



ORIGINAL PAPER

Open Access



Scalable flood inundation mapping using deep convolutional networks and traffic signage

Bahareh Alizadeh^{1*}  and Amir H. Behzadan¹ 

Abstract

Floods are one of the most prevalent and costliest natural hazards globally. The safe transit of people and goods during a flood event requires fast and reliable access to flood depth information with spatial granularity comparable to the road network. In this research, we propose to use crowdsourced photos of submerged traffic signs for street-level flood depth estimation and mapping. To this end, a deep convolutional neural network (CNN) is utilized to detect traffic signs in user-contributed photos, followed by comparing the lengths of the visible part of detected sign poles before and after the flood event. A tilt correction approach is also designed and implemented to rectify potential inaccuracy in pole length estimation caused by tilted stop signs in floodwaters. The mean absolute error (MAE) achieved for pole length estimation in pre- and post-flood photos is 1.723 and 2.846 in., respectively, leading to an MAE of 4.710 in. for flood depth estimation. The presented approach provides people and first responders with a reliable and geographically scalable solution for estimating and communicating real-time flood depth data at their locations.

Keywords Deep learning, Traffic signage, Flood, Floodwater depth, Crowdsourcing, Semantic segmentation

1 Introduction

Among weather-related disasters, floods are among the most frequent events causing billions of dollars of damage every year worldwide (Xie et al., 2017). In the U.S. alone, there have been an average of 16.2 weather and climate disasters annually between 2015 to 2020 (Smith, 2021), and nearly one-third of people who live in coastal areas are exposed to elevated coastal hazard risks (United States Census Bureau, 2019). In 2021 alone, 223 flood events occurred in the U.S. (Centre for Research on the Epidemiology of Disasters (CRED), 2022) which surpassed the above-mentioned annual average, leading to a sharp increase in the severe impacts of floods on communities and the built environment. For instance, the estimated cost for Hurricane Harvey in 2017 was \$198 billion, exceeding

that of Hurricane Katrina at \$158 billion in 2005 (or roughly \$194 billion in 2017 dollars) (Hicks & Burton, 2017). In November 2021, a major rainfall event induced a series of floods in northern parts of the U.S. state of Washington and southern parts of the Canadian province of British Columbia. In British Columbia, this flood caused record-breaking insured damage of \$450 million (Insurance Bureau of Canada, 2021). Floods also cause significant damage and loss in other parts of the world. In central China, the July 2021 flooding killed almost 100 people and resulted in more than \$11 billion in economic losses (Wang, 2021). In the same year, deadly floods in Germany resulted in \$40 billion of economic loss. In Belgium, the 2021 floods were the worst in over 100 years, resulting in an estimated damage of \$3 billion (Rodriguez Castro et al., 2022). Past research has identified the key reasons behind intense and widespread floods to be climate change (Alfieri et al., 2017; Arnell & Gosling, 2016; Bjorvatn, 2000; Bowes et al., 2021; Sahin & Hall, 1996; Ward et al., 2014a, b), deforestation (Bradshaw et al., 2007; Sokolova et al., 2019), growing population

*Correspondence:

Bahareh Alizadeh
bahareh.alizadeh@tamu.edu

¹ Department of Construction Science, Texas A&M University, College Station, TX, USA

(Changnon, 2000; Winsemius et al. 2016; Wing et al. 2018), and rapid urbanization (Singh & Singh, 2011; Suriya & Mudgal, 2012). A study by Forzieri et al. (2017) has shown that weather-related natural hazards could affect about two-thirds of Europe's population annually over the next century. Huang et al. (2015) investigated the combination of different climate scenarios in the five large basins in Germany and found that most rivers in the study area could experience more 50-year floods.

The monetary damage of floods to the infrastructure and housing stock is primarily calculated using key indicators such as water depth and building characteristics (Figueiredo et al., 2018; Gerl et al., 2016; Romali & Yusop, 2021). For example, the depth-damage function, used by the U.S. Army Corps of Engineers and the Federal Insurance Administration, estimates structural damages as a percentage of structure's value for a given water depth above or below the first occupied level of the structure (Davis & Skaggs, 1992; Wing et al., 2020). Likewise, in the aftermath of a flood event, effective response operations are highly dependent on having access to rapid and accurate flood depth data. Flood mapping systems are sought to provide flood depth data in a particular region or zone over the temporal scale. In the U.S., flood maps prepared by the Federal Emergency Management Agency (FEMA) are commonly used for estimating the extent of flood inundation. However, approximately 75% of FEMA flood maps have not been updated in the last five years, and 11% of them are significantly outdated (dating back to the 1970s and 80 s) (Eby and Ensor, 2019). While advances in urban sensing and computing technologies has led to an uptake in the use of sophisticated remote sensing methods (e.g., monostatic radars) for flood depth mapping, the communication of captured data to the public is often restricted or lagged (Chew et al., 2018). Overall, current flood mapping systems have key limitations, and existing data sources are sparse and not inclusive of many at-risk communities (resulting in data deserts) (Cutter et al., 2003; Van Zandt et al., 2012; Forati & Ghose, 2021; Arabi, 2021). This makes current flood mapping methods inefficient for delivering high-resolution, accurate, and real-time flood depth data to diverse stakeholders who live in flood-prone regions. This paper proposes a novel approach that enables the on-demand estimation of floodwater depth using deep neural networks in photos depicting submerged traffic signage. Compared to other flood mapping systems, the key advantage of this method is its simplicity (ease of use), accuracy, speed, and coverage (by significantly increasing the number of points where floodwater depth can be calculated and logged).

2 Literature review

2.1 Existing flood inundation mapping systems

Much of the information utilized for urban flood depth mapping is extracted from water level sensors and gauges (Crabit et al., 2011), remote sensing data (Feng et al., 2015; Perks et al., 2016), crowdsourcing platforms (Wang et al., 2018), and video surveillance (Liu et al., 2015). However, conventional gauge-based ground monitoring systems could result in uneven coverage of floodplains, and accrue significant sensor installation and operation costs (Dong et al., 2021; Lo et al., 2015). Also, since the estimated flood depth in each station is measured relative to the station level, further comprehensive data pooling is required for flood mapping (Lo et al., 2015). Thanks to recent advancements in remote sensing, flood maps can be generated with high-resolution spatial and temporal data. Examples include terrain data from light detection and ranging (LiDAR) (Brown et al., 2016), radar-based precipitation depths and synthetic aperture radar (SAR) fine-resolution images (Schumann et al., 2011), advanced streamflow measurement (Merwade et al., 2008), digital elevation model (DEM) with an X-band sensor (e.g., TerraSAR-X and COSMO-SkyMed Constellation) (Strozzi et al., 2009; Pulvirenti et al., 2011), and short-wavelength microwaves transmitted by satellites (Frappart et al., 2005). Despite some advantages, the reliable application of these methods can be hindered by several factors. For example, data collected from satellites often suffer from restrictions such as orbital cycles and inter-track spacing of satellite movements (Stone et al., 2000). Also, while LiDAR terrain data can be used as a standalone flood inundation mapping approach, filtering LiDAR data in dense urban areas is challenging due to the complex urban landscape (Wedajo, 2017), therefore such data must be juxtaposed with cross-sectional field surveys (Klemas, 2015), which demands additional effort to combine and leverage multiple data streams to generate a complete map (Stone et al., 2000; Kamari & Ham, 2021). Another major problem with LiDAR-based flood mapping is the difficulty of matching LiDAR data with hydraulic models especially in high flood depth (Merwade et al., 2008). Near-real-time SAR flood maps (Shen et al., 2019) can be generated using a radar-based precipitation approach based on data collected from the Next Generation Weather Radar (NEXRAD) system which currently comprises 160 sites across the U.S. (National Oceanic and Atmospheric Administration, 2022). Although radar-based precipitation data uses sophisticated distributed hydrological models, a significant degree of uncertainty is associated with the outcome due to the use of variables that are averaged out over space and time (Alinezhad et al., 2020; Merwade et al., 2008; Zhou et al., 2021). In contrast to the inherent uncertainty

in hydraulic and hydrologic models, probability-based predictors such as the Bayesian approach (Beck & Katafygiotis, 1998) used in conjunction with SAR imagery track changes in flood water levels, and use adjustable weights to progressively update model's estimates and improve prediction accuracy (Hsu et al., 2009).

Another practical barrier to using conventional remote sensing is its insensitivity to data variations and noise in densely vegetated areas, adverse weather, and humid tropical regions with excessive cloud coverage (Asner 2001). Smooth impervious surfaces and shadows in urban areas can also cause over-detection in SAR imagery, thus requiring more information to describe the geometry, orientation, building materials, as well as the direction of radar illumination (Ferro et al., 2011; Shen et al., 2019). Limited access to continuous monitoring, coupled with observing flood depth in only sparsely-located fixed points are among other barriers to the effective application of most remote sensing methods (Lo et al., 2015). In general, the accuracy of existing flood mapping systems is dependent on several factors such as the method used for collecting and describing topography, estimate of the design flow, inherent model uncertainties, noise in input data, and the calibration approach (Merwade et al., 2008; Vojtek et al., 2019).

In recent years, researchers have investigated the prospect of quantifying flood depth through visual examination and detection of key objects in images using computer vision and artificial intelligence (AI). Most object detection models are one-stage or two-stage detectors (Zhan et al., 2021). Jiang et al. (2019), for example, measured the waterlogging depth in videos using single-shot object detection (Liu et al., 2016) by comparing detected ubiquitous reference objects (e.g., traffic buckets) before and after a flood event, achieving a root mean squared error (RMSE) of 2.6 cm (or 1.02 in.) with an average processing time of 0.406 s per video frame. Cohen et al. (2019) estimated the depth of flood for coastal (using a 1-m DEM) and riverine (using a 10-m DEM) locations and reported an average absolute difference of 18–31 cm (or 7–12 in.). Moy de Vitry et al. (2019) re-trained a deep convolutional neural network (CNN) on videos of flooded areas to detect floodwater in surveillance footage (Liu et al., 2015). However, the applicability of this approach was limited by the field of view of the surveillance cameras. Chaudhary et al. (2020) estimated flood depth through analyzing submerged objects in images collected from social media, and comparing them with the average human height (as a reference object). In their method, however, the distance between the location of the camera and the location of detected objects was not considered, which increased the error of projecting extracted flood depth data onto flood inundation maps.

To remedy this problem, they also proposed (but did not accomplish) to use a computer vision technique called monoplottting (Golparvar & Wang, 2021) to calibrate the data based on the pixel-level correlation between DEM and the input photo (Marco et al., 2018). Park et al. (2021) estimated flood level with a precision of over 89% by analyzing images of submerged vehicles, and obtained a mean absolute error (MAE) of 6.49 cm (or 2.5 in.). Pally and Samadi (2021) used YOLOv3 (you-only-look-once version 3), Fast R-CNN (region-based CNN), Mask R-CNN, SSD MobileNet (single shot multibox detector MobileNet), and EfficientDet (efficient object detection) to detect the water surface in images of flooded areas. Hosseiny (2021) used U-Net (a CNN model) to estimate flood depth by comparing images of rivers, and achieved a maximum error of 2.7 m. Other proposed methods of flood depth estimation include taking bridges as measurement benchmarks in a study by Bhola et al. (2018), drawing a hypothetical line on walls visible in the camera footage (Sakaino, 2016), using a ruler in riverine areas (Kim et al., 2011), and comparing virtual markers determined by the operator in video frames (Lo et al., 2015). However, past research has particularly relied on in-situ measurements and site-specific calibration which require the pre-installation of target objects in the study area (Moy de Vitry et al., 2019). In our past work (Alizadeh et al., 2021; Alizadeh Kharazi & Behzadan, 2021), we proposed a method to remotely estimate flood depth by analyzing images of traffic signs as physical landmarks, considering that traffic signs are omnipresent and have standardized sizes in most parts of the world. In a nutshell, a combination of deep learning and image processing techniques was utilized to estimate flood depth by comparing crowdsourced photos of traffic signs before and after flood events, yielding an MAE of 12.62 in. for floodwater depth estimation on our in-house Blupix v.2020.1 dataset. Specifically, we utilized Mask R-CNN (which was pre-trained on the Microsoft COCO dataset (Lin et al., 2014) for detecting stop signs in photos. Consequently, image processing models (e.g., Canny Edge detection and Hough transform) were utilized to detect sign poles by searching the area underneath the traffic sign for near-vertical lines. The accuracy of the previous approach was, however, limited by factors such as image quality, noise, illumination, and excessive degree of pole tilt. In this paper, we expand our previous work by using semantic segmentation for detecting not only the traffic signs but also their poles aiming at increasing the accuracy and computational speed of the model. Specifically, this paper improves the pole detection outcome by training a neural network on 800 images with annotated stop signs and poles under various visual conditions. The performance of the new approach is presented

on an in-house dataset which demonstrates a significant improvement in terms of accuracy and computational speed. To generate a high-resolution flood inundation map using the proposed approach, a large number of images should be collected and analyzed. Therefore, a crowdsourcing application (named Blupix (Blupix, 2020) was developed and launched as part of the preliminary work that led to this paper. The primary purpose of this application is to facilitate the collection of flood photos by engaging ordinary people in affected areas (particularly in underserved neighborhoods and municipalities), and create an easy-to-use interface to deliver near real-time flood depth information to various stakeholders and communities. Moreover, a mobile app was developed with a built-in computer vision model to enable real-time estimation of flood depth in urban areas (Alizadeh & Behzadan, 2022a). In the field of disaster response and mitigation, crowdsourcing is an invaluable tool where people and stakeholders can report and share needed information collected from their surroundings, thus enhancing the spatiotemporal scalability and inclusiveness of the input data (Assumpção et al., 2018; See, 2019).

2.2 Convolutional neural networks for object detection

To detect target objects in an input image, various object detection methods have been previously used. Conventionally, approaches such as deformable parts models detected objects using a sliding window and a classifier that ran over the entire image (Felzenszwalb, 2010). More recently, Girshick et al. (2014) introduced region-based CNN (R-CNN) with improved performance in object detection. R-CNN can achieve approximately 47 frames per second (FPS) detection speed by considering several regions of interest in the input image and classifying those regions for containing target objects (Gandhi, 2018). In the R-CNN architecture, independent features are extracted from each region proposal separately, resulting in a long processing time. To overcome this issue, faster variants of R-CNN were also proposed, including Fast R-CNN (Girshick, 2015) which is about 213 times faster than R-CNN, Faster R-CNN (Ren et al., 2017) which is about 250 times faster than R-CNN, and Mask R-CNN (He et al., 2017). Later, more real-time detectors were proposed such as single shot detector (SSD) (Liu et al., 2016) in which proposal generation is eliminated and anchor boxes and feature maps are pre-defined. Similarly, the YOLO model was proposed by Redmon et al. (2016) in which bounding boxes and class probabilities are predicted in one round of image evaluation using a single neural network. Comparing the performance and computation time of various object detection methods reveals that YOLO can perform in real-time with sufficiently high accuracy, while being

less computationally expensive. Also, YOLO models can be easily converted to lighter versions, such as Tiny YOLO (Redmon et al., 2016) which is ideal for launching on mobile devices, one of the future directions of this research, i.e., large-scale crowdsourcing of floodwater depth data collection and analysis.

3 Methodology

The following sections present detailed descriptions of the flood depth estimation technique, data preparation (including augmentation), model training and validation, and performance measurement.

3.1 Flood depth estimation using street photos of traffic signs

According to the manual on Uniform and Traffic Control Devices (Federal Highway Administration, 2004), stop signs installed in U.S. roads should have a standardized width and height of 30 in. (on single-lane roads) or 36 in. (on multi-lane roads and expressways). Since our scope is residential areas, the focus of this research is stop signs in single lane streets that are 30 in. in width and height. To estimate flood depth in a particular location (described by a unique longitude and latitude), paired photos of a single stop sign before and after a flood are needed. As shown in Fig. 1a, knowing the height of the octagonal shape of the stop sign in both pixels (s) and inches (30"),



Fig. 1 Flood depth estimation in a paired **a** pre-flood photo and **b** post-flood photo (base photo in **b**) is courtesy of Erich Schlegel/ Getty Images)

a constant ratio r corresponding to the number of inches per pixel in the pre-flood photo is calculated. Using this ratio, the full length of the pole in inches is determined as the product of r and the pole length in pixels (p). Similarly, in Fig. 1b, knowing the height of the octagonal shape of the stop sign in pixels (s') and inches ($30''$), the number of inches corresponding to one pixel in the post-flood photo is obtained as a constant ratio r' . Using this ratio, the length of the visible parts of the pole (above the waterline) in inches is calculated by multiplying r' and the pole length in pixels (p'). It must be noted that ratios r and r' may not necessarily be equal since pre- and post-flood photos can be taken from different angles and distances from the stop sign.

3.2 Object detection model for pole detection

For visual recognition of stop signs and their poles, a robust and accurate object detection model is desired. Moreover, to implement the flood depth estimation technique on mobile devices, the model should be computationally light. To satisfy these two design conditions, we utilize YOLOv4 (Bochkovskiy et al., 2020) for stop sign and pole detection. YOLOv4 is fast and accurate, and features a light version (a.k.a., Tiny YOLO) for implementation on mobile devices. Other object detection models such as RetinaNet-101–500 (Lin et al., 2017a, b), R-FCN, SSD321 (Liu et al., 2016), and DSSD321 (Fu et al., 2017) achieve mean average precision (mAP) of 53.1% (at 11 FPS), 51.9% (at 12 FPS), 45.4% (at 16 FPS), and 46.1% (at 12 FPS) on the Microsoft COCO dataset (Lin et al., 2014), respectively. By comparison, YOLOv3-320, YOLOv3-416, and YOLOv3-608 models (Redmon & Farhadi, 2018) yield mAP of 51.5% (at 45 FPS), 55.3% (at 35 FPS), and 57.9% (at 20 FPS) on the same dataset, respectively. The term mAP is a metric used to evaluate

the performance of object detection models, and higher mAP indicates higher accuracy of the model (Henderson & Ferrari, 2017; Robertson, 2008). YOLOv4 surpasses YOLOv3 in terms of speed and accuracy, by achieving 65.7% mAP at 65 FPS on the Microsoft COCO dataset. This superior performance is primarily the result of using a different backbone in the YOLOv4 model. Particularly, the model utilizes the cross-stage-partial-connections (CSP) network with Darknet-53 (Wang et al., 2020) as the backbone for more efficient feature extraction. As shown in Fig. 2, this backbone extracts essential features from the input image, which are then fused in the neck of the YOLO model. The neck is comprised of layers that collect feature maps from different stages. This part of the model consists of two networks, namely spatial pyramid pooling (SPP) (He et al., 2015) and path aggregation network (PANet) (Liu et al., 2018). The neck consists of several top-to-bottom paths and bottom-to-top paths that better propagate layer information. Similar to the head of YOLOv3, the head of YOLOv4 adopts the feature pyramid network (FPN) (Lin et al., 2017a, 2017b), predicts object bounding boxes, and outputs the coordinates along with the widths and heights of detected boxes (Redmon and Farhadi, 2018) through three YOLO layers.

3.3 Pre-trained model

The YOLOv4 model is pre-trained on the publicly available Microsoft COCO dataset to detect 80 object classes (Lin et al., 2014). To train the adopted model on the target dataset in this study, transfer learning is used, which is a validated approach for training a CNN model on a relatively small dataset (a.k.a., target dataset) by transferring pre-defined weights (that the network has learned when trained on a large dataset) to allow the model to detect relevant intermediate

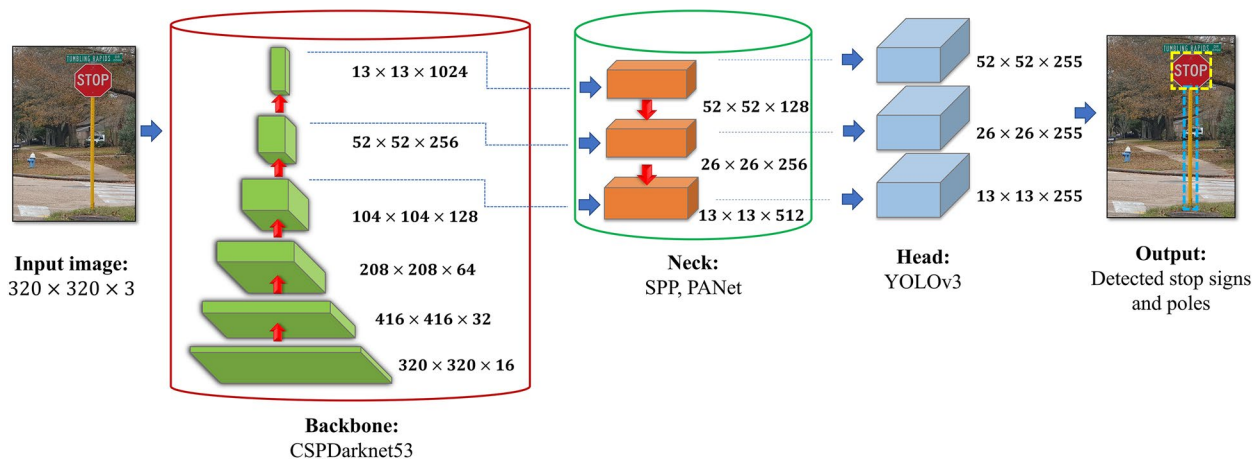


Fig. 2 The architecture of the YOLOv4 network adopted in this research

features (Gao & Mosalam, 2018; Han et al., 2018; Husain et al., 2018; Tammina, 2019). The dataset used for training has images containing two classes: stop sign and pole. This reduces the output size of YOLO layers from 80 to 2 classes. Using transfer learning, all network weights except those of the last three YOLO layers are kept constant (i.e., frozen). At the beginning of the training process, the weights of the three YOLO layers are randomly selected, and then constantly optimized with the goal of maximizing the mAP for pole and stop sign detections. At these optimal values, the model is neither overfitting (i.e., unable to detect objects in new data due to overly learned features in the training set and forgetting general features) nor underfitting (i.e., unable to detect objects in training data and new data due to limited features that were learned).

3.4 Clustering the training set

YOLO models use pre-defined anchor boxes, which are a set of candidate bounding boxes with fixed width and height initially selected based on the dataset, and subsequently scaled and shifted to fit the target objects (Ju et al., 2019). The YOLOv4 model, in particular, utilizes nine anchor boxes. Therefore, all 1,262 ground-truth boxes in the training set (containing instances of both stop sign and pole classes), shown in Fig. 3a, are clustered into nine groups using k -means clustering ($k=9$) (Redmon and Farhadi, 2018). The centroids of these nine clusters are used to define nine anchor boxes, as illustrated in Fig. 3b.

3.5 Training the model

Following Bochkovskiy (2020), the adopted YOLOv4 model is trained for 4,000 iterations (2,000 iterations for each class), with a learning rate of 0.001 using Adam optimizer (Kingma & Ba, 2014), with a batch size of 1 and subdivision of 64. The Darknet-53 (backbone of this model) is built in Windows on a Lenovo ThinkPad laptop computer with 7 cores, 9750H CPU, 16 GB RAM, and Nvidia Quadro T1000 GPU with a 4 GB memory. The network resolution (i.e., image input size) is reduced to the size of $320 \times 320 \times 3$ to lower computational cost and time. The total processing time for training the model is approximately 12 h, with an average loss of 0.567.

Random and real-time data augmentation is automatically applied to the training set to increase the size of training data by creating slightly modified copies of existing images. Past studies have investigated various approaches to data augmentation. In particular to the YOLO architecture, Kang et al. (2019) changed the hue, saturation and exposure of images for training a Tiny YOLO model to detect fire. Ma et al. (2020) applied color jittering and saturation, exposure, and hue change for augmenting images of thyroid nodules for training a YOLOv3 model. Koirala et al. (2019) augmented images of fruits for training a YOLO model by modifying hue, saturation, jitter and multiscale. Lastly, Niu et al. (2020) applied image mosaic, horizontal flipping, and image fusion for augmenting images of sanitary ceramics for training a Tiny YOLO model. In this study, hue, saturation, and exposure of training samples are changed within $[-18...+18]$, $[0.66...1.5]$, and $[0.66...1.5]$, respectively as recommended by Bochkovskiy et al., (2020). Also, a Jitter (random image cropping and resizing by changing the

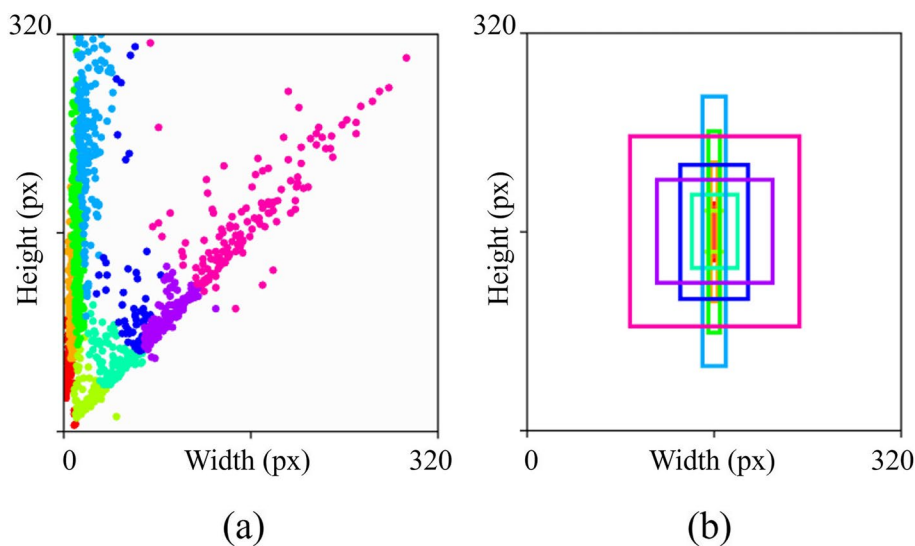


Fig. 3 **a** Nine clusters ($k=9$) corresponding to the training set, and **b** Recalculated anchor boxes for the in-house dataset

aspect ratio) of 0.3 is implemented for data augmentation. The maximum jitter allowed in data augmentation is 0.3 (Ma et al., 2020). Moreover, 50% of images are flipped horizontally but no image is flipped vertically (Hu et al., 2020). Lastly, 50% of images are augmented with a mosaic by combining four different images into one image (Hao & Zhili, 2020).

3.6 Validating the model

To prevent overfitting (i.e., a model that is exactly fitted to the training data, preventing it from correctly detecting new data), model performance is monitored on validation sets using a fivefold cross validation approach (Browne, 2000; Islam et al., 2020; Lyons et al., 2018; Seyrfar et al., 2021). Using this approach, 160 photos are randomly drawn (without replacement) from the training set (20% of the total of 800 images in the training set) for five times as validation sets, and the remaining photos are used for model training. The model is then trained on each training set for 4,000 iterations and validated on the corresponding validation set. The number of epochs is the number of iterations divided by the number of images over the batch size. With the batch size of 1, there are a total of 5 epochs in 4,000 iterations. During the training process, the highest mAPs on the validation sets along with the corresponding number of iterations are saved. Next, average performance is calculated as the average of obtained mAP values across all validation sets. The optimum number of iterations is also computed as the average of the best number of iterations (corresponding to the highest mAP) in all validation sets. In this study, the average mAP and average iteration numbers achieved in fivefold cross validation are 97.04% and ~ 3,000 (since model weights are saved at every 1,000 iterations, the optimum number of iterations is rounded to 3,000 to be exact). This means that after 3,000 iterations, the model shows a tendency to overfit to the training data. Ultimately, the model is trained on the entire training and validation

sets for the obtained optimum number of iterations (i.e., 3000), and the mAP at 3000th iteration is reported. The set of network weights saved at this number of iterations is marked as optimum and used for testing the model. Table 1 shows the validation output of the trained model on each validation set.

3.7 Tilt correction

Over time or as a result of floodwater flow, traffic signs can be tilted in any direction, leading to the underestimation of the submerged pole height by the YOLOv4 model (since detected bounding boxes are not tilted), and eventually erroneous floodwater depth calculation. In cases where both pre- and post-flood photos are tilted by the same angle, floodwater depth calculation is not impacted by the tilt. However, if the degrees of tilt differ between pre- and post-flood photos of the stop sign, pole length estimation should be adjusted prior to calculating the floodwater depth. In Fig. 4, pre-flood and post-flood stop signs are presented with unequal tilt degrees (α degrees for pre-flood stop sign and β degree for post-flood stop sign). For each photo, the actual pole length is the reverse projection of the height of the detected bounding box (P and P') by the degree of tilt (α and β). Eq. 1 presents the calculation of flood depth considering unequal degrees of tilt for a given paired stop sign photo.

$$W = P \left(\frac{\cos \alpha}{\cos \beta} \right) - P' \tag{1}$$

To automatically detect the degree of sideways tilt, a tilt correction technique is implemented and applied before stop sign and pole detection. By visually inspecting the photos in the dataset, it is observed that the maximum tilt does not exceed 25°. Thus, we select a range of [-25°... +25°] for tilt correction. Next, as shown in Fig. 5, the input image is rotated from 0 to -25° clockwise (in 5° intervals) and from 0 to +25° counterclockwise (in 5°

Table 1 The highest average performance of the trained model on five validation sets using fivefold cross validation (S: Stop sign; P: sign pole; S + P: stop sign and sign pole)

Metric	Class	Validation set					Average of 5 validation sets
		1	2	3	4	5	
Average IoU	S + P	83.74%	85.05%	86.82%	85.92%	84.75%	85.26%
Precision @ 0.25 conf	S + P	95.00%	97.00%	97.00%	96.00%	97.00%	96.40%
Recall @ 0.25 conf	S + P	94.00%	97.00%	96.00%	95.00%	94.00%	95.20%
AP	S	97.37%	97.37%	97.37%	97.37%	97.37%	97.37%
	P	97.37%	97.37%	97.37%	97.21%	94.20%	96.70%
mAP @ 0.50	S + P	97.37%	97.37%	97.37%	97.29%	95.78%	97.04%
No. of iterations	S + P	3,000	3,500	2,800	3,500	2,400	3,025

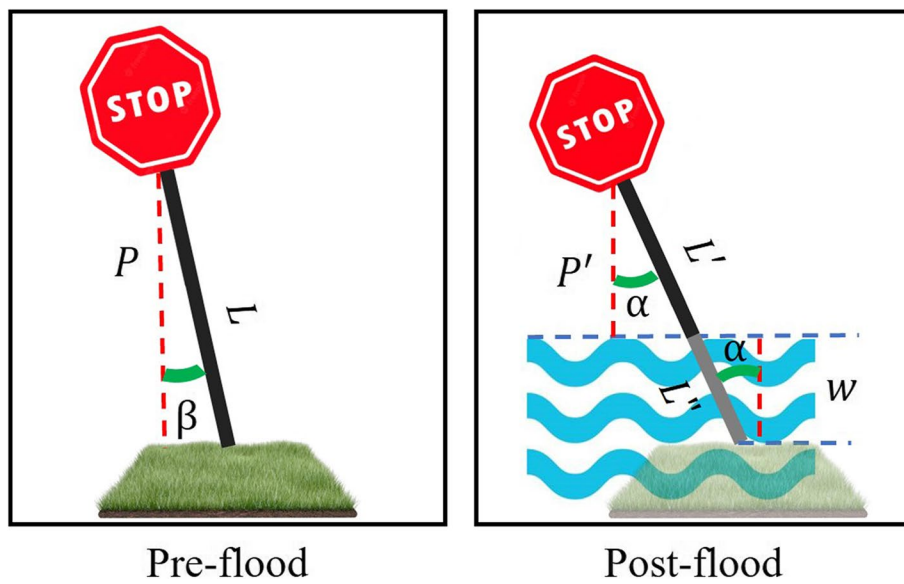


Fig. 4 Adjusting flood depth estimation for pre-flood and post-flood stop signs with unequal degrees of tilt

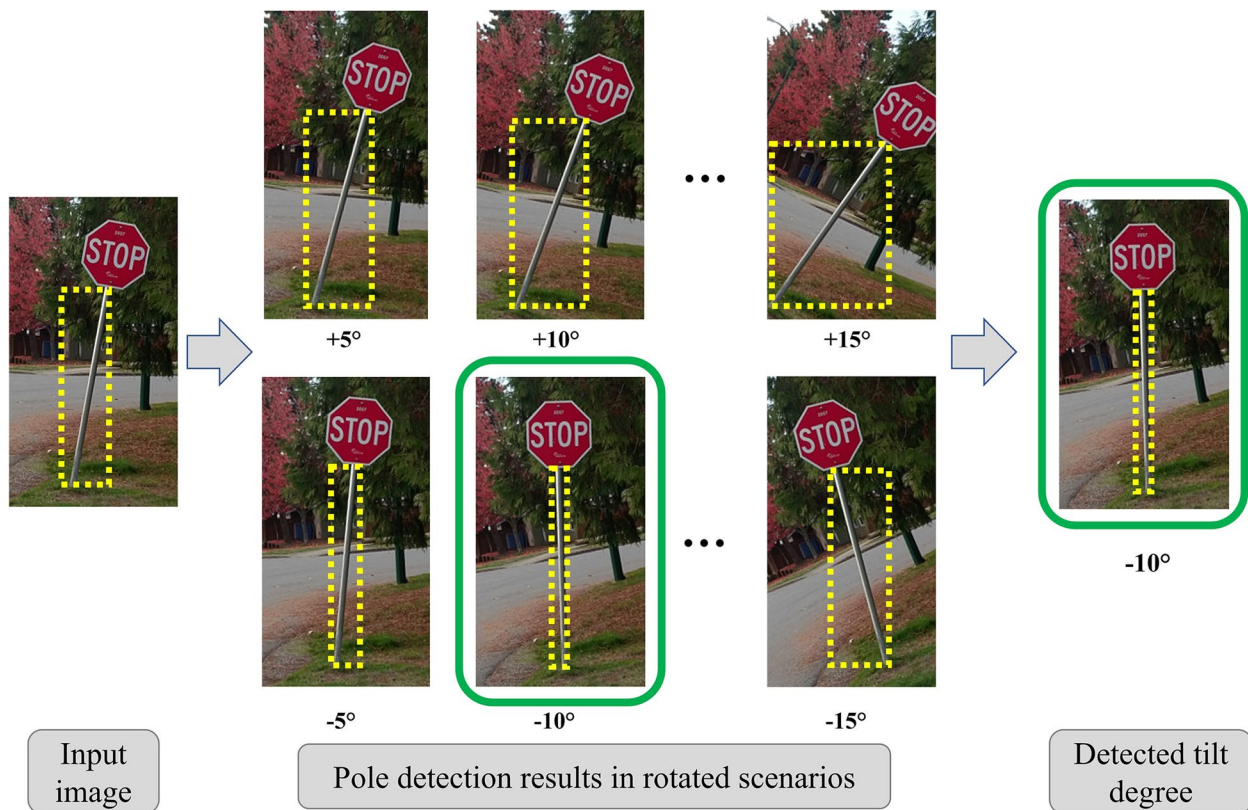


Fig. 5 Tilt detection approach (base photo is courtesy of KjzPhotos/Shutterstock)

intervals). Generated images are then processed by the trained YOLO model for stop sign and pole detection. The image with the minimum width of the pole detection bounding box is ultimately selected as the one containing the most vertical pole. Consequently, the degree of tilt that was applied to the original image to generate this image is calculated as the tilt angle (β degrees for pre-flood stop sign and α degree for post-flood stop sign).

3.8 Model performance

In object detection, a commonly used metric for measuring model performance is mAP (Ren et al., 2017; Turpin & Scholer, 2006). The basis of mAP calculation across all classes is the average precision (AP) for each individual class. In turn, the AP for any given class is measured using intersection over union (IoU) (Javadi et al., 2021; Kido et al., 2020; Nath & Behzadan, 2020), which is calculated for each detection based on the overlap between the predicted bounding box (B') and the ground-truth bounding box (B) (Eq. 2). Following this, the detected object is classified as correct (if the IoU is above a pre-defined threshold, typically 50%), or incorrect (if the IoU is below the threshold) (Alizadeh & Behzadan, 2022b; Alizadeh et al., 2022; Nath & Behzadan, 2020; Zhu et al., 2021). Based on the correctness of the detected object, true positive (TP; correct classification to a class), false positive (FP; incorrect classification to a class), and false negative (FN; incorrect classification to other classes) cases are counted. Next, Eq. 3 and Eq. 4 are used to calculate precision (model's ability to detect only relevant objects) and recall (model's ability to detect all relevant classes) based on TP, FP and FN for each class (Guo et al., 2021; Mao et al., 2021; Padilla et al., 2020; Xu et al., 2021). It must be noted that true negative cases are not considered in object detection when measuring model performance, as there are countless number of objects (belonging to a large number of classes) that should not be detected in the input image (Padilla et al., 2020).

$$IoU = \frac{B' \cap B}{B' \cup B} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

To calculate the AP of any object class, all detections are initially sorted based on their confidence scores in

descending order, and Eq. 5 is then applied. In this Equation, N refers to the total number of detected bounding boxes, i refers the rank of a particular detection in the sorted list, P_i refers to the precision of the i th detection, and Δr is the change in recall between two consecutive detections i th and $(i + 1)$ th. Finally, mAP is calculated as the average precision over all classes, as formulated by Eq. 6 (Lyu et al., 2019; Nath & Behzadan, 2020).

$$AP = \frac{1}{N} \sum_{i=1}^N P_i \Delta r_i \quad (5)$$

$$mAP = \sum_{i=1}^N AP_i \quad (6)$$

3.9 Flood depth estimation

Flood depth is estimated using the YOLOv4 model trained on photos of stop signs taken before and after flood events. The general framework for detecting stop signs and their poles in pre- and post-flood photos and estimating flood depth is illustrated in Fig. 6. As shown in this Figure, paired pre-flood and post-flood photos of a stop sign are processed by presenting them to the model as two separate inputs. The model then detects the stop sign and its pole in each image, and measures the length of the visible part of the detected poles using geometric calculations based on the size of stop signs (Sect. 3.1). Next, the depth of floodwater at the location of the stop sign is estimated as the difference between pole lengths in pre- and post-flood photos.

In addition to evaluating the model performance in stop sign and pole detection, its ability to estimate flood depth must be assessed. The literature in this field has used MAE as an informative metric to describe the discrepancy in flood depth estimation (Chaudhary et al., 2019; Cohen et al., 2019; Park et al., 2021; Alizadeh Kharazi and Behzadan, 2021; Alizadeh et al., 2021). In this study, the error of pole detection in pre-flood and post-flood photos is determined as the difference between ground-truth pole length (l^g) and detected pole length (l^d). The absolute error in a single image is then calculated as the cumulative error in pre-flood and post-flood photos. Since we measure the depth of flood based on the difference between pole lengths in M paired pre- and post-flood photos, the MAE for flood depth estimation can be determined as the average of absolute errors in all paired photos (Eq. 7).

$$MAE = \frac{1}{M} \sum_{i=1}^M \left| \left(l_{pre}^g - l_{pre}^d \right) + \left(l_{post}^g - l_{post}^d \right) \right| \quad (7)$$



Fig. 6 Workflow for estimating flood depth in a sample paired pre- and post-flood photos (base post-flood photo is courtesy of Erich Schlegel/ Getty Image)

4 Data description

Table 2 shows a detailed breakdown of the image datasets used to train and test the model. The data used for training the adopted YOLO model to detect stop signs and sign poles consists of two image datasets: pre-flood photos and post-flood photos. Each dataset is described in detail in the following sections. The data used for testing the YOLO model comprises 176 pre-flood and 172 post-flood photos drawn from the Blupix v.2021.1 dataset (containing 224 pairs of pre- and post-flood photos, i.e., 448 photos in total), by filtering out photos with very low resolution or those in which entire poles are not captured. The Blupix v.2021.1 dataset is an expanded version of the Blupix v.2020.1 (Alizadeh Kharazi & Behzadan, 2021) which contained 186 pairs of pre- and post-flood photos; i.e., 372 photos in total.

4.1 Pre-flood training dataset

There are publicly available large-scale datasets containing images of stop signs. The pre-flood training dataset is generated by extracting a subset of photos of stop signs with the entire stop sign poles in the photo from the Microsoft COCO dataset (Lin et al., 2014). Figure 7 shows examples of stop sign photos in different countries, with different forms and pole shapes, extracted from the Microsoft COCO dataset. Although all stop sign objects were already annotated in the Microsoft COCO dataset, it was found that some annotations were not as accurate as expected. For example, shapes of masks drawn over stop signs were not always octagonal. To resolve this problem, all extracted images were re-annotated by a trained annotator. Ground-truth bounding boxes were determined by manual labeling, e.g., outlining stop signs and poles by polygons using LabelMe software (Wada, 2016). Although annotating images with rectangular bounding boxes was sufficient for implementing the YOLO model, it

Table 2 Overview of the training and test datasets

	Subset	Source	Language	Count	Total	
Training set (Blupix v.2021.2)	Post-flood	Web-mining	English	196	800	
			Spanish	6		
			French	55		
	Pre-flood	COCO	Turkish	13		
			Synthetic	English	64	
			Non-labeled	English	61	
Test set (Blupix v.2021.1)	Post-flood	Web-mining	English, Spanish, French, Turkish, Arabic, etc	334	348	
			Non-labeled	English		71
			English	164		
	Pre-flood	Web-mining	French	8		
			English	168		
			French	8		



Fig. 7 Photos of various stop signs in different forms and languages available from the Microsoft COCO dataset

was ultimately decided to annotate using masks to achieve more accurate shapes (octagonal masks over stop signs and quadrilateral masks over sign poles) and to facilitate the generalizability of the annotations for future studies. At the conclusion of the annotation step, all masks were converted to bounding boxes which is the required input format of the YOLO model. Sample annotated pre- and post-flood photos are shown in Fig. 8. These photos depict a stop sign at the intersection of Tumbling Rapids Dr. and Hickory Downs Dr. in Houston, Texas. The post-flood photo was obtained via crowdsourcing after Hurricane Harvey in (2017), and the pre-flood photo was taken by the authors on January 23, 2021.

4.2 Post-flood training dataset

For post-flood photos, an in-house dataset is created which contains 270 web-mined photos of flooded stop signs. Web-mining is conducted using related keywords such as “flood stop sign”, “flood warning sign”, and their translations in three other languages (i.e., Spanish, French, Turkish). To increase the generalizability of the

model, we also include photos taken from the back side of the sign, those depicting tilted poles or reflections in water, as well as photos taken in daylight or nighttime, photos with clear or noisy backgrounds, and photos taken in different weather conditions. Additionally, to minimize detection error, the dataset is further balanced by generating synthetic training data (Feingersh et al., 2007; Hu et al., 2021; Tremblay et al., 2018; Shaghaghian & Yan, 2019; Nazari & Yan, 2021). In particular, a new set of post-flood photos depicting flooded traffic signs (other than stop signs) is imported in a photo editing tool where the depicted traffic signs are replaced with stop signs (keeping the pole unchanged). Using this method, 64 synthetic images are added to the dataset which results in a total of 334 post-flood photos.

4.3 Non-labeled objects

Since the model is trained on stop signs in different languages (with white text on a red octagonal shape), it could learn overly detailed features that may not be generalizable, resulting in potentially false positive cases. To



Fig. 8 Sample annotated **a** pre-flood photo, and **b** post-flood photo (base photo in (b) is courtesy of Erich Schlegel/Getty Images)

resolve this problem, the training set is further enriched with samples that are visually proximal to stop signs but are not stop signs, to force the model to learn distinctive characteristics specific to stop signs while avoiding false positives. In particular, 71 web-mined pre-flood photos and 61 web-mined post-flood photos of other traffic signs similar to stop signs (such as “Do Not Enter” signs) are added to the training dataset.

5 Results and analysis

5.1 Performance of the trained model

In this section, the performance of the trained YOLOv4 model is evaluated on the test dataset, and results are discussed. For the test set, we extracted 176 pre-flood and 172 post-flood photos from the Blupix v.2021.1 dataset by filtering out photos with significantly low resolution or those in which parts of poles and/or stop signs were not visible. Of these pre- and post-flood photos, 163 photos are paired. Table 3 summarizes model performance with the optimum trained weights when tested on the test set. As shown in this Table, the AP calculated for stop sign and pole detection in pre-flood photos is 100% and 99.41%, respectively. Similarly, the AP calculated for stop sign and pole detection in post-flood photos is 99.73%, and 98.73%, respectively. The mAP for pre- and post-flood photos is 99.70% and 99.23%, respectively. The relatively higher mAP for pre-flood photos can be attributed to less noise in these photos compared to post-flood photos. The average detection times for all detections

Table 3 Performance of the trained model in detecting stop signs and poles in pre- and post-flood photos of the test set (S: Stop sign; P: sign pole; S + P: stop sign and sign pole)

Metric	Class	Pre-flood (n = 176)	Post-flood (n = 172)	All photos (n = 348)
Average IoU	S + P	93.85%	86.25%	90.05%
Precision @ 0.25 conf	S + P	0.99	0.94	0.96
Recall @ 0.25 conf	S + P	0.99	0.95	0.97
AP	S	100.00%	99.73%	99.84%
	P	99.41%	98.73%	99.16%
mAP @ 0.50	S + P	99.70%	99.23%	99.50%
F1-score	S + P	0.99	0.94	0.97
Average detection time (sec)	S + P	0.05	0.07	0.05

in pre-flood and post-flood photos are 0.05 and 0.07 s, respectively, which is close to real-time.

Without considering the uneven degrees of tilt for few paired photos, the model can calculate pole lengths in test images with an MAE of 1.856 in. and 2.882 in. for pre- and post-flood photos, respectively. The slightly higher error corresponding to post-flood photos can be primarily attributed to the presence of visual noise in post-flood scenes. To examine the tilt correction method, 37 images of stop signs in the test set with uneven pole tilt degrees in pre- and post-flood photos are identified. After implementing the tilt correction technique, the MAE of the trained YOLOv4 model on the Blupix v.2021.1 dataset is reduced to 1.723 in. and 2.846 in. for pre- and post-flood photos, respectively, showing a slight improvement in flood depth estimation outcome as a result of implementing the tilt correction technique. However, it is anticipated that with more paired photos depicting uneven degrees of tilt, the reduction in error becomes more significant. Table 4 summarizes model performance on the test set. To calculate the error of flood depth estimation, ground-truth floodwater depth (i.e., the difference between ground-truth pole lengths in paired pre- and post-flood photos) is compared with the estimated floodwater depth (i.e., the difference between detected pole lengths in paired pre- and post-flood photos). The MAE of the model for flood depth estimation on 163 paired photos were achieved as 4.737 in. and 4.710 in. before and after implementing the tilt correction technique, respectively.

5.2 Pole length estimation using baseline methods

In addition to assessing the performance of the trained YOLOv4 model using metrics such as MAE and RMSE, we compare model performance with two baseline

Table 4 Performance of the trained model in estimating pole lengths in pre- and post-flood photos of the test before and after tilt correction

Source	Metric	Pre-flood (<i>n</i> = 176)	Post-flood (<i>n</i> = 172)	Paired photos (<i>n</i> = 163)
Before tilt correction	RMSE (in.)	3.924	5.557	7.847
	MAE (in.)	1.856	2.882	4.737
	Average processing time (s)	0.05	0.08	0.08
After tilt correction	RMSE (in.)	3.355	5.544	7.857
	MAE (in.)	1.723	2.846	4.710
	Average processing time (s)	0.05	0.08	0.08

approaches (a.k.a., dummy methods). The purpose of these dummy methods is to verify that the performance of the YOLOv4 model exceeds that of a simple model that returns average values given a set of pole length values. In dummy method I, for a given pre-flood (post-flood) test image, the model returns the average pole length of all pre-flood (post-flood) images in the training set. The average pole length for pre-flood and post-flood photos in the training set is 76.98 in. (*n* = 334) and 53.38 in. (*n* = 395), respectively. Using dummy method I, the MAE for the pre-flood photos and post-flood photos in the test set is thus determined as 44.628 in. (*n* = 176) and 49.804 in. (*n* = 172). In dummy method II, for a given pre-flood (post-flood) test image, the model returns the running average pole length of all previously seen pre-flood (post-flood) images in the test set. Running average is a common method for extracting an overall trend from a list of values, by continuously updating an average value considering all data points in the set until the calculation point (Cragger & Reitman, 1991; Du et al., 2008; Pierce, 1971; Tan et al., 2021). To reduce the order effect and allowing for a thorough examination of the variability and accuracy of the model across a range of randomized data sets, pole length values are recorded in 100 iterations each containing a randomized order of test images. Consequently, the performance of dummy method II is calculated as the MAE of all 100 obtained running averages. The MAE achieved for dummy method II for pre-flood and post-flood photos is 13.271 in. and 20.469 in., respectively. Also, the minimum and maximum MAE achieved for pre-flood (post-flood) is found to be 0.056 in. and 42.830 in. (0.344 and 45.824), respectively. Our results show that dummy method II is able to reduce the pole length estimation error to less than 1 in. in only one randomized set. However, dummy method II outperforms dummy method I, yet is highly sensitive to the order of values. Comparing the MAE of the proposed YOLOv4 model, (i.e., 1.723 in. for pre-flood photos and 2.846 in. for post-flood photos) with the MAEs of dummy

Table 5 Performance of dummy methods I and II on the test set

Baseline method	Category (test set)	MAE (in.)	RMSE (in.)
Dummy Method I	Pre-flood (<i>n</i> = 176)	44.628	51.884
	Post-flood (<i>n</i> = 172)	49.804	60.165
Dummy Method II	Pre-flood (<i>n</i> = 176)	13.271*	* MAE reported over 100 randomized iterations
	Post-flood (<i>n</i> = 172)	20.469*	

Table 6 The performance of the flood depth estimation model based on stop sign language

Category	Language	MAE (in.)	RMSE (in.)
Pre-flood	English (168)	1.585	3.078
	French (8)	4.628	6.977
	Total (176)	1.723	3.355
Post-flood	English (164)	2.908	5.662
	French (8)	1.565	1.876
	Total (172)	2.846	5.544

methods I and II, it is clear that our proposed model outperforms the two baseline methods. Table 5 summarizes the MAE and RSME obtained using dummy methods I and II.

5.3 Impact of stop sign language on model performance

To analyze the performance of the model in different languages, MAE and RMSE values are calculated for various subsets of photos (after implementing tilt correction), with results summarized in Table 6. The analysis indicates that in pre-flood photos, the MAE for stop signs in French is 4.628 in., which is higher than the MAE of 1.585 in. for stop signs in English. On the other hand, in post-flood photos, the MAE for stop signs in French is 1.565 in., which is lower than the MAE of 2.908 in. for stop signs in English. Further investigation reveals that

the MAE for pole length estimation is impacted by image quality rather than stop sign language. For example, web-mined post-flood photos of stop signs in French language are in high-resolution (taken by professional cameras), thus lowering the corresponding MAE. In contrast, the quality of one of the French pre-flood photos was significantly low which led to a higher MAE for pole length estimation in the corresponding subset.

5.4 Benchmarking

As stated earlier, the model achieved an MAE of 4.710 in. on 163 paired images in the test set after implementing tilt correction. By comparison, Cohen et al. (2019) obtained an average absolute difference of 18–31 cm (approximately 7–12 in.) for flood depth estimation in coastal and riverine areas, Chaudhary et al. (2019) reported a mean absolute error of 10 cm (approximately 4 in.) in estimating flood depth based on comparing submerged objects in social media images with their predefined sizes, and Park et al. (2021) presented a mean absolute error value of 6.49 cm (approximately 2.5 in.) by comparing visible parts of submerged vehicles with their estimated size. As summarized in Table 7, a comparison of the flood depth estimation error obtained in this research to previous studies indicates the reliability and generalizability of the developed technique in measuring floodwater depth with acceptable accuracy.

6 Summary and conclusion

Flooding is one of the most prevalent natural hazards that results in significant loss of life, and disrupts properties and infrastructure. Due to the constant change in water levels on the road network during a flood, reliable and real-time flood depth information at the street level is critical for decision-making in evacuation and rescue

operations. Current methods of obtaining flood depth (including water gauges, DEMs, hydrological models, and SAR) often suffer from shortage of data, inherent uncertainties, high installation and maintenance costs, and the need for heavy computing power. Recent advancements in computer vision and AI have created new opportunities for remotely estimating the flood depth based on comparing submerged objects with their predefined sizes. In this paper, a deep learning approach, based on the YOLOv4 architecture, was proposed for estimating floodwater depth in crowdsourced street photos using traffic signs. Since traffic signs have standardized sizes, the difference between pole lengths in paired pre- and post-flood photos of the same sign can be computed and used as the basis for estimating the depth of floodwater at the location of the sign. An in-house training set comprising web-mined photos and photos extracted from the Microsoft COCO dataset, was used for training the YOLOv4 model. The trained model was then validated using fivefold cross validation, and subsequently tested for flood depth estimation on 163 paired photos from our in-house test set. Results indicate an MAE of 1.723 and 2.846 in. for pole length estimation in pre- and post-flood photos, respectively, and an MAE of 4.710 in. for floodwater depth estimation. Also, the performance of the proposed model surpassed that of two baseline approaches (dummy method I, which returns the average pole length of all images in the training set for each image in the test set; and dummy method II, which calculates the running average of pole lengths in all previously seen images in the test set). In addition, a tilt correction method was developed to minimize the pole length estimation error in paired photos of stop signs with uneven degrees of tilt. As a part of the future direction of this research, the authors aim to increase the generalizability

Table 7 Comparison of the results of this study with the literature on floodwater depth estimation

Study	MAE	Method	Practical Considerations
Cohen et al. (2019)	18–31 cm (7–12 in.)	Augmenting DEM data for Coastal and riverine areas	Challenges in obtaining high-spatial-resolution DEMs, high uncertainty in highly fragmented flooded areas
Chaudhary et al. (2019)	10 cm (4 in.)	Submerged objects in social media images with their predefined sizes	Assuming predefined sizes for different objects
Park et al. (2021)	6.49 cm (2.5 in.)	Flooded vehicle images in ground-level view	Finding submerged vehicles, assuming predefined sizes for different vehicles
Alizadeh Kharazi and Behzadan (2021)	12.65 in. (32.13 cm)	Paired stop signs in crowdsourced photos, using pre-trained Mask R-CNN and line detection models. Tested on Blupixv.2020.1 dataset	Computationally not efficient, challenges in pairing photos
This paper	4.710 in. (11.96 cm) after tilt correction [4.737 in. (12.03 cm) before tilt correction]	Paired stop signs in crowdsourced photos, using a trained YOLOv4 model on an in-house dataset. Tested on Blupix v.2021.1 dataset	Relies on citizen science as a means for data collection

of the floodwater depth estimation model by training it on various forms of traffic signs and other standardized urban landmarks. Moreover, to evaluate the real-world performance and practicality of the proposed methodology, the authors are conducting a user study of people's perception of risk during flood events and the value of information provided by the proposed flood depth estimation method on their decisions.

Abbreviations

AI	Artificial intelligence
AP	Average precision
CNN	Convolutional neural network
CSP	Cross-stage-partial-connections
DEM	Digital elevation model
FEMA	Federal Emergency Management Agency
FN	False negative
FP	False positive
FPN	Feature pyramid network
FPS	Frames per second
IoU	Intersection over union
LiDAR	Light detection and ranging
MAE	Mean absolute error
mAP	Mean average precision
NEXRAD	Next Generation Weather Radar
R-CNN	Region-based CNN
RSME	Root mean squared error
SAR	Synthetic aperture radar
SPP	Spatial pyramid pooling
SSD	Single shot detector
TP	True positive
YOLOv3	You-only-look-once version 3

Acknowledgements

Authors also like to acknowledge the support of Mr. Greg Spiller and Mr. Chris Mouchyn (College of Geosciences IT professionals at Texas A&M University) for hosting the Blupix application on the web (<https://blupix.geos.tamu.edu>), and Diya Li (Ph.D. Student of Geography), for overseeing the application design and content maintenance. In addition, authors would like to thank Mr. Nathan Young (undergraduate student at Texas A&M University) for his assistance in data collection. Any opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily represent the views of the NOAA, Department of Commerce, or the individuals named above.

Code availability

Not applicable.

Author's contributions

Bahareh Alizadeh co-designed the framework, collected data, prepared ground truth information, implemented programing, and performed data analysis. Amir H. Behzadan developed the project concept, co-designed the framework, and addressed research methodology and evaluation metrics. Both authors have read and approved the final manuscript.

Funding

This study is funded by award #NA18OAR4170088 from the National Oceanic and Atmospheric Administration (NOAA), U.S. Department of Commerce. National Oceanic and Atmospheric Administration, NA18OAR4170088

Availability of data and materials

Not applicable.

Declarations

Competing interests

Any opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily represent the views of the NOAA, Department of Commerce, or the individuals named above.

Received: 20 February 2023 Revised: 7 March 2023 Accepted: 12 March 2023

Published online: 06 April 2023

References

- Alfieri, L., Bisselink, B., Dottori, F., Naumann, G., de Roo, A., Salamon, P., ... & Feyen, L. (2017). Global projections of river flood risk in a warmer world. *Earth's Future*, 5(2), 171–182. <https://doi.org/10.1002/2016EF000485>
- Alinezhad, A., Gohari, A., Eslamian, S., Baghbani, R. (2020). Uncertainty Analysis in Climate Change Projection Using Bayesian Approach. In World Environmental and Water Resources Congress. (2020). *Groundwater, Sustainability, Hydro-Climatology/Climate Change, and Environmental Engineering*, 167–174. Reston, VA: American Society of Civil Engineers. <https://doi.org/10.1061/9780784482964.017>
- Alizadeh, B., & Behzadan, A. H. (2022a). Blupix: citizen science for flood depth estimation in urban roads. In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on Advances in Resilient and Intelligent Cities*, 16–19. <https://doi.org/10.1145/3557916.3567824>
- Alizadeh, B., & Behzadan, A. H. (2022b). Crowdsourced-based deep convolutional networks for urban flood depth mapping. *arXiv preprint arXiv:2209.09200*. <https://arxiv.org/abs/2209.09200v1>
- Alizadeh B., Li D., Zhang Z., Behzadan A.H. (2021). Feasibility Study of Urban Flood Mapping Using Traffic Signs for Route Optimization. *28th EG-ICE International Workshop on Intelligent Computing in Engineering*. Berlin, Germany, 572–581. <https://arxiv.org/abs/2109.11712>.
- Alizadeh, B., Li, D., Hillin, J., Meyer, M. A., Thompson, C. M., Zhang, Z., & Behzadan, A. H. (2022). Human-centered flood mapping and intelligent routing through augmenting flood gauge data with crowdsourced street photos. *Advanced Engineering Informatics*, 54, 101730. <https://doi.org/10.1016/j.aei.2022.101730>
- Alizadeh Kharazi, B., & Behzadan, A. H. (2021). Flood depth mapping in street photos with image processing and deep neural networks. *Computers, Environment and Urban Systems*, 88, 101628. <https://doi.org/10.1016/j.compenvurbsys.2021.101628>
- Arabi, M., Hyun, K., & Mattingly, S. P. (2021). *Adaptable Resilience Assessment Framework to Evaluate an Impact of a Disruptive Event on Freight Operations No. TRBAM-21-03974*. <https://doi.org/10.1177/03611981211033864>
- Arnell, N. W., & Gosling, S. N. (2016). The impacts of climate change on river flood risk at the global scale. *Climatic Change*, 134(3), 387–401. <https://doi.org/10.1007/s10584-014-1084-5>
- Asner, G. P. (2001). Cloud cover in Landsat observations in the Brazilian Amazon. *International Journal of Remote Sensing*, 22, 3855–3862. <https://doi.org/10.1080/01431160010006926>
- Beck, J. L., & Katafygiotis, L. S. (1998). Updating models and their uncertainties. I: Bayesian statistical framework. *Journal of Engineering Mechanics*, 124(4), 455–461. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1998\)124:4\(455\)](https://doi.org/10.1061/(ASCE)0733-9399(1998)124:4(455))
- Bjorvatn, K. (2000). Urban infrastructure and industrialization. *J. Urban Economics*, 48(2), 205–218. <https://doi.org/10.1006/juec.1999.2162>
- Blupix (2020). Blupix application. Available at <https://blupix.geos.tamu.edu/>
- Bochkovskiy, A. Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint*. <https://arxiv.org/abs/2004.10934>.
- Bochkovskiy, A. (2020). Darknet: Open Source Neural Networks in Python. <https://github.com/AlexeyAB/darknet>. (Accessed June 2021).
- Bowes, B. D., Tavakoli, A., Wang, C., Heydarian, A., Behl, M., Beling, P. A., & Goodall, J. L. (2021). Flood mitigation in coastal urban catchments using real-time stormwater infrastructure control and reinforcement learning.

- Journal of Hydroinformatics*, 23(3), 529–547. <https://doi.org/10.2166/hydro.2020.080>
- Bradshaw, C. J., Sodhi, N. S., & PEH, K. S. H., & Brook, B. W. (2007). Global evidence that deforestation amplifies flood risk and severity in the developing world. *Global Change Biology*, 13(11), 2379–2395. <https://doi.org/10.1111/j.1365-2486.2007.01446.x>
- Brown, K. M., Hambidge, C. H., & Brownnett, J. M. (2016). Progress in operational flood mapping using satellite synthetic aperture radar (SAR) and airborne light detection and ranging (LiDAR) data. *Progress in Physical Geography*, 40(2), 196–214. <https://doi.org/10.1177/0309133316633570>
- Browne, M. W. (2000). *Cross-Validation Methods*. *Journal of Mathematical Psychology*, 44(1), 108–132. <https://doi.org/10.1006/jmps.1999.1279>
- Centre for Research on the Epidemiology of Disasters (CREED) (2022). 2021 Disasters in numbers. Institute Health and Society – UCLouvain, Brussels, Belgium. https://cred.be/sites/default/files/2021_EMDAT_report.pdf.
- Changnon, S. A. (2000). A defining event for flood mitigation policy in the United States. *floods*, 1, 288.
- 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 Journal of Photogrammetry and Remote Sensing*, 167, 252–262. <https://doi.org/10.1016/j.isprsjprs.2020.07.003>
- Chaudhary, P., D'Aronco, S., Moy de Vitry, M., Leitão, J. P., & Wegner, J. D. (2019). Flood-water level estimation from social media images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4(2/W5), 5–12. <https://doi.org/10.5194/isprs-annals-IV-2-W5-5-2019>
- Chew, C., Reager, J. T., & Small, E. (2018). CYGNSS data map flood inundation during the 2017 Atlantic hurricane season. *Scientific Reports*, 8(1), 1–8. <https://doi.org/10.1038/s41598-018-27673-x>
- Cohen, S., Raney, A., Munasinghe, D., Loftis, J. D., Molthan, A., Bell, J., ... & Tsang, Y. P. (2019). The Floodwater Depth Estimation Tool (FwDET v2. 0) for improved remote sensing analysis of coastal flooding. *Natural Hazards and Earth System Sciences*, 19(9), 2053–2065. <https://doi.org/10.5194/nhess-19-2053-2019>.
- Crabit, A., Colin, F., Bailly, J. S., Ayroles, H., & Garnier, F. (2011). Soft water level sensors for characterizing the hydrological behaviour of agricultural catchments. *Sensors*, 11(5), 4656–4673. <https://doi.org/10.3390/s110504656>
- Crager, M. R., & Reitman, M. A. (1991). Running average analysis of clinical trial ambulatory blood pressure data. *Biometrics*, 129–137. <https://www.jstor.org/stable/2532501>
- Cutter, S. L., Boruff, B. J., & Shirley, W. L. (2003). Social vulnerability to environmental hazards. *Social Science Quarterly*, 84(2), 242–261. <https://doi.org/10.1111/1540-6237.8402002>
- Davis, S. A., & Skaggs, L. L. (1992). Catalog of residential depth-damage functions used by the army corps of engineers in flood damage estimation. *ARMY ENGINEER INST FOR WATER RESOURCES ALEXANDRIA VA*. <https://apps.dtic.mil/sti/citations/ADA255462>.
- Dong, S., Yu, T., Farahmand, H., & Mostafavi, A. (2021). A hybrid deep learning model for predictive flood warning and situation awareness using channel network sensors data. *Computer-Aided Civil and Infrastructure Engineering*, 36(4), 402–420. <https://doi.org/10.1111/mice.12629>
- Du, Z. L., Wang, H. N., & Zhang, L. Y. (2008). A running average method for predicting the size and length of a solar cycle. *Chinese Journal of Astronomy and Astrophysics*, 8(4), 477. <https://doi.org/10.1088/1009-9271/8/4/12>
- Eby, M., & Ensor, C. (2019). Understanding FEMA flood maps and limitations. *First Street Foundation* (March 21). <https://firststreet.org/research-lab/published-research/understanding-fema-flood-maps-and-limitations/>.
- Farhadmanesh, M., Cross, C., Mashhadi, A. H., Rashidi, A., & Wempem, J. (2021). *Use of Mobile Photogrammetry Method for Highway Asset Management (No. TRBAM-21-01864)*. <https://doi.org/10.1177/03611981211001855>
- Federal Highway Administration (2004). *Manual on Uniform Traffic Control Devices (MUTCD): Standard highway signs*. https://mutcd.fhwa.dot.gov/ser-shs_millennium_eng.htm. (Accessed 08/03/2021)
- Feingersh, T., Ben-Dor, E., & Portugali, J. (2007). Construction of synthetic spectral reflectance of remotely sensed imagery for planning purposes. *Environmental Modelling & Software*, 22(3), 335–348. <https://doi.org/10.1016/j.envsoft.2005.11.005>
- Feng, Q., Liu, J., & Gong, J. (2015). Urban flood mapping based on unmanned aerial vehicle remote sensing and random forest classifier—A case of Yuyao. *China. Water*, 7(4), 1437–1455. <https://doi.org/10.3390/w7041437>
- Felzenszwalb, P. F., Girshick, R. B., & McAllester, D. (2010). Cascade object detection with deformable part models. In 2010 IEEE Computer society conference on computer vision and pattern recognition. IEEE. 2241–2248. <https://doi.org/10.1109/CVPR.2010.5539906>
- Ferro, A., Brunner, D., Bruzzone, L., & Lemoine, G. (2011). On the relationship between double bounce and the orientation of buildings in VHR SAR images. *IEEE Geoscience and Remote Sensing Letters*, 8(4), 612–616. <https://doi.org/10.1109/LGRS.2010.2097580>
- Figueiredo, R., Schröter, K., Weiss-Motz, A., Martina, M. L., & Kreibich, H. (2018). Multi-model ensembles for assessment of flood losses and associated uncertainty. *Natural Hazards and Earth System Sciences*, 18(5), 1297–1314. <https://doi.org/10.5194/nhess-18-1297-2018>
- Forati, A. M., & Ghose, R. (2021). Examining Community Vulnerabilities through multi-scale geospatial analysis of social media activity during Hurricane Irma. *International Journal of Disaster Risk Reduction*, 102701. <https://doi.org/10.1016/j.ijdrr.2021.102701>
- Forzieri, G., Cescatti, A., e Silva, F. B., & Feyen, L. (2017). Increasing risk over time of weather-related hazards to the European population: A data-driven prognostic study. *The Lancet Planetary Health*, 1(5), e200–e208. [https://doi.org/10.1016/S2542-5196\(17\)30082-7](https://doi.org/10.1016/S2542-5196(17)30082-7)
- Frappart, F., Seyler, F., Martinez, J. M., León, J. G., & Cazenave, A. (2005). Flood-plain water storage in the Negro River basin estimated from microwave remote sensing of inundation area and water levels. *Remote Sensing of Environment*, 99(4), 387–399. <https://doi.org/10.1016/j.rse.2005.08.016>
- Fu, C. Y., Liu, W., Ranga, A., Tyagi, A., & Berg, A. C. (2017). Dssd: Deconvolutional single shot detector. *arXiv preprint*. <https://arxiv.org/abs/1701.06659>.
- Gandhi, R. (2018). R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms. *Towards data science*. <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e>. (Accessed 08/058/2021)
- Gao, Y., & Mosalam, K. M. (2018). Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, 33(9), 748–768. <https://doi.org/10.1111/mice.12363>
- Gerl, T., Kreibich, H., Franco, G., Marechal, D., & Schröter, K. (2016). A review of flood loss models as basis for harmonization and benchmarking. *PLoS one*, 11(7), e0159791. <https://doi.org/10.1371/journal.pone.0159791>
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587. <https://doi.org/10.1109/CVPR.2014.81>.
- Girshick, R. J. C. S. (2015). Fast r-cnn. *arXiv preprint arXiv:1504.08083*.
- Golparvar, B., & Wang, R. Q. (2021). AI-supported Framework of Semi-Automatic Monoplotting for Monocular Oblique Visual Data Analysis. *arXiv preprint*. <https://arxiv.org/abs/2111.14021>
- Guo, H., Shi, Q., Marinoni, A., Du, B., & Zhang, L. (2021). Deep building footprint update network: A semi-supervised method for updating existing building footprint from bi-temporal remote sensing images. *Remote Sensing of Environment*, 264, 112589. <https://doi.org/10.1016/j.rse.2021.112589>
- Han, D., Liu, Q., & Fan, W. (2018). A new image classification method using CNN transfer learning and web data augmentation. *Expert Systems with Applications*, 95, 43–56. <https://doi.org/10.1016/j.eswa.2017.11.028>
- Hao, W., & Zhili, S. (2020). Improved Mosaic: Algorithms for more Complex Images. In *Journal of Physics: Conference Series*. IOP Publishing. 1684(1), 012094. <https://doi.org/10.1088/1742-6596/1684/1/012094>.
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9), 1904–1916. <https://doi.org/10.1109/TPAMI.2015.2389824>
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 2961–2969. <https://doi.org/10.48550/arXiv.1703.06870>
- Henderson, P., & Ferrari, V. (2017). End-to-end training of object class detectors for mean average precision. In *Computer Vision—ACCV 2016: 13th Asian Conference on Computer Vision*, Taipei, Taiwan, November 20–24, 2016, Revised Selected Papers, Part V 13, 198–213. Springer International Publishing. https://doi.org/10.1007/978-3-319-54193-8_13

- Hicks, M., & Burton, M. (2017). Hurricane Harvey: Preliminary estimates of commercial and public sector damages on the Houston metropolitan area. Ball State University. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.318.7580&rep=rep1&type=pdf>.
- Hosseiny, H. (2021). A deep learning model for predicting river flood depth and extent. *Environmental Modelling & Software*, 145, 105186. <https://doi.org/10.1016/j.envsoft.2021.105186>
- Hsu, K. L., Moradkhani, H., & Sorooshian, S. (2009). A sequential Bayesian approach for hydrologic model selection and prediction. *Water Resources Research*, 45(12). <https://doi.org/10.1029/2008WR006824>
- Hu, R., Zhang, S., Wang, P., Xu, G., Wang, D., & Qian, Y. (2020). The identification of corn leaf diseases based on transfer learning and data augmentation. In *Proceedings of the 2020 3rd International Conference on Computer Science and Software Engineering*, 58–65. <https://doi.org/10.1145/3403746.3403905>.
- Hu, T. Y., Armandpour, M., Shrivastava, A., Chang, J. H. R., Koppala, H., & Tuzel, O. (2021). Synt++: Utilizing Imperfect Synthetic Data to Improve Speech Recognition. *arXiv preprint arXiv:2110.11479*. <https://arxiv.org/abs/2110.11479>.
- Huang, S., Krysanova, V., & Hattermann, F. (2015). Projections of climate change impacts on floods and droughts in Germany using an ensemble of climate change scenarios. *Regional Environmental Change*, 15, 461–473. <https://doi.org/10.1007/s10113-014-0606-z>
- Hussain, M., Bird, J. J., & Faria, D. R. (2018). A study on cnn transfer learning for image classification. In *UK Workshop on computational Intelligence*, Springer, Cham, 191–202. https://doi.org/10.1007/978-3-319-97982-3_16.
- Islam, R., Lee, Y., Jaloli, M., Muhammad, I., Zhu, D., Rad, P., ... & Quarles, J. (2020). Automatic detection and prediction of cybersickness severity using deep neural networks from user's physiological signals. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* 400–411. <https://doi.org/10.1109/ISMAR50242.2020.00066>
- Javadi, S., Maghami, A., & Hosseini, S. M. (2021). A deep learning approach based on a data-driven tool for classification and prediction of thermoelastic wave's band structures for phononic crystals. *Mechanics of Advanced Materials and Structures*, 1–14. <https://doi.org/10.1080/15376494.2021.1983088>
- Jiang, J., Liu, J., Cheng, C., Huang, J., & Xue, A. (2019). Automatic estimation of urban waterlogging depths from video images based on ubiquitous reference objects. *Remote Sensing*, 11(5), 587. <https://doi.org/10.3390/rs11050587>
- Ju, M., Luo, H., Wang, Z., Hui, B., & Chang, Z. (2019). The application of improved YOLO V3 in multi-scale target detection. *Applied Sciences*, 9(18), 3775. <https://doi.org/10.3390/app9183775>
- Kamari, M., & Ham, Y. (2021). Vision-based volumetric measurements via deep learning-based point cloud segmentation for material management in jobsites. *Automation in Construction*, 121, 103430. <https://doi.org/10.1016/j.autcon.2020.103430>.
- L. -W. Kang, I. -S. Wang, K. -L. Chou, S. -Y. Chen and C. -Y. Chang (2019). Image-Based Real-Time Fire Detection using Deep Learning with Data Augmentation for Vision-Based Surveillance Applications, 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2019, 1–4, <https://doi.org/10.1109/AVSS.2019.8909899>.
- Kido, D., Fukuda, T., & Yabuki, N. (2020). Diminished reality system with real-time object detection using deep learning for onsite landscape simulation during redevelopment. *Environmental Modelling & Software*, 131, 104759. <https://doi.org/10.1016/j.envsoft.2020.104759>
- Kim, J., Han, Y., & Hahn, H. (2011). Embedded implementation of image-based water-level measurement system. *IET Computer Vision*, 5(2), 125–133. <https://doi.org/10.1049/iet-cvi.2009.0144>
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. <https://arxiv.org/abs/1412.6980>
- Klemas, V. (2015). Remote sensing of floods and flood-prone areas: An overview. *Journal of Coastal Research*, 31(4), 1005–1013. <https://doi.org/10.2112/JCOASTRES-D-14-00160.1>
- Koirala, A., Walsh, K. B., Wang, Z., & McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of 'MangoYOLO'. *Precision Agriculture*, 20(6), 1107–1135. <https://doi.org/10.1007/s11119-019-09642-0>
- Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*. Springer, Cham, 740–755. https://doi.org/10.1007/978-3-319-10602-1_48.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017a). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125. <https://arxiv.org/abs/1612.03144>.
- Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017b). Focal loss for dense object detection. In *Proceedings of The IEEE International Conference on Computer Vision*, 2980–2988. <https://arxiv.org/abs/1708.02002v2>.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European Conference on Computer Vision*. Springer, Cham, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2.
- Liu, S., Qi, L., Qin, H., Shi, J., Jia, J. (2018). Path Aggregation Network for Instance Segmentation. *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit.*, 8759–68. <https://arxiv.org/abs/1803.01534v4>.
- Liu, L., Liu, Y., Wang, X., Yu, D., Liu, K., Huang, H., & Hu, G. (2015). Developing an effective 2-D urban flood inundation model for city emergency management based on cellular automata. *Natural hazards and earth system sciences*, 15(3), 381–391. <https://doi.org/10.5194/nhess-15-381-2015>
- 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>
- Lyons, M. B., Keith, D. A., Phinn, S. R., Mason, T. J., & Elith, J. (2018). A comparison of resampling methods for remote sensing classification and accuracy assessment. *Remote Sensing of Environment*, 208, 145–153. <https://doi.org/10.1016/j.rse.2018.02.026>
- Lyu, Y., Bai, L., & Huang, X. (2019). Road segmentation using cnn and distributed lstm. In *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, 1–5. <https://doi.org/10.1109/ISCAS.2019.8702174>.
- Ma, J., Duan, S., Zhang, Y., Wang, J., Wang, Z., Li, R., ... & Ma, H. (2020). Efficient deep learning architecture for detection and recognition of thyroid nodules. *Computational Intelligence and Neuroscience*, 2020. <https://doi.org/10.1155/2020/1242781>.
- Mao, X., Chow, J. K., Su, Z., Wang, Y. H., Li, J., Wu, T., & Li, T. (2021). Deep learning-enhanced extraction of drainage networks from digital elevation models. *Environmental Modelling & Software*, 144, 105135. <https://doi.org/10.1016/j.envsoft.2021.105135>
- Marco, C., Claudio, B., Ueli, R., Thalia, B., & Patrik, K. (2018). Using the Monoplotting technique for documenting and analyzing natural hazard events. In *Natural Hazards-Risk Assessment and Vulnerability Reduction*. IntechOpen.
- Merwade, V., Olivera, F., Arabi, M., & Edleman, S. (2008). Uncertainty in flood inundation mapping: Current issues and future directions. *Journal of Hydrologic Engineering*, 13(7), 608–620. [https://doi.org/10.1061/\(ASCE\)1084-0699\(2008\)13:7\(608\)](https://doi.org/10.1061/(ASCE)1084-0699(2008)13:7(608))
- 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. *Hydrology and Earth System Sciences*, 23(11), 4621–4634. <https://doi.org/10.5194/hess-23-4621-2019>
- Nath, N. D., & Behzadan, A. H. (2020). Deep convolutional networks for construction object detection under different visual conditions. *Frontiers in Built Environment*, 6, 97. <https://doi.org/10.3389/fbuil.2020.00097>
- National Oceanic and Atmospheric Administration (NOAA) (2022). Radar Operation Center NEXRAD WSR-88D, About the ROC. <https://www.roc.noaa.gov/WSR88D/About.aspx>
- Nazari, F., & Yan, W. (2021). Convolutional versus Dense Neural Networks: Comparing the Two Neural Networks Performance in Predicting Building Operational Energy Use Based on the Building Shape. *arXiv preprint arXiv:2108.12929*. <https://arxiv.org/abs/2108.12929>
- Niu, J., Chen, Y., Yu, X., Li, Z., & Gao, H. (2020). Data Augmentation on Defect Detection of Sanitary Ceramics. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, 5317–5322. IEEE. <https://doi.org/10.1109/IECON43393.2020.9254518>.
- Padilla, R., Netto, S. L., & da Silva, E. A. (2020). A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP) IEEE*. 237–242. <https://doi.org/10.1109/IWSSIP48289.2020.9145130>.

- Pally, R. J., & Samadi, S. (2021). Application of image processing and convolutional neural networks for flood image classification and semantic segmentation. *Environmental Modelling & Software*, 105285. <https://doi.org/10.1016/j.envsoft.2021.105285>
- Park, S., Baek, F., Sohn, J., & Kim, H. (2021). Computer vision-based estimation of flood depth in flooded-vehicle images. *Journal of Computing in Civil Engineering*, 35(2), 04020072. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000956](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000956)
- Perks, M. T., Russell, A. J., & Large, A. R. (2016). Advances in flash flood monitoring using unmanned aerial vehicles (UAVs). *Hydrology and Earth System Sciences*, 20(10), 4005–4015. <https://doi.org/10.5194/hess-20-4005-2016>
- Pierce, D. A. (1971). Least squares estimation in the regression model with autoregressive-moving average errors. *Biometrika*, 58(2), 299–312. <https://doi.org/10.1093/biomet/58.2.299>
- Pulvirenti, L., Chini, M., Pierdicca, N., Guerriero, L., & Ferrazzoli, P. (2011). Flood monitoring using multi-temporal COSMO-SkyMed data: Image segmentation and signature interpretation. *Remote Sensing of Environment*, 115(4), 990–1002. <https://doi.org/10.1016/j.rse.2010.12.002>
- Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint*. <https://arxiv.org/abs/1804.02767>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards realtime object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 1137–1149. <https://doi.org/10.1109/tpami.2016.2577031>
- Robertson, S. (2008). A new interpretation of average precision. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 689–690. <https://doi.org/10.1145/1390334.1390453>
- Rodríguez Castro, D., Roucour, S., Archambeau, P., Cools, M., Epicum, S., Habchi, I., ... & Dewals, B. (2022). Modelling direct flood losses: what can we learn from the July 2021 flood in the Meuse basin (Belgium)? In *KAHR Science Conference*. <https://hdl.handle.net/2268/293640>
- Romali, N. S., & Yusop, Z. (2021). Flood damage and risk assessment for urban area in Malaysia. *Hydrology Research*, 52(1), 142–159. <https://doi.org/10.2166/nh.2020.121>
- Sahin, V., & Hall, M. J. (1996). The effects of afforestation and deforestation on water yields. *J. Hydrology*, 178(1–4), 293–309. [https://doi.org/10.1016/0022-1694\(95\)02825-0](https://doi.org/10.1016/0022-1694(95)02825-0)
- Sakaino, H. (2016). Camera-vision-based water level estimation. *IEEE Sensors Journal*, 16(21), 7564–7565. <https://doi.org/10.1109/JSEN.2016.2603524>
- Schumann, G. J. P., Neal, J. C., Mason, D. C., & Bates, P. D. (2011). The accuracy of sequential aerial photography and SAR data for observing urban flood dynamics, a case study of the UK summer 2007 floods. *Remote Sensing of Environment*, 115(10), 2536–2546. <https://doi.org/10.1016/j.rse.2011.04.039>
- See, L. (2019). A review of citizen science and crowdsourcing in applications of pluvial flooding. *Frontiers in Earth Science*, 7, 44. <https://doi.org/10.3389/feart.2019.00044>
- Seyrfar, A., Ataei, H., Movahedi, A., & Derrible, S. (2021). Data-Driven Approach for Evaluating the Energy Efficiency in Multifamily Residential Buildings. *Practice Periodical on Structural Design and Construction*, 26(2), 04020074. [https://doi.org/10.1061/\(ASCE\)SC.1943-5576.0000555](https://doi.org/10.1061/(ASCE)SC.1943-5576.0000555)
- Shaghaghian, Z., & Yan, W. (2019). Application of Deep Learning in Generating Desired Design Options: Experiments Using Synthetic Training Dataset. *arXiv preprint*. <https://arxiv.org/abs/2001.05849>
- Shen, X., Anagnostou, E. N., Allen, G. H., Brakenridge, G. R., & Kettner, A. J. (2019). Near-real-time non-obstructed flood inundation mapping using synthetic aperture radar. *Remote Sensing of Environment*, 221, 302–315. <https://doi.org/10.1016/j.rse.2018.11.008>
- Singh, R. B., & Singh, S. (2011). Rapid urbanization and induced flood risk in Noida. *India. Asian Geographer*, 28(2), 147–169. <https://doi.org/10.1080/10225706.2011.629417>
- Smith, A. (2021). (2020) U.S. billion-dollar weather and climate disasters in historical context. *Beyond the Data. Climate news, stories, images, & video*. <https://www.climate.gov/news-features/blogs/beyond-data/2020-us-billion-dollar-weather-and-climate-disasters-historical>. (Accessed 08/08/2021)
- Sokolova, G. V., Verkhotourov, A. L., & Korolev, S. P. (2019). Impact of deforestation on streamflow in the Amur River Basin. *Geosciences*, 9(6), 262. <https://doi.org/10.3390/geosciences9060262>
- Stone, W. C., Cheok, G., & Lipman, R. (2000). Automated earthmoving status determination. In *Robotics, 2000*, 111–119. [https://doi.org/10.1061/40476\(299\)14](https://doi.org/10.1061/40476(299)14)
- Strozzi, T., Teatini, P., & Tosi, L. (2009). TerraSAR-X reveals the impact of the mobile barrier works on Venice coastland stability. *Remote Sensing of Environment*, 113(12), 2682–2688. <https://doi.org/10.1016/j.rse.2009.08.001>
- Suriya, S., & Mudgal, B. V. (2012). Impact of urbanization on flooding: The Thirusoolam sub watershed—A case study. *Journal of Hydrology*, 412, 210–219. <https://doi.org/10.1016/j.jhydrol.2011.05.008>
- Tamma, S. (2019). Transfer learning using vgg-16 with deep convolutional neural network for classifying images. *International Journal of Scientific and Research Publications (IJSRP)*, 9(10), 143–150. <https://doi.org/10.29322/IJSRP.9.10.2019.p9420>
- Tan, L. Q., Yunus, N. A., Khu, W. H., & Abd Hamid, M. K. (2021). Quality Prediction of Refined, Bleached and Deodorised Palm Oil using Multiple Least Squares Regression. *Chemical Engineering Transactions*, 89, 1–6. <https://doi.org/10.3303/CET2189001>
- Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., ... & Birchfield, S. (2018). Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 969–977. <https://arxiv.org/abs/1804.06516v3>
- Turpin, A., and Scholer, F. (2006). User performance versus precision measures for simple search tasks. In *29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, WA, 11–18. <https://doi.org/10.1145/1148170.1148176>
- United States Census Bureau (2019). Coastline America. <https://www.census.gov/content/dam/Census/library/visualizations/2019/demo/coastline-america.pdf> (Accessed 12/19/2021).
- Van Zandt, S., Peacock, W. G., Henry, D. W., Grover, H., Highfield, W. E., & Brody, S. D. (2012). Mapping social vulnerability to enhance housing and neighborhood resilience. *Housing Policy Debate*, 22(1), 29–55. <https://doi.org/10.1080/10511482.2011.624528>
- Vojtek, M., Petroselli, A., Vojteková, J., & Asgharina, S. (2019). Flood inundation mapping in small and ungauged basins: Sensitivity analysis using the EBA4SUB and HEC-RAS modeling approach. *Hydrology Research*, 50(4), 1002–1019. <https://doi.org/10.2166/nh.2019.163>
- Wada, K. (2016). LabelMe: Image Polygonal Annotation with Python. <https://github.com/wkentaro/labelme>
- Wang, R. Q., Mao, H., Wang, Y., Rae, C., & Shaw, W. (2018). Hyper-resolution monitoring of urban flooding with social media and crowdsourcing data. *Computers & Geosciences*, 111, 139–147. <https://doi.org/10.1016/j.cageo.2017.11.008>
- Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of The IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 390–391. <https://arxiv.org/abs/1911.11929>
- Wang, T. (2021). Counting the financial cost of deadly flooding in China's central city of Zhengzhou. *South China Morning Post*. <https://www.scmp.com/video/scmp-originals/3143124/counting-financial-cost-deadly-flooding-chinas-central-city-zhengzhou>. (Accessed 08/08/2021)
- Ward, P. J., Jongman, B., Kumm, M., Dettinger, M. D., Weiland, F. C. S., & Winsemius, H. C. (2014a). Strong influence of El Niño Southern Oscillation on flood risk around the world. *Proceedings of the National Academy of Sciences*, 111(44), 15659–15664. <https://doi.org/10.1073/pnas.1409822111>
- Ward, P. J., Van Pelt, S. C., De Keizer, O., Aerts, J. C. J. H., Beersma, J. J., Van den Hurk, B. J. J. M., & Te Linde, A. H. (2014b). Including climate change projections in probabilistic flood risk assessment. *Journal of Flood Risk Management*, 7(2), 141–151. <https://doi.org/10.1111/jfr3.12029>
- Wedajo, G. K. (2017). LiDAR DEM Data for flood mapping and assessment; opportunities and challenges: A Review. *J. Remote Sens. Gis*, 6, 2015–2018. <https://doi.org/10.4172/2469-4134.1000211>
- Wing, O. E., Pinter, N., Bates, P. D., & Kousky, C. (2020). New insights into US flood vulnerability revealed from flood insurance big data. *Nature Communications*, 11(1), 1–10. <https://doi.org/10.1038/s41467-020-15264-2>

- Wing, O. E., Bates, P. D., Smith, A. M., Sampson, C. C., Johnson, K. A., Fargione, J., & Morefield, P. (2018). Estimates of present and future flood risk in the conterminous United States. *Environmental Research Letters*, 13(3), 034023. <https://doi.org/10.1088/1748-9326/aaac65>
- Winsemius, H. C., Aerts, J. C., Van Beek, L. P., Bierkens, M. F., Bouwman, A., Jongman, B., ... & Ward, P. J. (2016). Global drivers of future river flood risk. *Nature Climate Change*, 6(4), 381–385. <https://doi.org/10.1038/nclimate2893>
- Xie, J., Chen, H., Liao, Z., Gu, X., Zhu, D., & Zhang, J. (2017). An integrated assessment of urban flooding mitigation strategies for robust decision making. *Environmental Modelling & Software*, 95, 143–155. <https://doi.org/10.1016/j.envsoft.2017.06.027>
- Xu, Z., Wang, S., Stanislawski, L. V., Jiang, Z., Jaroenchai, N., Sainju, A. M., ... & Su, B. (2021). An attention U-Net model for detection of fine-scale hydrologic streamlines. *Environmental Modelling & Software*, 140, 104992. <https://doi.org/10.1016/j.envsoft.2021.104992>
- Zhan, C., Ghaderibaneh, M., Sahu P., & Gupta, H. (2021). DeepMTL: Deep Learning Based Multiple Transmitter Localization, 2021 IEEE 22nd International Symposium on a World of Wireless, *Mobile and Multimedia Networks (WoWMoM)*, 41–50, <https://doi.org/10.1109/WoWMoM51794.2021.00017>.
- Zhou, Y., Wu, W., Nathan, R., & Wang, Q. J. (2021). A rapid flood inundation modelling framework using deep learning with spatial reduction and reconstruction. *Environmental Modelling & Software*, 105112. <https://doi.org/10.1016/j.envsoft.2021.105112>
- Zhu, L., Xie, Z., Liu, L., Tao, B., & Tao, W. (2021). Iou-uniform r-cnn: Breaking through the limitations of rpn. *Pattern Recognition*, 112, 107816. <https://doi.org/10.1016/j.patcog.2021.107816>.