network is able to learn even with a very small training set. This result supports our assumption that learning features from two formats of bitemporal data helps the network to become better at detecting newly built constructions. The time taken by CORN in training 1500 pairs was 53 min, whereas for U-net, it was 48 min. For the single training of 500 pairs, CORN spent 18 min and U-net spent 15 min.

Table 9. Accuracies of the models in the different number of training data at the Bangkok site.

Validation Method	500 Pairs		1500 Pairs	
	CORN	**U-net**	**CORN**	**U-net**
False negative	40.860	27.512	53.254	47.232
False positive	1.467	17.970	0.383	0.683
Overall accuracy	98.136%	81.934%	99.085%	98.848%
Precision	0.310	0.100	0.590	0.492
Recall	0.591	0.725	0.467	0.528
F measure	0.321	0.107	0.571	0.488
F1 measure	0.400	0.165	0.507	0.485
Kappa	0.391	0.150	0.503	0.480
IOU	0.251	0.092	0.340	0.321

5.5. Result Discussion

Most of the experiments indicate that CORN can detect new constructions while avoiding the detection of other changes as seen in Figure 18–20. With the advantage of training using both Time 1–Time 2 and Time 2–Time 1, the use of CORN resulted in more precise detection at the edges of buildings, resulting in improved accuracies for almost all the tested datasets. On visual inspection, it can be seen that the edges of the detected buildings were more similar to the ground truth and had less false positive detections than those achieved with U-net, as shown in Figure 21. In terms of accuracy, both false positives and false negatives were mostly dropped from the previous work, which led to an increase in F measure, F1 measure, Kappa, and IOU.

(a) **(b)** **(c)**

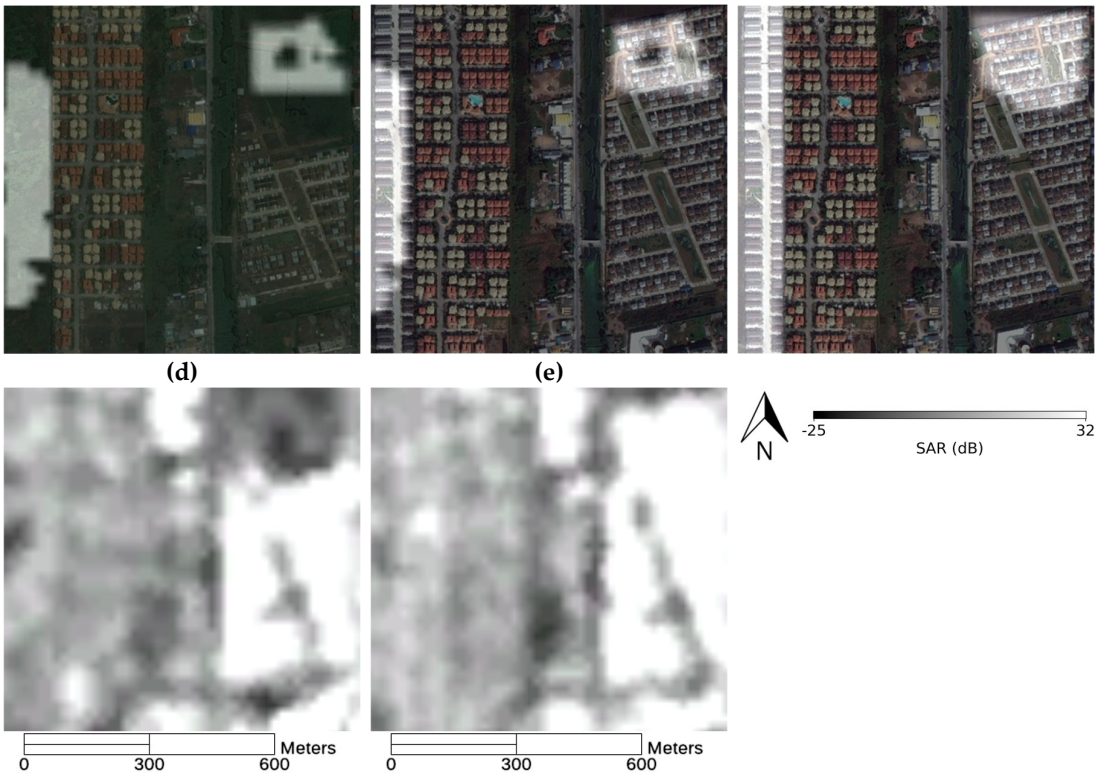


Figure 18. Example of detection results of CORN at Bangkok testing site for SAR pairs 12 January 2009/21 November 2009. (a) CORN result overlays on Time 1 optical image, (b) CORN result overlays on Time 2 optical image, (c) ground truth overlays on Time 2 optical image, (d) Time 1 SAR image, (e) Time 2 SAR image.

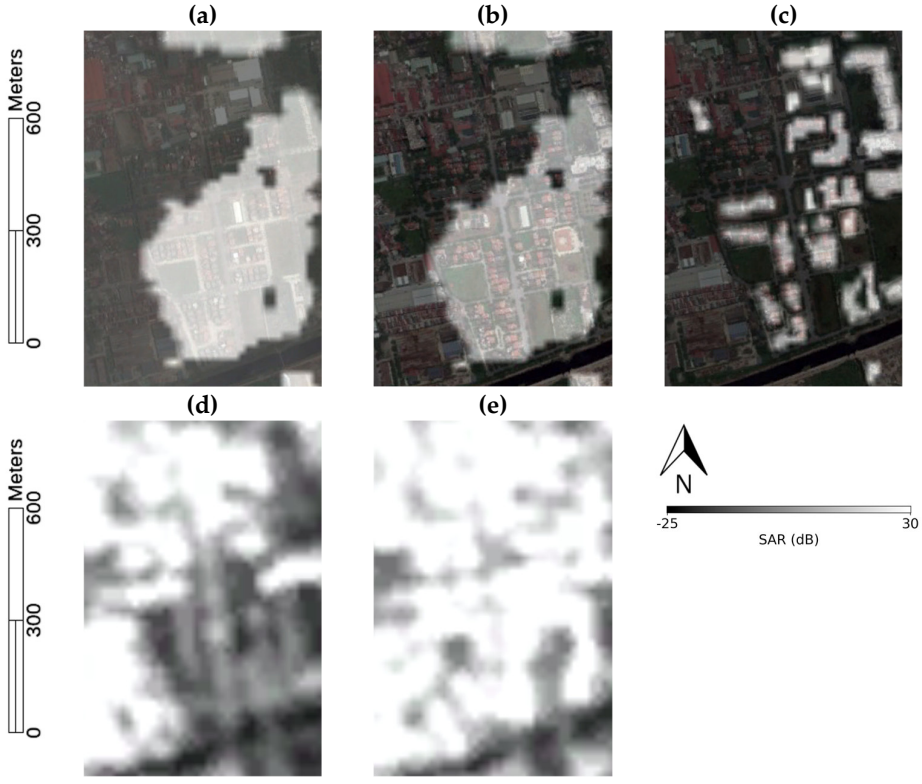


Figure 19. Example of detection results of CORN at Hanoi testing site. (a) CORN result overlays on Time 1 optical image, (b) CORN result overlays on Time 2 optical image, (c) ground truth overlays on Time 2 optical image, (d) Time 1 SAR image, (e) Time 2 SAR image.

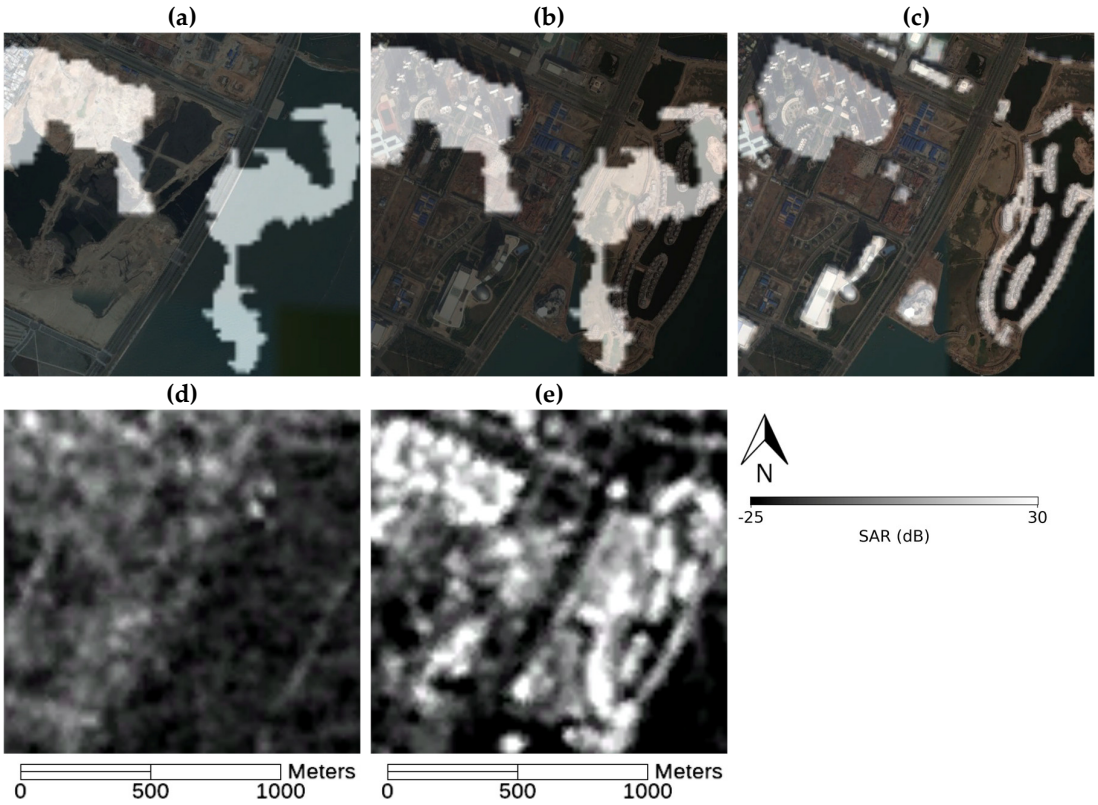


Figure 20. Example of detection results of CORN at Xiamen testing site. (a) CORN result overlays on Time 1 optical image, (b) CORN result overlays on Time 2 optical image, (c) ground truth overlays on Time 2 optical image, (d) Time 1 SAR image, (e) Time 2 SAR image.

Despite the improved accuracy, some areas, especially the Hanoi site, still have a relatively high false negative rate. This is due to the fact that most of the constructions in the training data have a larger size than those in the Hanoi area, and the construction shapes are also way too different from each other. As a result, the model failed to detect some of the constructions and caused the high false negative rate in such areas. To tackle this problem, supplementing the training data with various sizes and shapes of buildings could be very helpful.

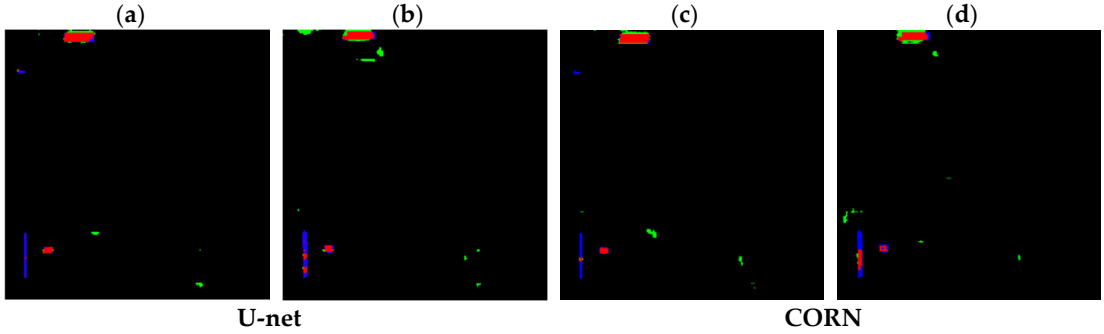


Figure 21. Comparison of the proposed model's result and the ground truth of the Bangkok testing site: (**a**) result of U-net for the SAR pair of 27 November 2008/15 January 2010; (**b**) result of U-net for the SAR pair of 12 January 2009/21 November 2009; (**c**) result of CORN for the SAR pair of 27 November 2008/15 January 2010; (**d**) result of CORN for the SAR pair of 12 January 2009/21 November

2009 ((red—true positive area; green—false positive area, blue—false negative area). The size of each image is 6 × 6 km.

Further evidence that shows the benefit of the more robust detection provided by our model is the result of using Sentiel-1 satellite. While the area in Figure 14, which was captured in a completely different setting to those in the training data, involved a lot of intensity change (as shown in Figure 22), CORN was able to detect the building correctly and avoided seasonal changes even though they have a similar intensity of change to buildings, especially the bright spot in the middle of the image which is brighter than the actual building, making it visually very similar to the building change. Although U-net was also able do the same thing, it still detected many incorrect changes in the image. For the same area, Figure 16 is the best example, showing that CORN is robust against changing terrain, as it can avoid changes in mountains while U-net cannot. Please note that in Figure 14, the detection results for both networks appear to shift to the south-east from the real building location in the optical image. This is probably because the intensity of the building in the SAR image was too low to display obvious features, and therefore it was difficult for the models to detect it in its exact position. The reason this building has a lower intensity than the surrounding buildings is because its roof shape and orientation are different from the other buildings. The high intensity of surrounding buildings is possibly the result of the double bounce on the walls or a strong single bounce on the roofs, which did not happen with the detected building due to the reason stated above.

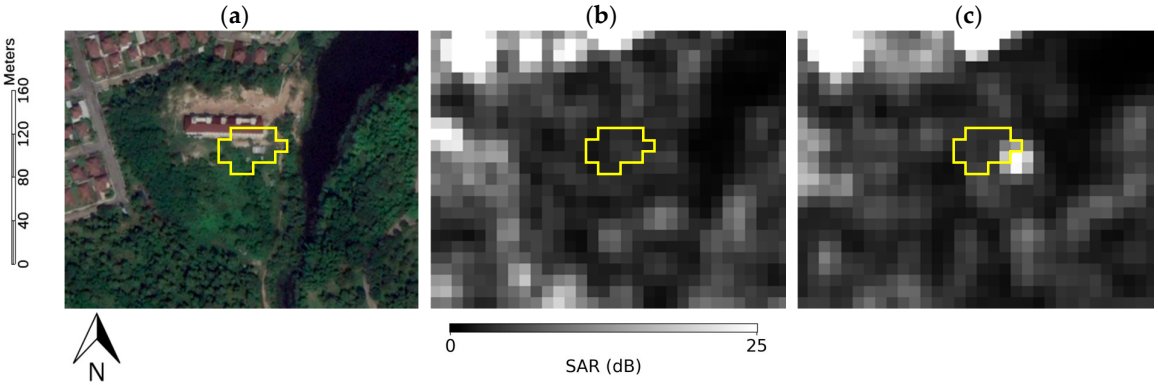


Figure 22. Detection results of CORN in the first area of Chiang Mai at 18°51′23.49″N 98°57′17.90″E in yellow hollowed polygon overlays on (**a**) a Time 2 optical images; (**b**) a Time 1 SAR image 'Copernicus Sentinel data [2015]' and (**c**) a Time 2 SAR image 'Copernicus Sentinel data [2017]'.

Since CORN achieved benefits by training on both ordinary and reverse datasets, it was inevitable that it would take a longer time than U-net in training. Still, although the training time was longer than U-net, it was not by much, or was even shorter when the training data was reduced. In fact, a decrease in the training set size in CORN can shorten the training time, as stated in Section 5.4, while the accuracies only slightly dropped. It is worth noting that, while the accuracy of results when training 1500 pairs with CORN were higher than when training 2028 pairs in U-net in every way, the training time was lower than training U-net with 2028 pairs (53 min and 55 min, respectively). As a result, we believe this architecture can replace U-net for detecting constructions in time-series SAR images with the right amount of training data.

6. Conclusions

In this research, we proposed "CORN", a new deep-learning architecture for newly built construction detection using bitemporal SAR time-series images. The architecture consists of two U-net bases for the network to learn differences—both forward and backward—by training it using Time 1–Time 2 and Time 2–Time 1 data. The features between the two U-net adaptations are shared through encoder 8, and the addition of encoders before feeding to decoders via skip connection. The detection results of Bangkok, Hanoi, and Xiamen show that the new model can achieve higher accuracies than the U-net model, without having to use more training data or ground truths. The

results also suggest that the new model can be used with images from other SAR satellites, as it can detect some changes without much false detection in Sentinel-1 L-band images after having been trained with C-band SAR images from ALOS-PALSAR.

Author Contributions: R.J. proposed the method, conducted the experiments, and wrote the manuscript. P.R. provided opinions on methodology and experiments. M.M. improved the structure of the manuscript and provided information of SAR data. R.N. provided the dataset and corresponding information, and supplied the server for conducting the experiments. All authors read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the grant-in-aid for scientific research (KAKENHI No.: 17H02050).

Acknowledgments: The authors would like to thank ESA (European Space Agency) for the Sentinel-1 data, METI (Ministry of Economy, Trade and Industry) for ALOS-PALSAR data, and the Remote Sensing Society of Japan for the support of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ma, L.; Liu, Y.; Zhang, X.; Ye, Y.; Yin, G.; Johnson, B.A. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS J. Photogramm. Remote Sens.* **2019**, *152*, 166–177.
2. Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.P.; Shyu, M.L.; Chen, S.C.; Iyengar, S.S. A survey on deep learning: Algorithms, techniques, and applications. *ACM Comput. Surv. (CSUR)* **2019**, *51*, 92.
3. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Inter-Vention, Munich, Germany, 5–9 October 2015; pp. 234–241.
4. Sun, C.; Shrivastava, A.; Singh, S.; Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 843–852.
5. Elachi, C. *Spaceborne Radar Remote Sensing: Applications and Techniques*; IEEE Press: New York, NY, USA, 1988; p. 285.
6. Tyo, J.S.; Goldstein, D.L.; Chenault, D.B.; Shaw, J.A. Review of passive imaging polarimetry for remote sensing applications. *Appl. Opt.* **2006**, *45*, 5453–5469.
7. Rogan, J.; Chen, D. Remote sensing technology for mapping and monitoring land-cover and land-use change. *Prog. Plan.* **2004**, *61*, 301–325.
8. Shalaby, A.; Tateishi, R. Remote sensing and GIS for mapping and monitoring land cover and land-use changes in the Northwestern coastal zone of Egypt. *Appl. Geogr.* **2007**, *27*, 28–41.
9. Dewan, A.M.; Yamaguchi, Y. Land use and land cover change in Greater Dhaka, Bangladesh: Using remote sensing to promote sustainable urbanization. *Appl. Geogr.* **2009**, *29*, 390–401.
10. Balogun, I.A.; Adeyewa, D.Z.; Balogun, A.A.; Morakinyo, T.E. Analysis of urban expansion and land use changes in Akure, Nigeria, using remote sensing and geographic information system (GIS) techniques. *J. Geogr. Reg. Plan.* **2011**, *4*, 533–541.
11. Du, S.; Zhang, Y.; Zou, Z.; Xu, S.; Chen, S. Automatic building extraction from LiDAR data fusion of point and grid-based features. *ISPRS J. Photogramm. Remote Sens.* **2017**, *130*, 294–307.
12. Ban, Y.; Yousif, O. Multitemporal Spaceborne SAR Data for Urban Change Detection in China. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2012**, *5*, 1087–1094.
13. Jaturapitpornchai, R.; Matsuoka, M.; Kanemoto, N.; Kuzuoka, S.; Ito, R.; Nakamura, R. Newly Built Construction Detection in SAR Images Using Deep Learning. *Remote Sens.* **2019**, *11*, 1444.
14. Yang, M.; Jiao, L.; Liu, F.; Hou, B.; Yang, S. Transferred Deep Learning-Based Change Detection in Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6960–6973.
15. Daudt, R.C.; Le Saux, B.; Boulch, A.; Gousseau, Y. Urban change detection for multispectral earth observation using convolutional neural networks. In Proceedings of the IGARSS 2018–2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 2115–2118.
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.

17. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 850–865.
18. Xu, Y.; Wu, L.; Xie, Z.; Chen, Z. Building Extraction in Very High Resolution Remote Sensing Imagery Using Deep Learning and Guided Filters. *Remote Sens.* **2018**, *10*, 144.
19. Benedek, C.; Descombes, X.; Zerubia, J. Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 33–50.
20. Shi, W.; Mao, Z.; Liu, J. Building area extraction from the high spatial resolution remote sensing imagery. *Earth Sci. Inform.* **2019**, *12*, 19–29.
21. Konstantinidis, D.; Argyriou, V.; Stathaki, T.; Grammalidis, N. A modular CNN-based building detector for remote sensing images. *Comput. Netw.* **2020**, *168*, 107034.
22. Kwan, C.; Ayhan, B.; Larkin, J.; Kwan, L.; Bernabé, S.; Plaza, A. Performance of Change Detection Algorithms Using Heterogeneous Images and Extended Multi-attribute Profiles (EMAPs). *Remote Sens.* **2019**, *11*, 2377.
23. Zhang, P.; Gong, M.; Su, L.; Liu, J.; Li, Z. Change detection based on deep feature representation and mapping transformation for multi-spatial-resolution remote sensing images. *ISPRS J. Photogramm. Remote Sens*. **2016**, *116*, 24–41.
24. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
25. Haixiang, G.; Yijing, L.; Shang, J.; Mingyun, G.; Yuanyue, H.; Bing, G. Learning from class-imbalanced data: Review of methods and applications. *Expert Syst. Appl*. **2017**, *73*, 220–239.
26. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. *arXiv* **2016**, arXiv:1611.07004.
27. Lee, J.S. Speckle analysis and smoothing of synthetic aperture radar images. *Comput. Graph. Image Process.* **1981**, *17*, 24–32.
28. Krizhevsky, A.; Hinton, G. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; University of Toronto: Toronto, ON, Canada, 2009; Volume 1, p. 7.
29. Shelhamer, E.; Long, J.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell*. **2017**, *39*, 640–651, doi:10.1109/TPAMI.2016.2572683.
30. Bezdek, J.C.; Ehrlich, R.; Full, W. FCM: The fuzzy c-means clustering algorithm. *Comput. Geosci*. **1984**, *10*, 191–203.
31. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern*. **1979**, *9*, 62–66, doi:10.1109/TSMC.1979.4310076.