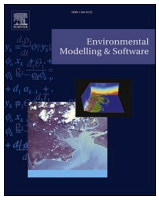
Environmental Modelling and Software 173 (2024) 105939

Contents lists available at ScienceDirect

Environmental Modelling and Software

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



Urban flood extent segmentation and evaluation from real-world surveillance camera images using deep convolutional neural network

Yidi Wang [a,*], Yawen Shen [b], Behrouz Salahshour [c], Mecit Cetin [c], Khan Iftekharuddin [d], Navid Tahvildari [c], Guoping Huang [e], Devin K. Harris [a], Kwame Ampofo [d], Jonathan L. Goodall [a,**]

[a] Department of Civil and Environmental Engineering, University of Virginia, Charlottesville, VA, 22904, USA
[b] Corteva Agriscience Inc, 7000 NW 62nd Ave, Johnston, IA, 50131, USA
[c] Department of Civil and Environmental Engineering, Old Dominion University, Norfolk, VA, 23529, USA
[d] Department of Electrical and Computer Engineering, Old Dominion University, Norfolk, VA, 23529, USA
[e] Spatial Sciences Institute, University of Southern California, Los Angeles, CA, 90089, USA

ARTICLE INFO

Keywords:
Deep learning
Deep convolutional neural network
Semantic segmentation
Flood extent

ABSTRACT

This study explores the use of Deep Convolutional Neural Network (DCNN) for semantic segmentation of flood images. Imagery datasets of urban flooding were used to train two DCNN-based models, and camera images were used to test the application of the models with real-world data. Validation results show that both models extracted flood extent with a mean F1-score over 0.9. The factors that affected the performance included still water surface with specular reflection, wet road surface, and low illumination. In testing, reduced visibility during a storm and raindrops on surveillance cameras were major problems that affected the segmentation of flood extent. High-definition web cameras can be an alternative tool with the models trained on the data it collected. In conclusion, DCNN-based models can extract flood extent from camera images of urban flooding. The challenges with using these models on real-world data identified through this research present opportunities for future research.

1. Introduction

Cities across the globe are becoming increasingly vulnerable to flooding primarily due to climate change and urbanization, which motivates the need to enhance flood resilience for cities (Hallegatte et al., 2013; Li et al., 2013; Neumann et al., 2015; Moy de Vitry et al., 2019). One of the key components to building flood resilient cities is real-time decision support, which requires collecting real-time information from flood monitoring and sensor networks (Manzoor et al., 2014; Perez et al., 2015; Loftis et al., 2018). Urban flood monitoring networks that can be used for real-time decision making are growing but still lack the desired coverage of monitoring stations needed to make fine-scale decisions during flood events (Helmrich et al., 2021; Rosenzweig et al., 2021). Research is needed to advance sensing approaches for fine-scale, real--time urban flood monitoring (Moy de Vitry et al., 2019; Muhadi et al., 2020).

Current state-of-the-art flood monitoring most often uses in-situ water level sensors (Loftis et al., 2018; Carlson et al., 2019) although remote sensing using satellites (Li et al., 2019), unmanned aerial vehicles (UAVs) (Gebrehiwot et al., 2019) or fixed-point surveillance cameras (Moy de Vitry et al., 2019) has also been explored. Each of these sensing modalities has different strengths and shortcomings. In-situ water level sensors can collect high-accuracy water level data at critical points. However, they lack the ability to represent the spatial distribution of flooding since they only observe a single point (Lo et al., 2015). Having spatially distributed flood observations captured through remote sensing can be valuable not only for aiding decision makers, but also for building and calibrating hydrological and hydraulic models of urban drainage system for forecasting purposes (Chao et al., 2019; Shen et al., 2019; Liu et al., 2020).

* Corresponding author.
** Corresponding author.
E-mail addresses: yw5vq@virginia.edu (Y. Wang), ys5dv@virginia.edu (Y. Shen), bsala001@odu.edu (B. Salahshour), mcetin@odu.edu (M. Cetin), kiftekha@odu.edu (K. Iftekharuddin), ntahvild@odu.edu (N. Tahvildari), guopingh@usc.edu (G. Huang), dharris@virginia.edu (D.K. Harris), kampo001@odu.edu (K. Ampofo), goodall@virginia.edu (J.L. Goodall).

https://doi.org/10.1016/j.envsoft.2023.105939
Received 1 September 2023; Received in revised form 10 December 2023; Accepted 27 December 2023
Available online 27 December 2023

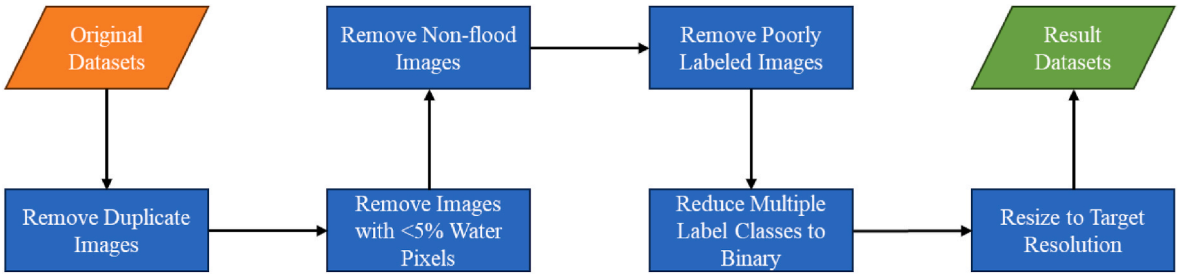


Fig. 1. Flow chart of preprocessing of the two training datasets.

While remote sensing is suitable for large-scale flood monitoring, satellite or UAV mounted sensors can be limited to low spatial or temporal resolution or be impacted by clouds during storms (Colosio et al., 2022; Munawar et al., 2022). Moreover, the operation of a UAV during a storm event or in urban areas with airspace restrictions can be barriers to their use in urban flood monitoring. In contrast, using surveillance cameras already deployed in many cities in the world for traffic and safety monitoring can provide a powerful means for flood monitoring (Goold et al., 2010; Moy de Vitry et al., 2019). They are less impacted by weather conditions compared to satellite or UAV remote sensing. Furthermore, surveillance cameras can have high temporal resolution and, with sufficient density within a city, high spatial coverage (Muhadi et al., 2021). These characteristics make surveillance cameras an attractive approach for flood monitoring in urban environments, but their application for this purpose, especially in any automated application that does not require humans to monitor for floods in the camera imagery, has not been widely adopted to date.

The use of surveillance cameras for water monitoring is growing. For example, the U.S. Southeast Coastal Ocean Observing Regional Association (SECOORA) in partnership with the National Oceanic and Atmospheric Administration (NOAA) has built up a web camera observation network through the project of Webcams for Coastal Observations and Operational Support (WebCOOS), also known as the Web Camera Applications Testbed (WebCAT) in the past (Dusek et al., 2019; Taylor et al., 2022; WebCOOS, 2023). This network has been installed to detect rip currents, observe coastal water levels, and monitor flooding. The current phase of WebCOOS aggregates and displays all available web camera feeds, allowing the user to view photo and video galleries and access the data via the application programming interface. Using a machine learning (ML)-based method, these data have been successful at detecting rip currents without human intervention (de Silva et al., 2023). This suggests ML approaches could also be used for automated, machine-assisted monitoring of street-scale nuisance floods, although this application has not been widely implemented yet. Within the context of the larger goal of establishing an urban flood monitoring network, an ML algorithm trained, tested, and validated for identifying flooded areas in surveillance imagery could be routinely used to detect and determine the severity of flooding across cities, leverage already deployed camera networks. This capability can provide notification and documentation of these events in near real-time to better understand and remedy flooding impacts (WebCOOS, 2023).

Previous research has shown that automated, ML-based methods applied to imagery data can potentially aid in monitoring floods in urban areas, especially when using Deep Convolutional Neural Network (DCNN). Earlier work before the use of DCNN included Lo et al. (2015) who proposed a visual sensing method of urban floods that uses a virtual seed to analyze the waterbody surface texture and aid the segmentation of waterbodies in surveillance camera images, and several virtual markers to indicate the actual flooded areas. Although the results showed that the method can be used in monitoring and determining if flooding exists, the complexity of manually setting virtual seeds can limit the applications on a large scale. Moy de Vitry et al. (2019) used U-Net to segment flood water in surveillance camera images from different scenarios, which provided an Intersection over Union (IoU) higher than 90% on average. Vandaele et al. (2021) compared the performances of two different DCNN models in the segmentation of water in the images from a river camera and found that Deeplabv3 achieved better performance. Muhadi et al. (2021) compared two other DCNN models and estimated river water levels by overlaying the water surface extents segmented by DCNN to the predefined water level markers extracted from LiDAR data. The estimated water levels showed a high correlation with the observed water levels. In a recent study, Sazara et al. (2022) applied different DCNNs (U-Net, VGGNet, MobileNet, and ResNet for semantic segmentation and YOLO for object detection) for predicting the floodwater depth using sideview images of vehicle tires partially submerged in still water as reference objects.

This paper advances on the long-term goal of automated, surveillance camera-based flood sensing and previous research of using ML-approaches to identify and segment water within images. In previous research, applying DCNN on image classification for flood monitoring showed good results when using simple and idealized images of natural open water bodies such as rivers, canals or lakes. However, studies applying these algorithms to real-world imagery obtained from already deployed surveillance cameras are limited, especially in urban contexts. There has been some work using surveillance cameras dedicated to water surface monitoring of fluvial flooding, where flooded areas were typically adjacent to a constant open water body (e.g., river, pond, lake, etc.) in the images. However, the camera-based real-time monitoring of urban pluvial flooding, which often exists as urban nuisance flooding in practice, with shallower water depth, more scattered flooded areas, and more rapid and drastic changes in flood extent is not sufficiently studied We argue that there are unique and unaddressed research challenges with the segmentation of pluvial flooding in imagery compared to fluvial flooding. Moreover, the use of surveillance cameras installed originally for purposes such as security or traffic monitoring for environmental monitoring can cause difficulties. The location of installation, the video resolution and other specifications of the surveillance camera may present barriers that must be overcome to reach the goal of urban flood monitoring. Finally, automated monitoring of flooding that does not require human intervention can also be a challenge. The complexity of urban environments, with many people, vehicles, and other dynamic conditions can also bring challenges to the detection and segmentation of flood extent with DCNN. However, if ML algorithms can be trained to overcome these challenges presented by real-world conditions, they could greatly benefit flood warning and mitigation strategies within cities.

Given this motivation and unaddressed research challenges, the goal of this study is to address the feasibility of the monitoring of pluvial flooding in urban environments using existing real-world surveillance cameras deployed without a direct intent for flood observation and ML algorithms. To achieve this goal, this study consists of the following three objectives: (1) training ML segmentation models using a variety of urban flood images and exploring the factors that limit the performance of the models in segmenting flood extent, (2) testing and evaluating the performance of the ML segmentation models using real-world surveillance data and discussing the factors that are still limiting the broad

Table 1
Summary of the image datasets used for training, validation and test.

Dataset Name	Original Resolution	Resolution After Resizing	Total Size	Training Set Size	Validation Set Size	Test Set Size
Sazara	512 × 384	512 × 384	253	203	50	–
Deepflood	Non-Uniform	512 × 384	1040	832	208	–
Deepflood HD	Non-Uniform	1024 × 768	216	173	43	–
ODU	640 × 480	512 × 384	36	–	–	36
WebCOOS	2560 × 1920	1024 × 768	35	4[a]	1[a]	30

[a] These five images were added to the Deepflood HD dataset to create the extended Deepflood HD dataset.

adoption of cameras for flood monitoring, and (3) discussing the role of the result of this study in a real-time urban flood monitoring system that can serve as motivation for future research.

2. Materials and methods

2.1. Datasets

In this study, three datasets, referred to as Sazara, Deepflood and Deepflood HD, are employed for training and validation of the semantic segmentation models. Two other datasets from cameras, referred to as ODU and WebCOOS, are used for testing the models, with a small number of images in the WebCOOS dataset used for training in some trials. All datasets contained data in the form of pairs of camera image files and the corresponding label files of flood extent and underwent a series of preprocessing steps to be prepared for training, validation or testing (Fig. 1). Duplicate images within and across the datasets were filtered out using the Duplicate Image Finder (Landman, 2022). Images with a small percentage of flood pixels (less than 5%) were also removed from the dataset. Images with flood extent poorly labeled, and images of natural water bodies irrelevant to urban flood warning were manually removed. While most images had binary label classes of flooded or non-flooded, a small portion of the Deepflood dataset contained multiple classes of objects (e.g., humans, cars, buses, buildings, etc.). Since this study is focused on urban flooding, we made these label files binary as well. Although the datasets vary in sizes, the main purpose of using different datasets in this study is not comparing them, but ensuring the variety of data sources and data quality in training and testing the segmentation models to reduce bias. We have included the number of images of each dataset in Table 1 to better clarify the context.

2.1.1. Sazara dataset

The Sazara dataset contained data in the form of pairs of camera image files and the corresponding label files of flood extent (Sazara et al., 2019). Label files were in the format of image files, but with only pixel values of 1 or 0, indicating whether each pixel is flooded or non-flooded. The Sazara dataset consists of 253 images containing flooding (Sazara et al., 2019). All images have the same resolution of 512 × 384. The dataset was split at a ratio of 80:20 for training and validation, respectively, which resulted in a training set of 203 and a validation set of 50 images (Table 1).

2.1.2. Deepflood and Deepflood HD dataset

The Deepflood dataset consists of 8145 images in non-uniform resolutions with 1259 images having the corresponding labeled flood extent (Chaudhary et al., 2020). After the preprocessing, the Deepflood dataset contained 1040 images and was then split into a training set of 832 and a validation set of 208 images (Table 1). In accordance with Sazara dataset, all images and the corresponding labels in Deepflood dataset were resized to a uniform resolution of 512 × 384. Additionally, the images that have an original resolution greater than 1024 × 768 were selected to create the Deepflood HD dataset. The Deepflood HD dataset contained 216 images, which were then resized to a uniform resolution of 1024 × 768 and split into a training set of 173 and a validation set of 43 images.

2.1.3. ODU dataset

To assess the performance of the models in extracting the flood extent in real-world images from surveillance cameras, this study created an image dataset based on the video recordings from a surveillance camera on the ODU campus in Norfolk, Virginia. The camera recorded a video of one flood event from 2:00pm to 3:00pm on the 16th of August 2021. The frame rate of the camera was 10 frames per second, and the image resolution of the video was 640 × 480. This study captured images from the video at a temporal resolution of 1 min. Images without significant flooding were removed, and a dataset was created with 36 images captured from each minute between 2:25pm and 3:00pm (Table 1). The labels of flood extent were manually sketched using Labelbox. Similarly, all images and labels were resized to the same resolution as the Sazara and the Deepflood dataset.

2.1.4. WebCOOS dataset

As a supplementation of the limited image quality of the ODU dataset, this study created another image dataset based on the image recordings from a high-resolution web camera affiliated with the WebCOOS project in Charleston, SC. Hurricane Ian on September 30th, 2022 has caused the highest daily rainfall in the city since 2021. Despite several cameras that recorded this flood event, they were installed in the same neighborhood, with similar environmental and flooding conditions. Therefore, this study selected Hurricane Ian as the representative flood event, and collected flood images from one of the cameras that had the best captures of flooding. The original resolution of the images was 2560 × 1920, which was 16 times that of the ODU dataset. The time interval between two consecutive images was primarily 10 min in most cases with occasional instances of missing or extra captures. By collecting the images with a clear view and good illumination condition, a subset of 35 images spanning from 9:40am to 4:30pm was selected, and the labels of flood extent were manually sketched using Labelbox. All images and labels were resized to the same resolution as the Deepflood HD dataset. Five images were added to the Deepflood HD dataset to create an extended Deepflood HD dataset, and the rest 30 images were used for model testing (Table 1). To enhance the utility of this dataset, this study added attributes such as brightness, weather conditions, blurriness and camera source to each image. The attribute table is available together with the dataset (see the Data Availability section).

2.2. Models

Semantic segmentation is the process of associating each pixel of an image to a class label such as human, vehicle or waterbody (Audebert et al., 2017; Tomar, 2021; Pally and Samadi, 2022; Liang et al., 2023). This study made use of Deeplabv3+ and LinkNet, two different ML algorithms capable of semantic segmentation. While Deeplabv3+ is categorized as dilated convolutional model due to its unique features such as the ASPP module, LinkNet is one of the general image segmentation models. Comparing the two models can improve our understanding of the behavior of the models under different conditions and their strengths and weaknesses. Although the two algorithms were developed years ago, they are still state-of-the-art networks for the research objective of this study. Previous research showed that Deeplabv3+ can outperform some of the more recent networks in image

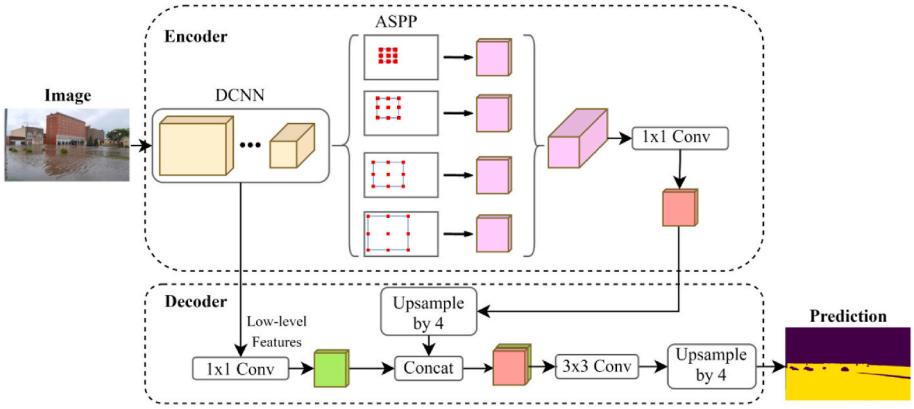


Fig. 2. Structure of Deeplabv3+ as used in this study (Chen et al., 2018). The encoder module encodes multi-scale contextual information by applying atrous convolution at multiple scales, and the decoder module refines the segmentation results along object boundaries.

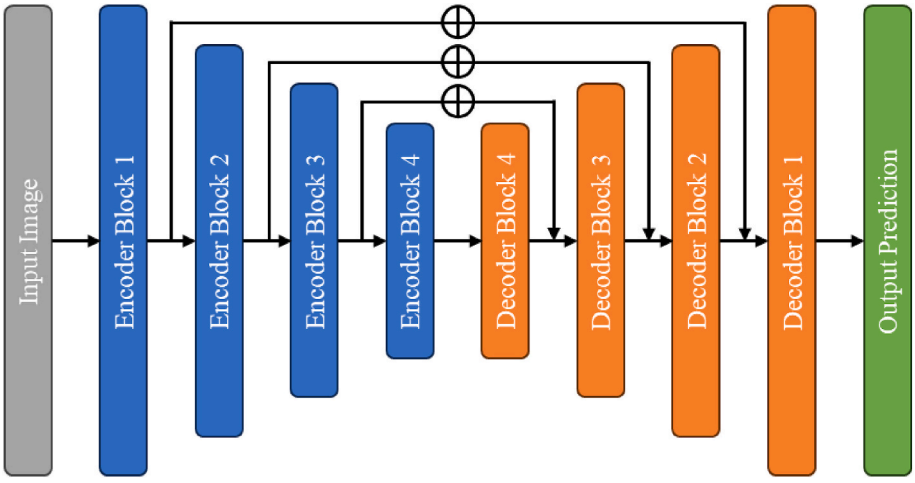


Fig. 3. Structure of LinkNet as used in this study.

segmentation (Minaee et al., 2021). Considering our task involves the segmentation of only one label class in a fixed category of images, the networks that we selected are capable enough to accomplish the task. Other algorithms, such as EfficientNet and NAS-FPN, can be potentially selected for more complicated segmentation tasks in the future (Ghiasi et al., 2019; Tan and Le, 2019).

2.2.1. *Deeplabv3+*

The first model used in this study was Deeplabv3+ with the backbone of ResNet50 for semantic segmentation of flood extent. Deeplabv3+ was built on Tensorflow and extended from the Deeplabv3 with an encoder-decoder structure to refine the segmentation results especially object boundaries (Chen et al., 2017, 2018). Deeplabv3 was adopted as the encoder module of the encoder-decoder structure of Deeplabv3+. This structure enhances the segmentation with shaper object boundaries by gradually recovering the spatial information (Chen et al., 2018). In Deeplabv3+, atrous convolution, also called dilation convolution, is used as a context module for Atrous Spatial Pyramid Pooling (ASPP) (Fig. 2). Atrous convolution can enhance segmentation by addressing the challenge of capturing both local details and broader contextual information. Traditional convolutions have limited receptive fields, restricting the ability to capture long-range dependencies crucial for segmentation tasks. Atrous convolution introduces gaps in the filters, enabling the integration of information from a wider spatial context without downsampling the feature map. This mechanism can preserve the fine-grained details while allowing the model to understand relationships between pixels across a larger region. The use of atrous convolution facilitates multi-scale feature integration, as different dilation rates can be employed to simultaneously capture fine details and larger contextual patterns. Therefore, atrous convolution can improve the segmentation performance by providing the model with a more comprehensive understanding of the image, enabling more accurate pixel-wise prediction. Given the rapid changes of flooded area during a storm event, we believe that this will improve the ability of Deeplabv3+ to detect and extract flood extent more accurately.

2.2.2. *LinkNet*

The second model used in this study was LinkNet with the same backbone of ResNet50 as the reference model for baseline comparison to Deeplabv3+. LinkNet is an efficient DCNN which takes advantage of skip connections, residual blocks and encoder-decoder architecture (He et al., 2016). The original LinkNet uses ResNet18 as the encoder, which is lightweight but outperforming (Zhou et al., 2018). Similar to Deeplabv3+, LinkNet has an encoder-decoder structure, but the novelty of LinkNet is that each encoder is linked with a decoder (Fig. 3). Multiple downsampling operations performed in an encoder can result in the loss of some spatial information. In LinkNet, the input of each encoder is passed to the output of its corresponding decoder, which can recover the lost spatial information. Furthermore, due to the information passed from the encoder at the same layer, the decoders of each layer can use fewer hyperparameters. This results in an overall more efficient network when compared to some state-of-the-art segmentation networks (Chaurasia and Culurciello, 2017).

Table 2
Computational cost for training and validation of the two models using the three training sets.

Model	Training Set	Resolution	Training Set Size	Batch Size	Actual Number of Epochs	Total Computational Time (min) [a]	Average Computational Time per Step (μs)
Deeplabv3+	Sazara	512 × 384	203	2	37	19.21	202
	Deepflood	512 × 384	832	8	43	62.20	733
	Deepflood HD	1024 × 768	173	2	39	46.49	625
LinkNet	Sazara	512 × 384	203	2	33	13.20	162
	Deepflood	512 × 384	832	8	51	59.43	542
	Deepflood HD	1024 × 768	173	2	41	43.54	547

[a] This includes the computational time of validation at the end of each epoch.

2.3. Training, validation and test

In this study we call training the process of fitting the hyperparameters of the model to the data in the training dataset and validation the process of updating the hyperparameters according to the estimated generalization error during or after training (Gareth et al., 2013; Goodfellow et al., 2016). After being trained and validated for sufficient epochs, the models were tested on test dataset for assessment.

Both Deeplabv3+ and LinkNet were trained on Sazara and Deepflood training datasets separately. All layers of the two models were trained. The training process was controlled by multiple hyperparameters, including batch size, learning rate, number of epochs, etc. Validation loss and the other metrics were computed and recorded at the end of each epoch. The number of epochs in this study was set to be 100, and the training was set to terminate through early stopping when the validation loss was not improved for 20 consecutive epochs. With an initial value of 0.0001, the learning rate was set to be reduced by 10 times when the validation loss was not improved for 5 consecutive epochs. To balance between the limitation of capacity of memory and the average improvement over an epoch, the batch size of training in this study was set to be 8 for the Deepflood dataset and 2 for the others, so that the numbers of data in an epoch of each training session were similar. The model was trained and validated using Python 3.10.11 on Google Colaboratory with the configurations as following: Nvidia Tesla K80 GPU @ 0.82 GHz with 12 GB VRAM (Kegenbekov and Jackson, 2021). Total computational time spent on training and the average computational time per step, i.e., the time of training the model on one image, was recorded at the end of each training session.

To explore the impact of image quality on the ability of the models to segment flood extent and evaluate the potential of leveraging web cameras for urban flood monitoring, this study also used the Deepflood HD dataset for the training and validation of the models, and the WebCOOS dataset for testing. The training and validation of the models at this image resolution were divided into two groups. In one group, both models were trained and validated using the Deepflood HD dataset only. In the other group, an extended Deepflood HD dataset was created by adding four images from the WebCOOS dataset to the training set and one image to the test set, and then used for training and validating the models. After training, models in both groups were then tested on the rest of the images in the WebCOOS dataset.

In validation and testing, mean Intersection over Union (IoU), F1-score, precision and recall were used to assess the performance of the semantic segmentation models. IoU, which is also known as the Jaccard coefficient, quantifies the percentage overlap between the ground-truth label and the result of the model. It is defined as

$$IoU = \frac{intersecion}{union} = \frac{ground\ truth \cap prediction}{ground\ truth \cup prediction} \quad (1)$$

where the intersection is the number of flooded pixels found in both the ground-truth label and the modeled output, and the union is the number of flooded pixels found in either the ground-truth label or the modeled output. Precision and recall are the other two commonly used metrics for semantic segmentation (Sazara et al., 2019). The relationship of the three metrics is defined as the F1-score as defined in Equation (2).

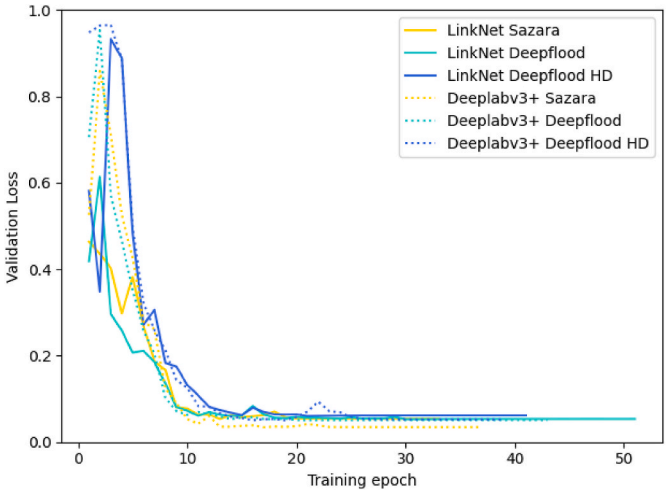


Fig. 4. Validation loss versus the number of epochs during the training of Deeplabv3+ and LinkNet on the three training datasets.

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (2)$$

While IoU specifically measures the overlap between the predicted and ground truth regions, F1-score focuses on the model's overall accuracy by combining precision and recall. In addition, previous research showed that the accuracy of image-based segmentation is limited at boundaries of water extent, especially under bad weather conditions (Liang et al., 2023). Therefore, this study also extracted the edge areas of the segmented flood extent, and computed boundary IoU as another metric to quantitatively evaluate the segmentation quality at boundaries.

The two models were then tested by segmenting the flood extents in the images from the ODU and the WebCOOS dataset, and measured using the four metrics described above. Results from the two models were compared to each other and to the original images for assessing the applicability of the models.

3. Results

3.1. Training of models

The computational time of training was summarized in Table 2. Due to the randomness of weight changing in the training process, the actual number of epochs of training a model varies, which made the total computational time of training each model not comparable. However,

Table 3
Results of all training and validation trials using the Sazara and the Deepflood dataset.

Model	Dataset	Training Set Size	Validation Set Size	IoU (%)	Boundary IoU (%)	F1-score (%)	Recall (%)	Precision (%)
Deeplabv3+	Sazara	203	50	93.29	48.21	96.36	97.11	95.95
	Deepflood	832	208	89.75	54.50	94.26	94.77	94.18
LinkNet	Sazara	203	50	90.60	32.12	94.87	95.91	94.26
	Deepflood	832	208	89.31	53.30	94.02	94.42	94.13

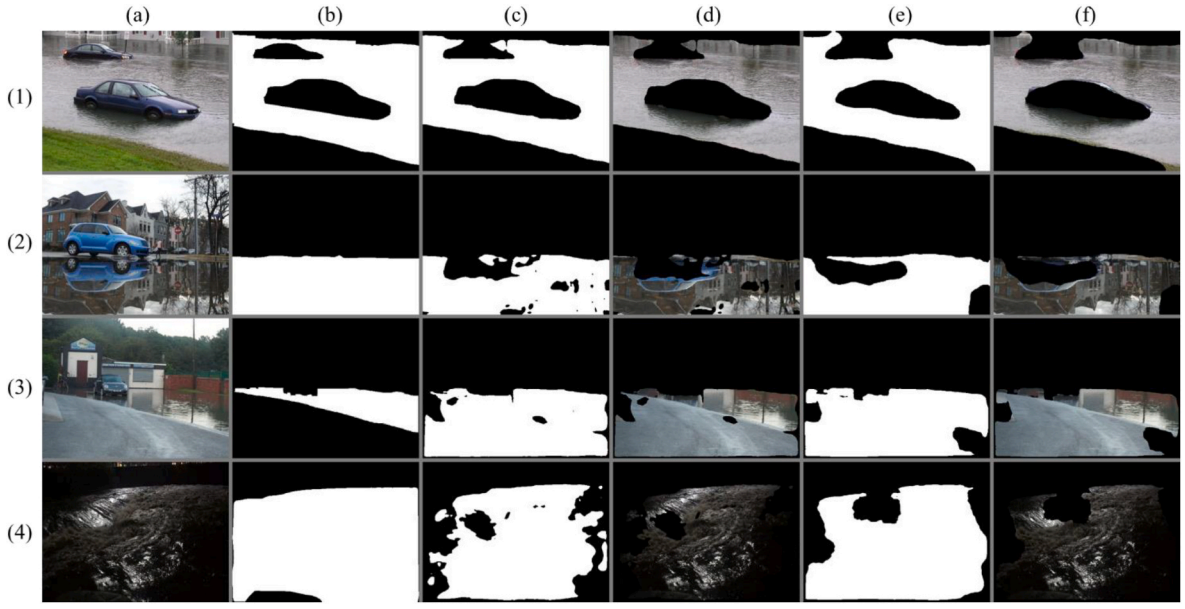


Fig. 5. Examples of model performance in validation. From top to bottom, the four cases are: (1) slightly wavy water surface with normal illumination, (2) still water surface with specular reflection, (3) wet road surface, and (4) night condition. In each case, the six images from left to right are: (a) the original image, (b) ground-truth flood extent, (c) segmented flood extent using Deeplabv3+, (d) image masked with Deeplabv3+ result, (e) segmented flood extent using LinkNet, and (f) image masked with LinkNet result.

the average computational time per step of training Deeplabv3+ was higher than that of training LinkNet, which was a reason that made Deeplabv3+ more time-consuming than LinkNet in this study in terms of the total computational time of training. Despite different resolutions, the average computational time per step on the Deepflood dataset was close to that on the Deepflood HD dataset, but higher than that on the Sazara dataset, which had the same resolution. Early stopping was triggered in all model training sessions prior to completing the pre-defined number of epochs (Fig. 4), which showed that the training of the models in all sessions was complete. While the models trained on the Deepflood HD dataset exhibited a relatively slower decrease in validation loss compared to the other two datasets, the validation loss for all training sessions reached below 0.1 within 12 epochs.

3.2. Validation of models

The validation results of both semantic segmentation models trained on the Sazara and Deepflood datasets were compared in Table 3. Based on the mean IoU, F1-score, recall and precision of all validations, both models achieved similarly good performance, with Deeplabv3+ showing a slightly better performance than LinkNet. The mean IoU, mean boundary IoU, mean F1-score, mean recall, and mean precision in all validations using Deeplabv3+ were 90.44%, 53.28%, 94.67%, 95.22% and 94.52%, respectively. The results of LinkNet were slightly lower than those of Deeplabv3+, which were 89.56%, 49.20%, 94.18% 94.71% and 94.16%, respectively.

We compared our model results to the results of the Fully Convolutional Neural Network (FCNN) model published in previous literature (Sazara et al., 2019). Both models were trained on the Sazara dataset. The FCNN model, both trained and validated on the Sazara dataset, reached a precision of 92%, a recall of 90%, and an F1-score of 91%. On the same dataset, the Deeplabv3+ model developed in this study achieved a precision of 95.95%, a recall of 97.11%, and an F1-score of 96.36%, and these metrics for LinkNet were 94.26%, 95.91%, and 94.87, respectively (Table 3). Both models in this study showed better results than those of the FCNN model.

To better understand the factors that affect the performance of the models in segmenting flood extent, four representative examples of segmentation results in validation were visualized as shown in Fig. 5, which showed different conditions of water surface and light illumination. The image in Case (1) was under normal light condition with relatively smooth water surface, and the simulated flood extent was close to the ground-truth, which gave an IoU of 95.29% with Deeplabv3+ and 91.13% with LinkNet. This is the predominant case in all validation results. Case (2) showed the problem with still water with specular reflection, where some part of the water was not identified as flood pixels due to the virtual images of objects in water. Despite the precision over 97% for both models, the recall of this case was 81.53% with Deeplabv3+ and 77.51% with LinkNet. Case (3) showed a failed validation due to the high reflectance of the wet road surface. In this case, both models incorporated the non-flooded road surface as part of the flood extent, which resulted in a precision of 29.63% with Deeplabv3+ and 31.40% with LinkNet. Case (4) showed the model performance under night conditions, where both models failed to identify the pixels of poorly illuminated flood near the image boundary. This case gave a recall of 70.62% with Deeplabv3+ and 78.58% with LinkNet.

Table 4
Results of the models tested on the ODU dataset.

Model	Training Set	IoU (%)	Boundary IoU (%)	F1-score (%)	Recall (%)	Precision (%)
Deeplabv3+	Sazara	51.45	8.90	67.78	96.19	52.62
	Deepflood	60.12	12.67	74.89	84.44	67.74
LinkNet	Sazara	55.57	5.50	71.28	93.55	57.83
	Deepflood	61.58	14.43	75.99	84.88	69.78

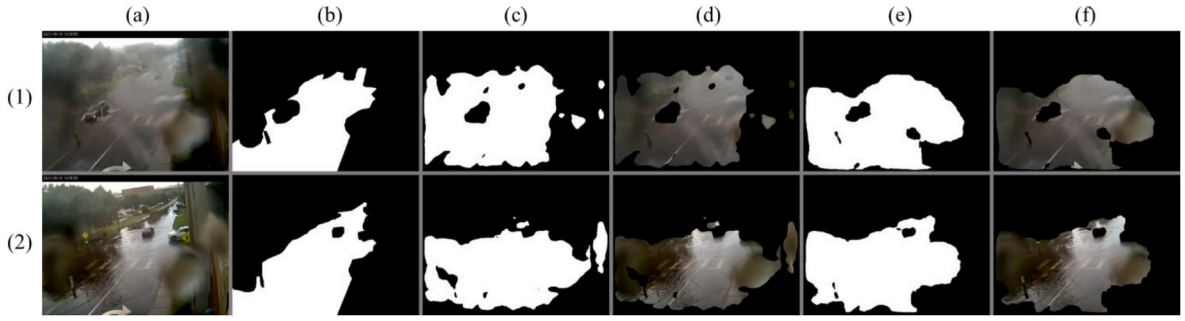


Fig. 6. Examples of model performance in segmenting the flood extent (1) during the storm (near 2:31pm) and (2) after the storm (near 2:57pm) in the ODU dataset. In each row, the six images from left to right are: (a) the original image, (b) ground-truth flood extent, (c) segmented flood extent using Deeplabv3+ (d) image masked with Deeplabv3+ result, (e) segmented flood extent using LinkNet, and (f) image masked with LinkNet result.

3.3. Testing using surveillance camera images of real-world urban flooding

For model testing, the two models trained using different datasets were applied on the real-world flood images in the ODU dataset. Results showed that the performance of both models were not as good as in validation (Table 4). The mean IoU, mean boundary IoU, mean F1-score, mean recall, and mean precision for Deeplabv3+ were 55.79%, 10.79%, 71.34%, 90.31% and 60.18%, while these metrics for LinkNet were 58.58%, 9.97%, 73.64%, 89.22% and 63.81%, respectively. All four cases showed higher recall than precision.

To better see the differences in the model performances and explore the potential factors that caused the testing results, examples of segmented flood extent during and after the storm event using both models trained on the Deepflood dataset were compared (Fig. 6). In general, both models faced significant constraints imposed by environmental factors. During the storm, the camera view became blurry due to the reduced atmospheric visibility, which complicated the identification of water on the road surface. In the meantime, raindrops started to adhere to the camera lens, which blocked the view of the surveillance camera. After the end of the storm, the raindrops remained on the lens, sustaining the adverse impacts on the accuracy of flood extent segmentation. Depending on the clarity of the camera view, Deeplabv3+ had the potential to integrate the raindrop-blocked areas into the flood extent, with LinkNet exhibiting an even greater probability for this potential. Compared to LinkNet, the segmented flood extents using Deeplabv3+ were more fragmented, characterized by irregular boundaries and dispersed clusters of identified pixels.

3.4. Training and testing using high-resolution web camera images

The validation results of the models trained on the Deepflood HD and the extended Deepflood HD dataset, and their testing results on the WebCOOS dataset were compared in Table 5. Both models showed similar performance in validation between the two training sets. The mean IoU, mean boundary IoU, mean F1-score, mean recall, and mean precision of both models trained on the Deepflood HD dataset were 89.01%, 39.47%, 93.65%, 94.92% and 93.30%, respectively, whereas those of the models trained on the extended Deepflood HD dataset were 88.16%, 36.46%, 93.13%, 95.87% and 91.69%, respectively. Deeplabv3+ achieved slightly higher scores in these metrics than LinkNet.

Compared to validation, the test results based on the same test set were more distinct between the models trained on the two different training sets. Despite the higher image resolution of the data used in training, validation and testing, the models that were trained on the Deepflood HD dataset achieved a mean IoU of 53.45%, a mean boundary IoU of 36.46%, a mean F1-score of 69.36%, a mean recall of 82.01% and a mean precision of 61.31% in testing, whereas the test results based on the extended Deepflood HD dataset were much improved, with 78.11%, 19.60%, 87.22%, 88.86% and 86.42% for those metrics when tested on the same test set. Both models trained on the Deepflood HD dataset showed similar results in testing, but Deeplabv3+ performed slightly better than LinkNet after both models were trained on the extended dataset.

Examples of segmented flood extent by different models were compared to show the factors that led to the differences in the test results (Fig. 7). Given the higher image resolution, the clarity of the images in the WebCOOS dataset was higher than that in the ODU dataset, which made it easier to identify the flooded areas. In the example images, only half of the road surface was flooded, but the models that were trained on the Deepflood HD dataset incorrectly incorporated the wetted but not flooded road surface in the flood extent, which increased the false positives. The grass on the sidewalk was submerged under water, but the models based on the Deepflood HD dataset did not include those areas in the flood extent, which was a source of false negatives. In comparison, the models trained on the extended Deepflood HD dataset mitigated these problems more effectively and therefore improved the accuracy of the segmentation.

4. Discussion

4.1. Ability of DCNN to extract flood extent

The models used in this study showed good performance in extracting flood extent from images of flood in an urban context, which indicates the potential application of DCNN-based models in the monitoring of urban floods. The datasets for training and validation in this study consist of the images of urban flooding in a wide variety of

Table 5
Results of the training, validation and testing of the models in different trials.

Model	Training Set	Training Set Size	Validation Set Size	Validation					Testing Set	Testing Set Size	Testing				
				IoU (%)	Boundary IoU (%)	F1-score (%)	Recall (%)	Precision (%)			IoU (%)	Boundary IoU (%)	F1-score (%)	Recall (%)	Precision (%)
Deeplabv3+	Deepflood HD	173	43	89.85	40.37	94.28	95.57	93.73	WebCOOS	30	53.75	13.65	69.67	80.79	62.55
LinkNet				88.17	38.57	93.03	94.28	92.87			53.15	9.65	69.05	83.23	60.06
Deeplabv3+	Extended	177	44	88.44	37.03	93.40	95.55	92.38			79.48	21.49	88.09	89.54	87.47
LinkNet	Deepflood HD			87.87	35.88	92.87	96.19	90.99			76.74	17.70	86.36	88.19	85.36

scenarios, but future applications may require further training on the datasets with some specific characteristics. In the training process, the average computational time per step required for training LinkNet was shorter than that for training Deeplabv3+, suggesting LinkNet potentially more efficient. However, the overall total computational time needed for training a model is also affected by the specific training set used in real practice. Based on the better results tested on the Sazara dataset, this study advanced the results of similar models described in previously published literature. The mean boundary IoU for all test and validation trials were much lower than the mean IoU, which indicates that the segmentation models still lacked accuracy at the boundaries of flood extent.

The results from both validation and testing highlight the consistent influence of environmental factors on the performance of DCNN-based models in flood extent segmentation. During flooding events, instances can arise where a road surface experiences partial flooding, with specific sections that are wetted but not entirely submerged. These sections are prone to being misidentified as flooded areas by the models, leading to an overestimation of flood extent. Furthermore, submerged objects within the floodwater and virtual images produced by specular reflection on still water surfaces can act as distractions, causing the models to struggle in accurately identifying the true extent of the flood. Additionally, the limitations of many surveillance cameras in producing high-quality, colored images under conditions of limited illumination render the models incapable of effectively segmenting flood extent during nighttime scenarios. Since this study is mainly focused on the segmentation of flood extent in good lighting conditions, in order to address the challenges with limited illumination, large varieties of flood images taken in the nighttime should be collected to create a specialized dataset, and segmentation models tailored to nighttime images should be selected for training. This could be a potential direction of study to pursue in the future.

With further approaches to overcome these factors, the flood extent extracted from surveillance camera images can be further processed with georeferencing or the other geospatial analyses and transformed into flood area and depth. Overall, the performance of the two models in this study proves that DCNN-based models can be potentially useful in real-time monitoring of floods in an urban context for engineering purposes.

4.2. Applicability and limitations of surveillance camera in urban flood monitoring

With the real-time images from a network of surveillance cameras in urban areas, the method presented in this study provides the prospect of building a real-time flood monitoring system. However, certain problems regarding applicability require resolution. In the test on real-world camera imagery, the performance of the models did not match the validation results due to inherent limitations of surveillance cameras. The test results show that compromised image resolution in surveillance cameras and the presence of raindrops on the camera lens are two principal technical factors contributing to inaccuracies. The suboptimal resolution of some older surveillance cameras, constrained by technical capabilities, impedes accurate flood extent segmentation. Particularly in instances of storm-induced low atmospheric visibility, the challenge of accurate flood extent extraction amplifies when the initial camera resolution is insufficient. Therefore, urban flood monitoring necessitates high-definition capability for surveillance cameras. The raindrops adhered to the camera lens can substantially compromise the accuracy of the extracted flood extent, persisting even well after the end of a storm event. In this case, surveillance cameras endowed with automated raindrop-cleaning mechanisms are considered the better option. For example, the AI 36X/42X Speed Dome Network Camera of Milesight has optional smart rain-sensing wiper, which can detect and automatically clear the rain at the speed and frequency intelligently adjusted according to the rain, providing sharp imagery during and after a storm event

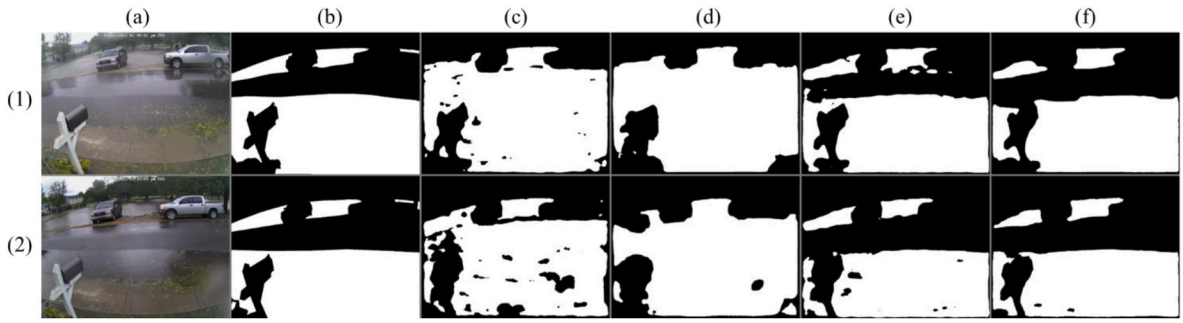


Fig. 7. Comparison of segmenting the flood extent in the WebCOOS dataset (1) during the storm and (2) after the storm using the models trained on the Deepflood HD dataset versus the extended Deepflood HD dataset. In each row, the six images from left to right are: (a) the original image, (b) ground-truth flood extent, the segmented flood extent using (c) Deeplabv3+ trained on the Deepflood HD dataset, (d) LinkNet trained on the Deepflood HD dataset, (e) Deeplabv3+ trained on the extended Deepflood HD dataset, and (f) LinkNet trained on the extended Deepflood HD dataset.

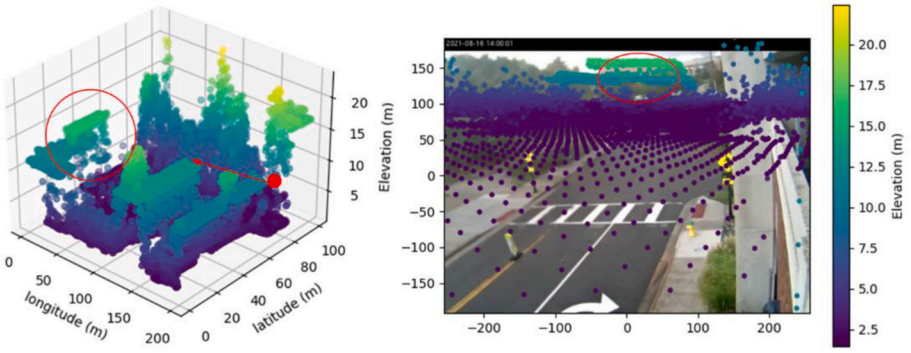


Fig. 8. Schematic diagram of the reprojection of the 1-m DEM of Norfolk, VA to fit the view of the surveillance camera that captured the images in the ODU dataset. Based on the geographic location and the 3-D azimuth angle of the surveillance camera (illustrated with the red point and the arrow), the elevation points can be reprojected to fit the objects in the view of the camera, e.g., the circled building.

(Milesight, 2022).

While the latest cameras offer high-resolution images that are capable of carrying more information, this alone does not ensure that the segmentation models can perform accurate flood extent segmentation. The WebCOOS camera used in this study was a high-resolution web camera. Although the image resolution of the Deepflood HD dataset is four times that of the other training datasets, the models trained on them did not ensure a high accuracy in segmenting flood extent in the test. However, with the inclusion of a small number of sample images of flooding captured by the camera into the training set, the models were able to better capture the distinctive features of flooding conditions at the camera installation site. The results based on the extended Deepflood HD dataset have underscored the efficacy of this approach in enhancing the accuracy of the flood extent segmentation. Based on the distinct characteristics of flooding at each location, this study suggests training the segmentation models individually for each camera, incorporating a small selection of sample images of flooding from the respective camera into the universal foundational training set. Drawing from the test results on the ODU and the WebCOOS datasets, repurposing pre-existing surveillance cameras not originally designated for flood monitoring might present challenges, but individualized model training for high-definition web cameras can be used as a supplementary tool for automated flood monitoring.

4.3. Future applications in an urban flood monitoring system

The method of flood detection using surveillance cameras discussed in this study can potentially play an important role in building a real-time camera-based flood monitoring system in urban environments. High-resolution topographic information, such as 1-m or sub-meter resolution Digital Elevation Model (DEM) or Digital Surface Model (DSM) from Light Detection and Ranging (LiDAR) technology, is widely available in coastal urban areas. It can be reprojected from local coordinates to camera coordinates based on the geographical position and orientation of each specific camera so that the topography can be aligned with the field of view of the camera (Fig. 8). The reprojected elevation data can then be superimposed over the flood extent segmented by the DCNN-based models. By calculating the disparity between the maximum and minimum elevation values within the flood area, an estimation of flood depth becomes feasible. The frequency of consistently capturing images from surveillance cameras ensures temporal resolution, and the spatial distribution of surveillance cameras in an urban area ensures spatial coverage. In this way, a real-time camera-based flood monitoring network can be potentially implemented in urban environments. In addition, water depth estimation based on high-resolution elevation data uses the whole boundary of the segmented flood extent, which can make the estimated flood depth less impacted by any local inaccuracy of segmentation at the boundary. This can resolve the impact of inaccurate boundaries of the flood extent segmented by the models.

5. Conclusion

This study applied Deeplabv3+ and LinkNet, two DCNN-based semantic segmentation models, for the precise segmentation of flood extent in urban flooding images. Based on the training and validation results of online-source datasets, both Deeplabv3+ and LinkNet can extract flood extent accurately, with Deeplabv3+ slightly better. While there are factors that affect the segmentation of flood extent primarily, including specular reflection due to still water surface, high reflectance

from wet road surface, night condition with poor illumination, reduced atmospheric visibility during a storm, and raindrops adhered to the camera lens during and after a storm, both models have the potential to be used in building an urban flood monitoring. Images of urban flooding captured by high-quality cameras that are intended for other purposes can be used in flood monitoring. Considering the intricate and diverse nature of real-world urban flooding images, this study suggests individualized model training for each camera, as opposed to developing a universal model for all cameras. Adding a small number of flood images captured by the camera to the foundational training set can improve the accuracy of the model. In practice, high-definition surveillance cameras with smart rain wipers could mitigate the problem of low atmospheric visibility and adhered raindrops. With additional steps to georeference the segmented flood extent, it would be possible to establish a camera-based flood monitoring framework that can report real-time extent and depth of urban floods. These findings underscore the advancements that this study brings to the development of effective flood monitoring systems in an urban context.

CRediT authorship contribution statement

Yidi Wang: Conceptualization, Methodology, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Yawen Shen:** Conceptualization, Methodology, Resources, Visualization. **Behrouz Salahshour:** Data curation, Methodology, Resources, Software. **Mecit Cetin:** Methodology, Project administration, Resources. **Khan Iftekharuddin:** Funding acquisition, Methodology, Project administration, Resources. **Navid Tahvildari:** Methodology, Project administration, Resources. **Guoping Huang:** Methodology. **Devin K. Harris:** Methodology. **Kwame Ampofo:** Methodology. **Jonathan L. Goodall:** Methodology, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used Chat-GPT in order to improve readability. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

I declare that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

Data availability

The datasets in this study are available in Hydrology Share at the following address: https://www.hydroshare.org/resource/24866122a6ee456c8f7c80aa87a9abcb. The source code for the models in this study are available in Github at the following address: https://github.com/yw5vq/DCNN-Flood-Extent-Segmentation.

Acknowledgements

This paper is based upon work supported by the National Science Foundation Research Traineeship (NRT) Program under Grant No. 1829004 and NSF Award Project SCC-IRG Track 2: Scalable Modeling and Adaptive Real-time Trust-based Communication (SMARTc) System for Roadway Inundations in Flood-Prone Communities under Award No. 1951745. ODU is acknowledged for providing camera videos of urban flooding. SECOORA is acknowledged for leading the WebCOOS program, which collected images of urban flooding in the WebCOOS dataset in this study.

References

Audebert, N., Le Saux, B., Lefèvre, S., 2017. Segment-before-detect: vehicle detection and classification through semantic segmentation of aerial images. Rem. Sens. 9 (4), 368.
Carlson, K., Chowdhury, A., Kepley, A., Somerville, E., Warshaw, K., Goodall, J., 2019. Smart cities solutions for more flood resilient communities. In: 2019 Systems and Information Engineering Design Symposium. IEEE, pp. 1–6. https://doi.org/10.1109/SIEDS.2019.8735625.
Chao, L., Zhang, K., Li, Z., Wang, J., Yao, C., Li, Q., 2019. Applicability assessment of the CASCade Two Dimensional SEDiment (CASC2D-SED) distributed hydrological model for flood forecasting across four typical medium and small watersheds in China. Journal of Flood Risk Management 12, e12518.
Chaudhary, P., D'Aronco, S., Leitão, J.P., Schindler, K., Wegner, J.D., 2020. Water level prediction from social media images with a multi-task ranking approach. ISPRS J. Photogrammetry Remote Sens. 167, 252–262.
Chaurasia, A., Culurciello, E., 2017. Linknet: exploiting encoder representations for efficient semantic segmentation. In: 2017 IEEE Visual Communications and Image Processing (VCIP). IEEE, pp. 1–4.
Chen, L.C., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv preprint arXiv:1706.05587*.
Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision. ECCV), pp. 801–818.
Colosio, P., Tedesco, M., Tellman, E., 2022. Flood monitoring using enhanced resolution passive microwave data: a test case over Bangladesh. Rem. Sens. 14 (5), 1180.
de Silva, A., Zhao, M., Stewart, D., Hasan, F., Dusek, G., Davis, J., Pang, A., 2023. RipViz: Finding Rip Currents by Learning Pathline Behavior. IEEE Transactions on Visualization and Computer Graphics.
Dusek, G., Hernandez, D., Willis, M., Brown, J.A., Long, J.W., Porter, D.E., Vance, T.C., 2019. WebCAT: piloting the development of a web camera coastal observing network for diverse applications. Front. Mar. Sci. 6, 353.
Gareth, J., Daniela, W., Trevor, H., Robert, T., 2013. *An Introduction to Statistical Learning: with Applications in R*. Spinger.
Gebrehiwot, A., Hashemi-Beni, L., Thompson, G., Kordjamshidi, P., Langan, T.E., 2019. Deep convolutional neural network for flood extent mapping using unmanned aerial vehicles data. Sensors 19 (7). https://doi.org/10.3390/s19071486.
Ghiasi, G., Lin, T.Y., Le, Q.V., 2019. Nas-fpn: learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7036–7045.
Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT press.
Goold, B., Van Den Hoven, J., Lim, L., Seck, M., Squires, P., Töpfer, E., 2010. Citizens, Cities and Video Surveillance. European Forum for Urban Security, Paris, France.
Hallegatte, S., Green, C., Nicholls, R.J., Corfee-Morlot, J., 2013. Future flood losses in major coastal cities. Nat. Clim. Change 3, 802–806. https://doi.org/10.1038/nclimate1979.
He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.
Helmrich, A.M., Ruddell, B.L., Bessem, K., Chester, M.V., Chohan, N., Doerry, E., Eppinger, J., Garcia, M., Goodall, J.L., Lowry, C., Zahura, F.T., 2021. Opportunities for crowdsourcing in urban flood monitoring. Environ. Model. Software 143, 105124.
Kegenbekov, Z., Jackson, I., 2021. Adaptive supply chain: demand–supply synchronization using deep reinforcement learning. Algorithms 14 (8), 240.
Landman, Elise, 2022. Duplicate Image Finder. GitHub. https://github.com/elisemercury/Duplicate-Image-Finder.git.
Li, G.F., Xiang, X.Y., Tong, Y.Y., Wang, H.M., 2013. Impact assessment of urbanization on flood risk in the Yangtze River Delta. Stoch. Environ. Res. Risk Assess. 27 (7), 1683–1693.
Li, Y., Martinis, S., Wieland, M., 2019. Urban flood mapping with an active self-learning convolutional neural network based on TerraSAR-X intensity and interferometric coherence. Journal of Photogrammetry and Remote Sensing 152, 178–191. https://doi.org/10.1016/j.isprsjprs.2019.04.014.
Liang, Y., Li, X., Tsai, B., Chen, Q., Jafari, N., 2023. V-FloodNet: a video segmentation system for urban flood detection and quantification. Environ. Model. Software 160, 105586.
Liu, J., Shao, W., Xiang, C., Mei, C., Li, Z., 2020. Uncertainties of urban flood modeling: influence of parameters for different underlying surfaces. Environ. Res. 182, 108929.
Lo, S.W., Wu, J.H., Lin, F.P., Hsu, C.H., 2015. Visual sensing for urban flood monitoring. Sensors 15 (8), 20006–20029. https://doi.org/10.3390/s150820006.
Loftis, J.D., Forrest, D., Katragadda, S., Spencer, K., Organski, T., Nguyen, C., Rhee, S., 2018. StormSense: a new integrated network of IoT water level sensors in the smart cities of Hampton roads, VA. Mar. Technol. Soc. J. 52 (2), 56–67. https://doi.org/10.4031/MTSJ.52.2.7.
Manzoor, A., Patsakis, C., Morris, A., McCarthy, J., Mullarkey, G., Pham, H., et al., 2014. CityWatch: exploiting sensor data to manage cities better. Transactions on Emerging Telecommunications Technologies 25, 64–80. https://doi.org/10.1002/ett.2786.
Milesight, 2022. AI 36X/42X Speed Dome Network Camera. Retrieved from. https://www.milesight.com/product/36x42x-ai-speed-dome.
Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D., 2021. Image segmentation using deep learning: a survey. IEEE Trans. Pattern Anal. Mach. Intell. 44 (7), 3523–3542.
Moy de Vitry, M., Kramer, S., Wegner, J.D., Leitão, J.P., 2019. Scalable flood level trend monitoring with surveillance cameras using a deep convolutional neural network. Hydrol. Earth Syst. Sci. 23 (11), 4621–4634.

Muhadi, N.A., Abdullah, A.F., Bejo, S.K., Mahadi, M.R., Mijic, A., 2020. Image segmentation methods for flood monitoring system. Water 12 (6), 1825.
Muhadi, N.A., Abdullah, A.F., Bejo, S.K., Mahadi, M.R., Mijic, A., 2021. Deep learning semantic segmentation for water level estimation using surveillance camera. Appl. Sci. 11 (20), 9691.
Munawar, H.S., Hammad, A.W., Waller, S.T., 2022. Remote sensing methods for flood prediction: a review. Sensors 22 (3), 960.
Neumann, J.E., Price, J., Chinowsky, P., Wright, L., Ludwig, L., Streeter, R., et al., 2015. Climate change risks to US infrastructure: impacts on roads, bridges, coastal development, and urban drainage. Climatic Change 131 (1), 97–109.
Pally, R.J., Samadi, S., 2022. Application of image processing and convolutional neural networks for flood image classification and semantic segmentation. Environ. Model. Software 148, 105285.
Perez, P., Chemin, T. H. Du, Turpin, E., Clarke, R., 2015. Citizen-driven flood mapping in Jakarta: a self-organising socio-technical system. In: Proceedings -2015 IEEE 9th International Conference on Self-Adaptive and Self-Organizing Systems Workshops. IEEE, pp. 174–178. https://doi.org/10.1109/SASOW.2015.40.
Rosenzweig, B.R., Herreros Cantis, P., Kim, Y., Cohn, A., Grove, K., Brock, J., et al., 2021. The value of urban flood modeling. Earth's Future 9 (1), e2020EF001739.
Sazara, C., Cetin, M., Iftekharuddin, K.M., 2019 October. Detecting floodwater on roadways from image data with handcrafted features and deep transfer learning. In: *2019 IEEE intelligent transportation systems conference (ITSC)*. IEEE, pp. 804–809. https://doi.org/10.1109/ITSC.2019.8917368.
Sazara, C., Salahshour, B., Cetin, M., Iftekharuddin, K., 2022. A deep learning method for floodwater depth prediction on roadways from side-view real and synthetic images of vehicles. Journal of Big Data Analytics in Transportation 4, 85–101.
Shen, Y., Morsy, M.M., Huxley, C., Tahvildari, N., Goodall, J.L., 2019. Flood risk assessment and increased resilience for coastal urban watersheds under the combined impact of storm tide and heavy rainfall. J. Hydrol. 579, 124159.
Tan, M., Le, Q., 2019. Efficientnet: rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. PMLR, pp. 6105–6114.
Taylor, J., Levine, N.S., Muhammad, E., Porter, D.E., Watson, A.M., Sandifer, P.A., 2022. Participatory and spatial analyses of environmental justice communities' concerns about a proposed storm surge and flood protection seawall. Int. J. Environ. Res. Publ. Health 19 (18), 11192. https://doi.org/10.3390/ijerph191811192.
Tomar, Nikhil, 2021. Human Image Segmentation with DeepLabV3Plus in TensorFlow. GitHub. https://github.com/nikhilroxtomar/Human-Image-Segmentation-with-DeepLabV3Plus-in-TensorFlow.git.
Vandaele, R., Dance, S.L., Ojha, V., 2021. Deep learning for automated river-level monitoring through river-camera images: an approach based on water segmentation and transfer learning. Hydrol. Earth Syst. Sci. 25 (8), 4435–4453.
WebCOOS, 2023. https://webcoos.org/. (Accessed 3 May 2023).
Zhou, L., Zhang, C., Wu, M., 2018. D-LinkNet: LinkNet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 182–186.