

# Sentinel-1 Polarization Comparison for Flood Segmentation Using Deep Learning <sup>†</sup>

Mohammadali Abbasi, Reza Shah-Hosseini \*  and Mohammad Aghdami-Nia

School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran 1417466191, Iran; mh.abbasi7899@ut.ac.ir (M.A.); m.aghdaminia@gmail.com (M.A.-N.)

\* Correspondence: rshahosseini@ut.ac.ir

† Presented at the 4th International Electronic Conference on Geosciences, 1–15 December 2022; Available online: <https://sciforum.net/event/IECG2022>.

**Abstract:** Flood is one of the most damaging natural hazards, and timely detection of it is very important to save human lives and assess the level of damage. The occurrence of floods in cloudy weather conditions makes the use of radar-based sensors for real-time flood mapping inevitable. In the present study, the ETCI 2021 flood event detection competition dataset, organized by the NASA Advanced Concepts and Implementation Team in collaboration with the IEEE GRSS Geoscience Informatics Technical Committee, has been used. Moreover, we have utilized the U-Net and X-Net architecture as a segmentation model to map flooded regions. This study aimed to identify the optimum polarization of the Sentinel-1 satellite for flood detection. By examining and comparing the obtained results, it was observed that the VV polarization offered better results in both models. Furthermore, U-Net had a better performance than X-Net in both polarizations.

**Keywords:** flood detection; remote sensing; SAR; Sentinel-1; deep learning; U-Net

## 1. Introduction

Flooding is one of the most common and destructive natural hazards, occurring when the water level of rivers rises, and excess water flows into the dry river bed. Therefore, quick and timely flood detection is essential for saving human lives and assessing damage. This issue highlights the importance of using advanced tools to quickly and accurately identify flooded areas so that the evacuation process can be started more quickly. With the improvements in satellite technologies, remote sensing has become one of the most suitable and cost-effective ways of mapping large-scale floods. Nowadays, valuable satellite data is freely available thanks to projects such as Landsat and Sentinel. However, there is still the need for developing efficient detection systems that can extract useful information from these data. When it comes to flood mapping, active radar satellites are the best choice due to the excessive rainfall and cloudy conditions while flooding, rendering the use of optical satellites impractical. Regarding the mapping algorithm, typical SAR image processing frameworks are usually time-consuming and computationally demanding. Thus, machine learning techniques are the preferred choice to mitigate these drawbacks.

Machine learning [1,2] and deep learning [3,4] methods have been responsible for a lot of advancements in different remote sensing fields in recent years including flood mapping. Nemani et al. [5] labeled the UNOSAT dataset manually with a histogram-based method and trained the U-Net and XNet models. To improve the efficiency of the algorithm, they used the ResNet Backbone. Katyar et al. [6] used the Sen1floods11 dataset and examined two types of manual labels and weak labels, and they trained U-Net and SegNet in three modes of Sentinel-1 and Sentinel-2 images. Using the SAR data of the Sentinel-1 A/B satellite, Kim et al. [7] trained the U-Net and SegNet models. The result of this research indicated that the performance of the U-Net model was better than the SegNet model, but SegNet achieved faster run times.



**Citation:** Abbasi, M.; Shah-Hosseini, R.; Aghdami-Nia, M. Sentinel-1 Polarization Comparison for Flood Segmentation Using Deep Learning. *Proceedings* **2023**, *87*, 14. <https://doi.org/10.3390/IECG2022-14069>

Academic Editor: Deodato Tapete

Published: 22 February 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Zhang et al. [8] used a multi-source satellite dataset including the Gaofen series and Zhuhai-1 hyperspectral images and trained the U-Net model, which had a good performance for identifying and monitoring flood areas. Ghosh et al. [9] implemented U-Net and a Feature Pyramid Network (FPN), both based on the EfficientNet-B7 backbone. They evaluated the performance of the models using the Sentinel-1 images. Using several machine learning methods including MLP, SVM, and a deep neural network (DNN), Islam et al. [10] identified flood areas in SPOT-5 and radar image sets. Tanim et al. [11] evaluated the performance of supervised and unsupervised machine learning models including Random Forest, SVM, and Maximum Likelihood using Sentinel-1 satellite images.

In this paper, we present an automatic flood detection and mapping framework based on deep learning. We utilized the ETCI 2021 flood event detection competition dataset, which was collected from Sentinel-1 images in two polarizations of VV and VH. To evaluate the effect of polarization on the segmentation performance, we implemented U-Net and X-Net models and separately trained them on VV and VH images. By assessing the trends in the results of the two models, the best polarization can be determined.

## 2. Study Area and Dataset

### 2.1. Study Area

The dataset in question was collected from three different regions of Nebraska in the center of the United States, Alabama in the southeast of the United States, and Bangladesh in the southeast of Asia under different conditions. Each one of these regions contained 12, 16, and 3 full-frame images, respectively. Moreover, the images were acquired in 2017 and 2019 in different months of the year. Figure 1 depicts the spatial distribution of the dataset.



**Figure 1.** Red dots indicate the locations in ETCI 2021 flood detection dataset.

### 2.2. Dataset Description and Pre-Processing

The ETCI 2021 dataset has not been used in many studies, which leaves room for further experimentation. It provides Sentinel-1 images obtained in Interferometric Wide mode with a resolution of  $5 \times 20$  m, which featured labeled pixels before and after the flood [12]. This dataset contains 33,405 image patches in each polarization of VV and VH with a size of  $256 \times 256$  pixels. There are separate binary ground truth images for water bodies and floods in each patch, with the latter one being the focus of this study.

Two pre-processing steps have been conducted on the dataset to prepare it for the training. To begin with, no-data patches, i.e., patches containing no flood pixels, were

removed from the dataset. Upon investigating the remaining patches, it was revealed that a big proportion of the pixels in many of these patches were not flooded. Such an imbalance can have a significant impact on the performance of the model and should be reduced [5]. To tackle this data imbalance, a threshold of 5% was set on the flood pixels in each patch to further filter the dataset. This process assures that at least 5% of the pixels in each patch contain flooding, so the deep learning network can be trained better. Finally, 30% of the remaining patches were dedicated to testing and validation, while the rest were used for training. The pre-processing steps of the dataset are shown in Figure 2.



Figure 2. Pre-processing steps.

### 3. Methodology

Convolutional neural networks (CNN) have been developed for many computer vision tasks such as object detection and semantic segmentation. In this paper, we implemented the U-Net and X-Net architectures for flood mapping and evaluated the performance of the trained networks. Both models use encoder and decoder modules. The encoder module includes a series of convolution layers for feature extraction, along with max-pooling layers that perform downsampling. The decoder is applied after feature extraction and performs upsampling to create a segmentation mask with the same dimensions as the input. The decoder also consists of convolutional layers that allow the extraction of additional features and thus produce a dense feature map [13].

The final convolutional layer features the Sigmoid activation function to produce the binary classification output, while the rest of the layers use the ReLU activation. Cross-entropy is a typical loss function that most models use; however, it does not offer a proper performance when dealing with imbalanced datasets. One good substitute in this situation is the dice loss function [14]. Equation (1) shows the dice loss function, where  $p$  and  $g$  represent the prediction and ground truth images, respectively. The added 1 in the numerator and denominator is to prevent potential undefined values.

$$\text{Dice loss} = 1 - \frac{2 p \times g + 1}{p + g + 1} \tag{1}$$

The encoder branch of the U-Net includes 4 convolutional blocks, each one with batch normalization and max pooling layers. When reaching the bottleneck the convolutional block excludes the max pooling so that the decoding can start. The decoder branch repeats the same convolution operation but uses transpose convolution to retrieve the resolution. It takes 4 blocks in the decoder to rebuild the original image resolution. Another major feature of the U-Net is the concatenation process that transfers the outputs from each block in the encoder to the corresponding block in the decoder. Figure 3 depicts the general scheme of the U-Net model.

X-Net shares the same basic elements as U-Net but introduces a major change in the flow of the features. Instead of 4 convolutional blocks in the decoder, it uses 3 before the bottleneck section followed by 2 blocks of decoding. From here on, the output features enter another encoder and reach the second bottleneck after 2 convolutional blocks. Finally, the second decoder upsamples the outputs to generate the initial resolution. Overall, X-Net is two U-Net models connected in a sequence as shown in Figure 4.

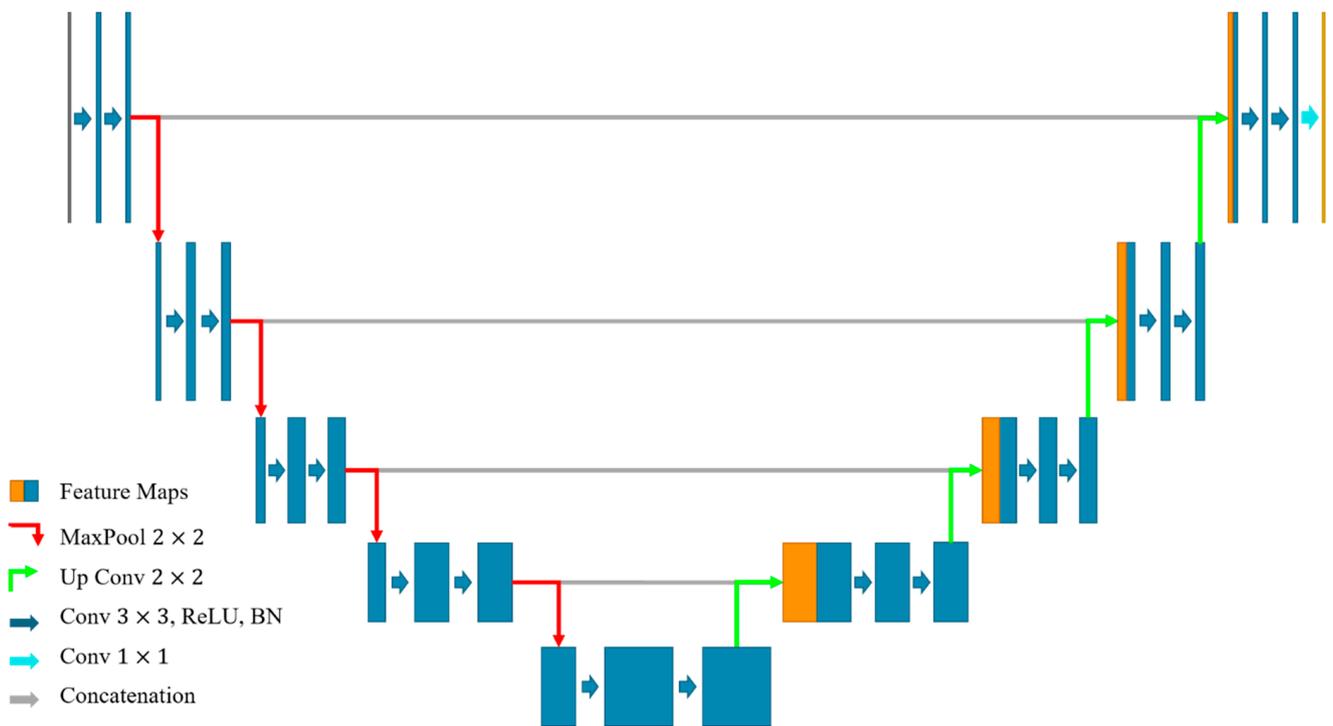


Figure 3. U-Net model architecture.

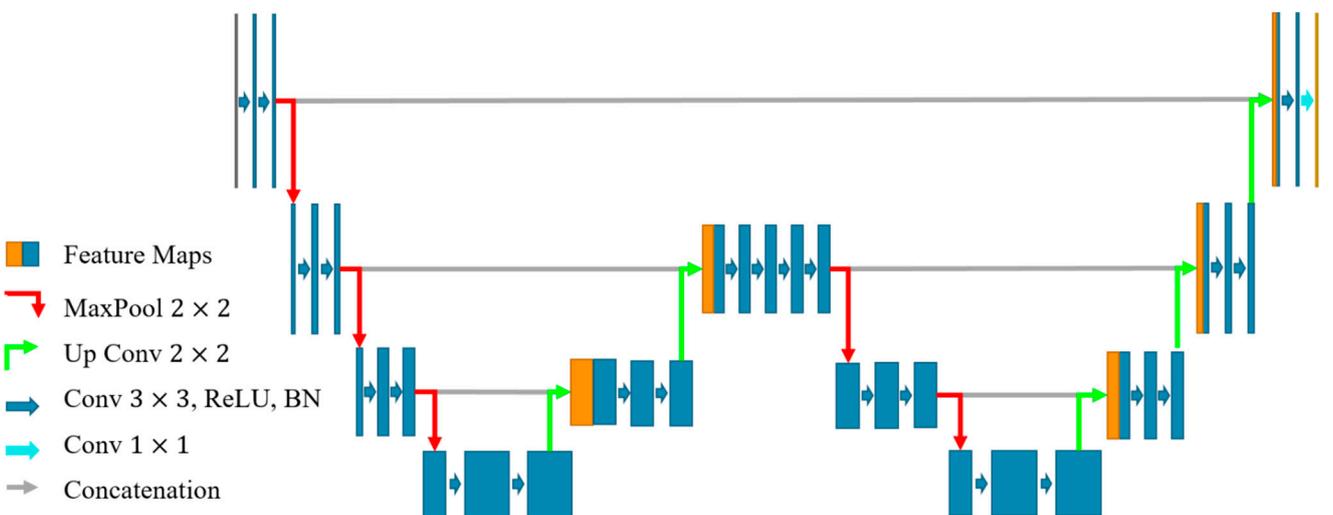


Figure 4. X-Net model architecture.

#### 4. Results and Discussion

In order to assess the performance of the trained models, we utilized the five different evaluation metrics of accuracy, precision, recall, F1-score, and intersection over union (*IOU*). The formulas for these criteria are as follows in which *TP*, *TN*, *FP*, and *FN* represent the parameters of the confusion matrix.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$precision = \frac{TP}{TP + FP} \tag{3}$$

$$recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \tag{5}$$

$$IOU = \frac{TP}{TP + FP + FN} \tag{6}$$

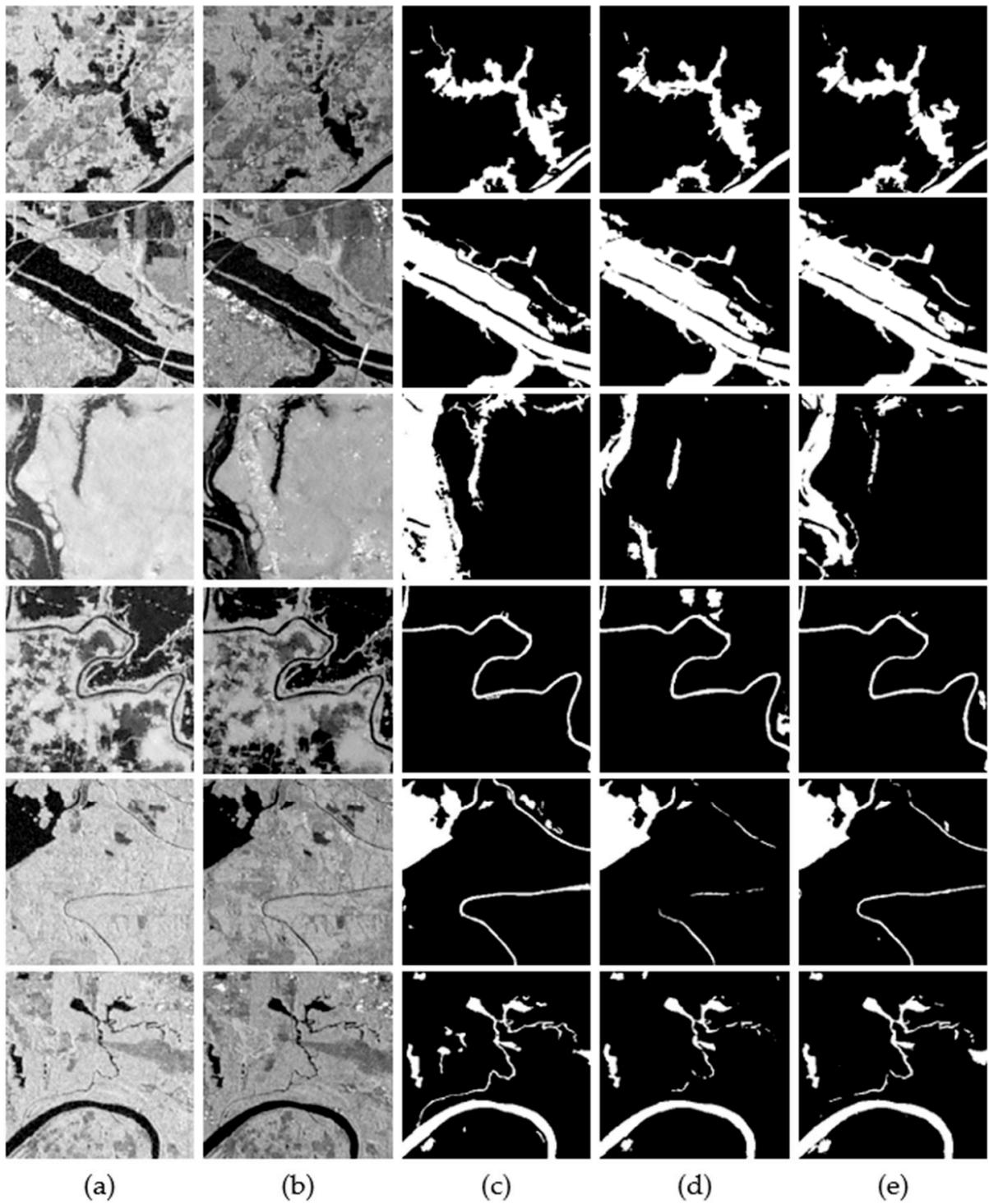
The best metrics to evaluate the model performance are F1-score and *IOU* as they consider the overlap between the prediction and ground truth images. It is especially important in this study because of the imbalanced dataset. Table 1 represents the quantitative results of U-Net and X-Net in two polarizations of VV and VH.

**Table 1.** Quantitative results of the trained models.

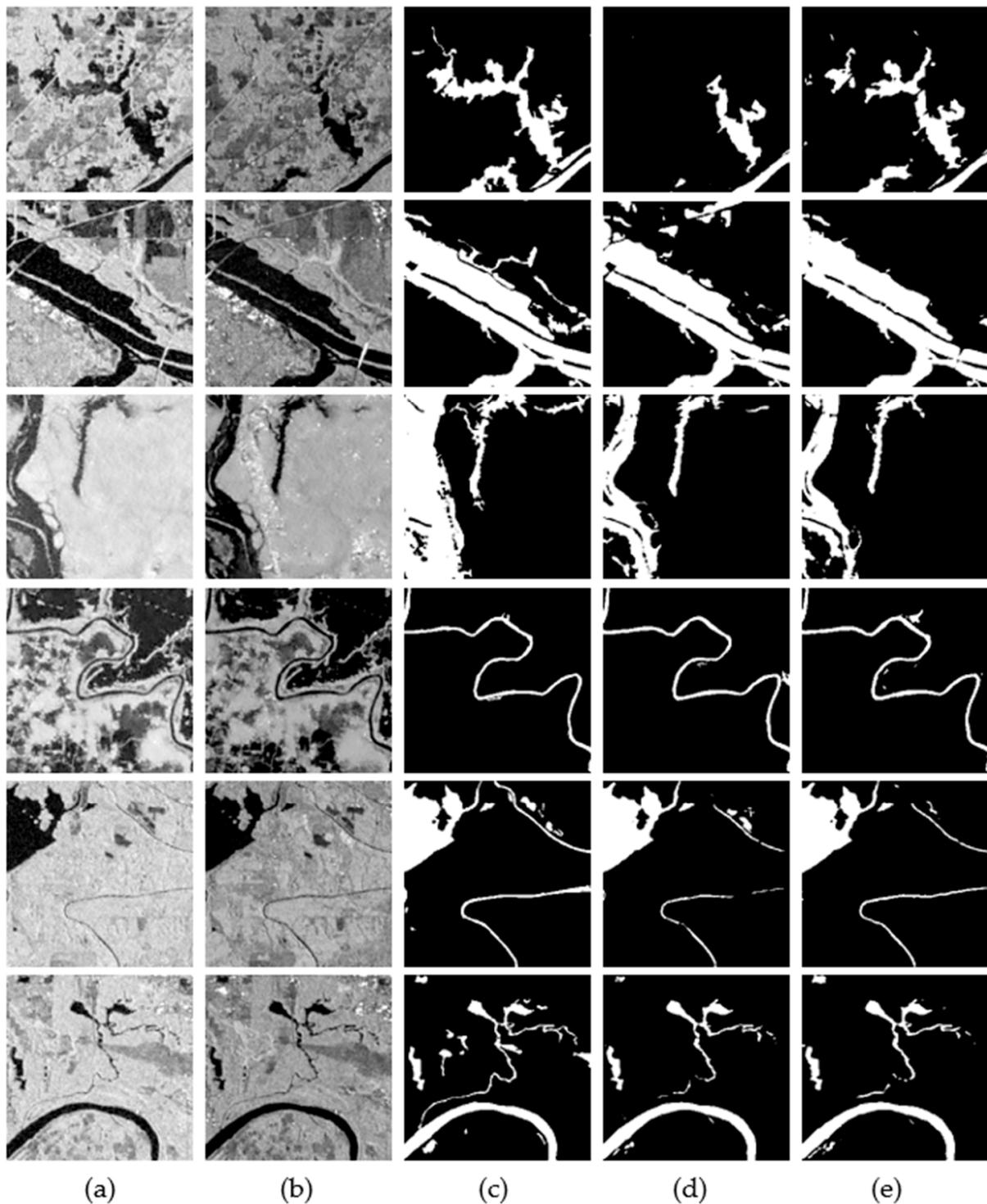
Model	Polarization	Accuracy	Precision	Recall	F1 Score	IOU
U-Net	VH	97.39	84.18	68.80	72.43	64.46
	VV	97.32	83.16	74.10	75.80	67.35
X-Net	VH	96.95	83.10	68.00	71.22	62.54
	VV	97.23	82.18	70.64	72.98	64.38

Overall, VV polarization offered better performance both in U-Net and X-Net with a difference of 2.89% and 1.84% in the *IOU* score, respectively, compared to VH polarization. The F1-score and recall also show similar trends; however, VH achieved slightly better results in the precision score in both models. When comparing the models, U-Net outperformed X-Net in both polarizations and all the metrics. The highest *IOU* score is 67.35%, which is not far from the highest *IOU* score ever achieved using the ETCI-2021 dataset (76.54%) [12]. However, directly comparing this result with the outputs of this study is not fair. That is because the main focus of this study is to find the optimum polarization and model to facilitate the further ablation studies that are usually required for suggesting the best possible model. The visual outputs of the testing phase of U-Net and X-Net are depicted in Figures 5 and 6, respectively.

The visual outputs of both models demonstrate the expected results from the quantitative outputs. As can be seen in Figures 5 and 6, VH polarization in both models introduced noticeable artifacts compared to the VV polarization. Moreover, the former polarization was not able to detect flooded pixels to the same efficiency as the latter, hence achieving higher false negatives. As a result, VV could produce more detailed outputs while better maintaining sharp edges. Regarding the inter-model comparisons, the same trends in the polarizations apply to the models, with U-Net achieving better visual outputs.



**Figure 5.** Visual outputs of U-Net model. (a) VH image; (b) VV image; (c) ground truth; (d) VH prediction; (e) VV prediction.



**Figure 6.** Visual outputs of X-Net model. (a) VH image; (b) VV image; (c) ground truth; (d) VH prediction; (e) VV prediction.

### 5. Conclusions

Timely detection of flooded areas is of key importance to mitigate the damage caused by this devastating natural hazard. Although a big archive of radar imagery is available free of charge, there is a need for a proper framework that can efficiently extract the flooded regions. This study aimed to facilitate this process by examining two polarizations of Sentinel-1 as well as two deep segmentation models. The ETCI 2021 flood event detection

competition dataset was used to train the models, and the outputs were compared by different evaluation metrics. The VV polarization offered better results compared to VH in U-Net and X-Net with an IOU score of 67.35% and 64.38%, respectively. Moreover, U-Net outperformed X-Net with the IOU of VH polarization being 64.46%, which is higher than both polarizations in X-Net. The four testing scenarios proved that it is best to focus on U-Net and VV polarization to further enhance the segmentation outputs. As a result, polarization and model architecture can be excluded in an ablation study to maximize the performance, reducing the number of testing scenarios and the run times substantially.

**Author Contributions:** Conceptualization, M.A.; R.S.-H.; M.A.-N.; methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing visualization, M.A.; R.S.-H.; M.A.-N.; supervision, R.S.-H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** No new data were created.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Rostami, A.; Akhoondzadeh, M.; Amani, M. A Fuzzy-Based Flood Warning System Using 19-Year Remote Sensing Time Series Data in the Google Earth Engine Cloud Platform. *Adv. Sp. Res.* **2022**, *70*, 1406–1428. [[CrossRef](#)]
2. Seyed Mousavi, S.M.; Akhoondzadeh Hanzaei, M. Monitoring and Prediction of the Changes in Water Zone of Wetlands Using an Intelligent Neural-Fuzzy System Based on Data from Google Earth Engine System (Case Study of Anzali Wetland, 2000–2019). *Eng. J. Geospat. Inf. Technol.* **2022**, *9*, 19–42. [[CrossRef](#)]
3. Rostami, A.; Shah-Hosseini, R.; Asgari, S.; Zarei, A.; Aghdami-Nia, M.; Homayouni, S. Active Fire Detection from Landsat-8 Imagery Using Deep Multiple Kernel Learning. *Remote Sens.* **2022**, *14*, 992. [[CrossRef](#)]
4. Aghdami-Nia, M.; Shah-Hosseini, R.; Rostami, A.; Homayouni, S. Automatic Coastline Extraction through Enhanced Sea-Land Segmentation by Modifying Standard U-Net. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *109*, 102785. [[CrossRef](#)]
5. Nemni, E.; Bullock, J.; Belabbes, S.; Bromley, L. Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery. *Remote Sens.* **2020**, *12*, 2532. [[CrossRef](#)]
6. Katiyar, V.; Tamkuan, N.; Nagai, M. Near-Real-Time Flood Mapping Using off-the-Shelf Models with Sar Imagery and Deep Learning. *Remote Sens.* **2021**, *13*, 2334. [[CrossRef](#)]
7. Kim, J.; Kim, D. Extracting Flooded Areas in Southeast Asia Using SegNet and U-Net. *J. Korean Soc. Remote Sens.* **2020**, *36*, 1095–1107.
8. Zhang, L.; Xia, J. Flood Detection Using Multiple Chinese Satellite Datasets during 2020 China Summer Floods. *Remote Sens.* **2022**, *14*, 51. [[CrossRef](#)]
9. Ghosh, B.; Garg, S.; Motagh, M. Automatic Flood Detection from Sentinel-1 Data Using Deep Learning Architectures. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2022**, *3*, 201–208. [[CrossRef](#)]
10. Islam, K.A.; Uddin, M.S.; Kwan, C.; Li, J. Flood Detection Using Multi-Modal and Multi-Temporal Images: A Comparative Study. *Remote Sens.* **2020**, *12*, 2455. [[CrossRef](#)]
11. Tanim, A.H.; McRae, C.B.; Tavakol-davani, H.; Goharian, E. Flood Detection in Urban Areas Using Satellite Imagery and Machine Learning. *Water* **2022**, *14*, 1140. [[CrossRef](#)]
12. Paul, S.; Ganju, S. Flood Segmentation on Sentinel-1 SAR Imagery with Semi-Supervised Learning. *arXiv* **2021**, arXiv:2107.08369.
13. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015, Proceedings of the 18th International Conference, Munich, Germany, 5–9 October 2015*; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; Volume 9351, pp. 234–241.
14. Jadon, S. A Survey of Loss Functions for Semantic Segmentation. In *Proceedings of the 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, Via del Mar, Chile, 27–29 October 2020. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.