Automated Floodwater Extent Detection on Roadways from Video Using Edge Analysis

Behrouz Salahshour
Ph.D. candidate
Department of Civil and Environmental Engineering
Old Dominion University, Norfolk, VA, 23529
Email: bsala001@odu.edu

Dr. Mecit Cetin
Professor
Department of Civil and Environmental Engineering
Old Dominion University, Norfolk, VA, 23529
Email: mcetin@odu.edu

Dr. Khan Iftekharuddin
Professor
Department of Engineering and Technology
Old Dominion University, Norfolk, VA, 23529
Email: kiftekha@odu.edu

Word Count: 3,681 words + 1 table (250 words per table) = 3,931 words

Submitted [Aug 1st ,2023]

ABSTRACT

Recurrent flooding is becoming more prevalent due to climate change, heavy rain, and sea level rise causing significant disruption to the surface transportation network. Detecting and monitoring the progression of road inundation due to flooding is important for effective management of transportation systems and informing the traveling public. Most of the current studies found in literature focus on more severe floodings and are suitable for larger scale analyses. In this paper we propose a 4-step methodology using a combination of deep learning and traditional computer vision techniques to estimate the extent of flooding on roadways from video data. The model estimates the progression of flooding across the travel lanes. We have collected data from a flooding event in Norfolk, VA using a camara installed on a street pole to test our methodology. The results indicate good agreement between the proposed automated method and ground truth.

Keywords: Flood, Flood extent, Partial flooding, Recurrent floodings, Computer vision, Edge detection, CCTV

INTRODUCTION

Floods are exceedingly disrupting public life and are among the most dangerous natural disasters in the United States [1]. Aside from major floodings, recurrent nuisance floodings are becoming more prevalent due to extreme rainfall and high tides that are exacerbated by climate changes. Such flooding events can occur frequently specially in coastal areas where rising sea levels are causing more tidal floodings than ever before [2]. Although partial floods can be relatively less dangerous, due their increased frequency, the disruption to infrastructures such as roadway blockings are becoming a major issue for many cities [3]. For example, nuisance flooding has increased 325% since 1960 in Norfolk, VA, and the city is projected to experience well over 200 flood events in the year 2049 [4]. Since roadway closures are extremely costly to traffic operations, more research needs to be done to determine when and where these closures are absolutely necessary.

On-site gauge measurement methods can be implemented to determine flood depth and alert drivers and city traffic managers about risky corridors [5]. However, installing such sensors on every street can be extremely costly [6]. Remote sensing methods are an alternative method that have been increasingly used for flood mapping and flood extent estimation over the years. Such methods use remotely captured data such as synthetic aperture radar (SAR) data from satellites to estimate the flood extent and depth. For instance, Tiampo et. al [7] developed a machine learning based model using SAR data to develop large-scale flood extent maps and provide information for a global flood alerting system called DisasterAware©. Although these novel methodologies can help mitigate the disastrous effects of major flooding events, most are focused on large-scale mappings as opposed to fine-grained street level analysis [5, 7-12]. Therefore, a huge gap still exists in the literature for studies of partial flooding events that can still affect traffic operations.

With an increased number of surveillance cameras installed throughout cities' infrastructures, an opportunity exists to use this optical imagery data to develop remote sensing methods for flood extent and depth estimation. Furthermore, UAVs are becoming more commonly used and methods based on optical imageries can also be applied to their feeds for a detailed fine-grade analysis at any chosen location. This type of data makes it possible to estimate flood extent variations inside a specific region such as a street. For example, Hashemi-Beni et. al. [13] proposed an integrated method using a convolutional neural network (CNN) classifier combined with region growing techniques to develop flood extent maps from UAV optical imagery. In a similar study, Rahnemoonfar et. al. [14] created a high-resolution UAV imagery captured after hurricane Harvey and emphasize the benefits of using UAV imagery as compared to satellite imagery for a more accurate analysis of disaster affected areas. Kharazi et. al. [15] emphasized how using satellite and DEM data can yield large errors. They propose a deep learning method using stop signs as the object of interest for determining flood water depth using photos taken from flooded roads and intersections to bridge this research gap. After using an object detector to identify stop signs, they make use of canny edge detector and probabilistic Hough transformation to calculate the pole length and subsequently the floodwater depth. Cem et. al. [16] developed a deep learning method for flood depth prediction using vehicles as the object of interest. In their methodology they find the depth of the water after localizing the vehicle tires and estimating the water depth in relation to the tire. Since vehicles are abundantly found on inundated roads, this method has a higher potential to be applicable to a wider range of images compared to methods using other objects as reference.

Although deep learning models are the trendiest method of choice for many researches in flood extent detection in recent years, more traditional methods can also be used to determine flood extent with high reliability and lower computational costs. For instance, Ariawan et. al. [17] use a traditional grayscale and thresholding method with edge detection to automatically determine the water level on riverbed gauges. They first determine the region of interest, then adjust the brightness and contrast to make their inputs more uniform before finding water edge points. Zhang et. al. [18] argue that many methods in the literature cannot be used for real-time flood detection due to their high processing time. They suggest considering edge detection techniques for improving the monitoring of flooded areas. Akbar et. al. [19] designed a simulation environment to develop a four-step early flood detection and warning system using traditional computer vision methods found in OpenCV library such as thresholding and edge detection. They report an accuracy of almost 96% in the laboratory setting for their proposed method. More recently, Utomo et. al. [20] performed a similar experiment in the lab to prove the reliability and accuracy of using edge detection algorithms for detecting water levels in real-time. They emphasize that many models using deep learning methods are much more computationally expensive and are not well suited for early flood warnings.

In this article we use a combination of deep learning models and traditional computer vision models to estimate the flood extent across lanes of streets. We use a static camera installed on a street pole to test our proposed methodology. We use an object detector to first find the frames in which no vehicles are present on the road. Then an edge gradient is calculated using several consecutive frames. Consequently, we use a thresholding criterion to determine the flooded sections of each lane. We compare these results with the ground truth data obtained from manually labeling the frames. Our results indicate that this methodology can effectively estimate the flood extent in real-time.

DATA COLLECTION AND METHODOLOGY

We have collected video footage of a flooding event in Norfolk, VA, on October 3rd, 2022. The video is collected using a rechargeable camera installed on a powerline pole on the northbound of Hampton Boulevard in a residential area with a speed limit of 35mph. The collected video consists of around 40K frames over approximately 22 minutes at the rate of 30fps. Figure 1 shows a sample frame of this video footage.

Figure 1- Sample Frame from Video footage

The choice of methodology for flood extent estimation depends on multiple factors such as available computational power, accessible data types, and the rate at which warnings need to be issued. In this study our aim is to provide a pipeline for flood extent estimation at the street level which requires minimal computational power yet is reliable and robust using video data from CCTVs or UAVs. We aim to make the pipeline accessible to use for public agencies by only using open-sourced tools and software. To achieve this goal, we designed a 4-step pipeline as shown in Figure 2.

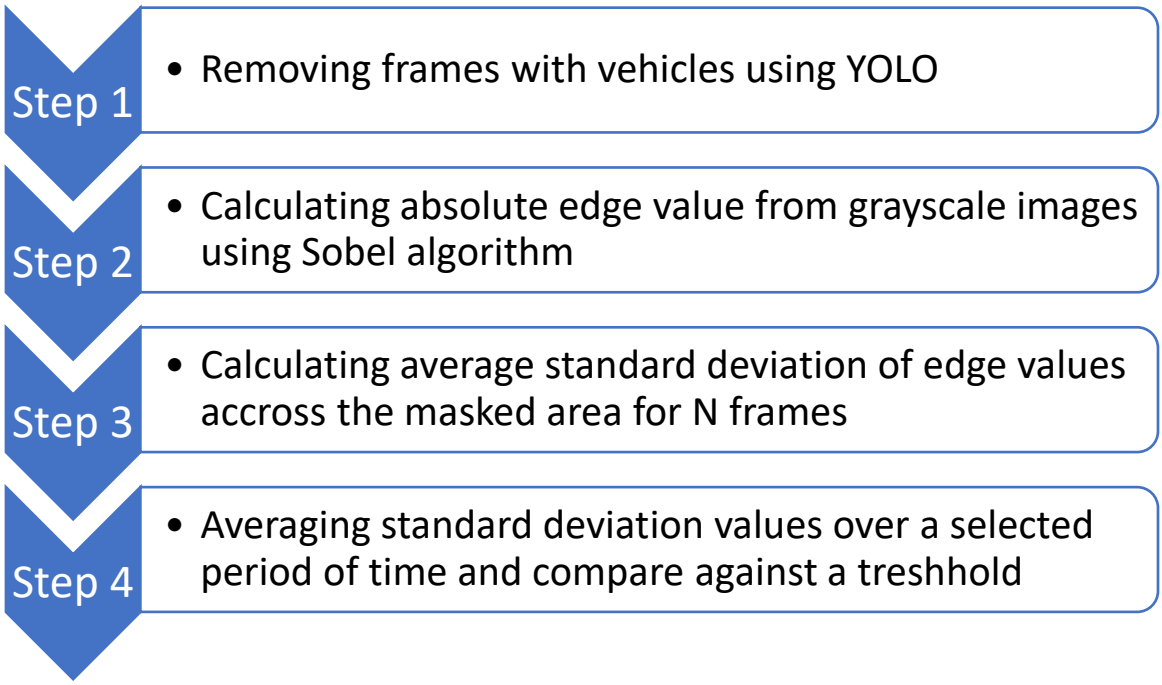

Figure 2- Proposed methodology for flood extent estimation

Step 1: First, we need to find the time periods where no cars are present on the road so that only water is visible on the roadway. This is possible for the selected corridor since there are breaks in traffic flow due to the upstream traffic signals. This step is crucial to ensure no false alarms are created. For this step, we use off the shelf YOLOv5 object detector trained on COCO dataset to find the bounding boxes of vehicles passing through the street. YOLO is a fast yet accurate object detector that can provide real-time predictions about objects seen in an image or video frame. There is no need to fine-tune this model as the accuracy for vehicle detection is already satisfactory for our application. This greatly reduces the time and effort needed by professionals for wide implementation. The input for YOLO is each frame in the video feed and the output is the bounding boxes of objects of interest which are vehicles in our case. We define a region of interest (ROI) in which our flood extent detection is designed to work. After finding the bounding boxes, we only use the frames in which no bounding boxes overlap with the ROI. To ensure that no cars are present in the ROI, we have also refrain from using a few frames after a vehicle has just passed the region. This is done since the object detector might fail to detect vehicles when they are partly present in the frame as they are leaving the scene. This extra safety measure also ensures that the effects of the vehicle on the water body on the roadway is not producing wrong readings in the next steps.

Step 2: After extracting frames in which only water is present on the street, we transform each frame to grayscale and feed each frame to an edge detector such as Sobel or Laplace to calculate the edge values or gradient in the region of interest. This step is performed using the open-source library OpenCV which provides many tools for computer vision analysis. We then extract the edge values in the specific masked area that is used to represent different street sections. Figure 3 shows the output of this step. The region of study is marked with a rectangle.

Step 3: After calculating edge values for each frame, we then calculate the standard deviation of edge values in each pixel of the masked area to detect changes from frame to frame which represents water waves and ripples. This ensures that only any other edges present on the roadway such as lane markings are not mistaken for water edges since they are stationary, and their standard deviation would be zero. We then calculate the average or median value of edges along the direction of the masked crossed section area and can plot their values over each section of each lane. Figure 4 shows three of these calculated standard deviations for different times of the studied video.

Step 4: After extracting the average standard deviation of edge values along the masked area, we use a predefined threshold to determine whether each particular section is flooded or not. If the standard deviation value meets this criterion, it means that movement of edges has been present over that section in the chosen period. Since other objects have been already removed, these movements are representative of waves and ripples produced by a water body.

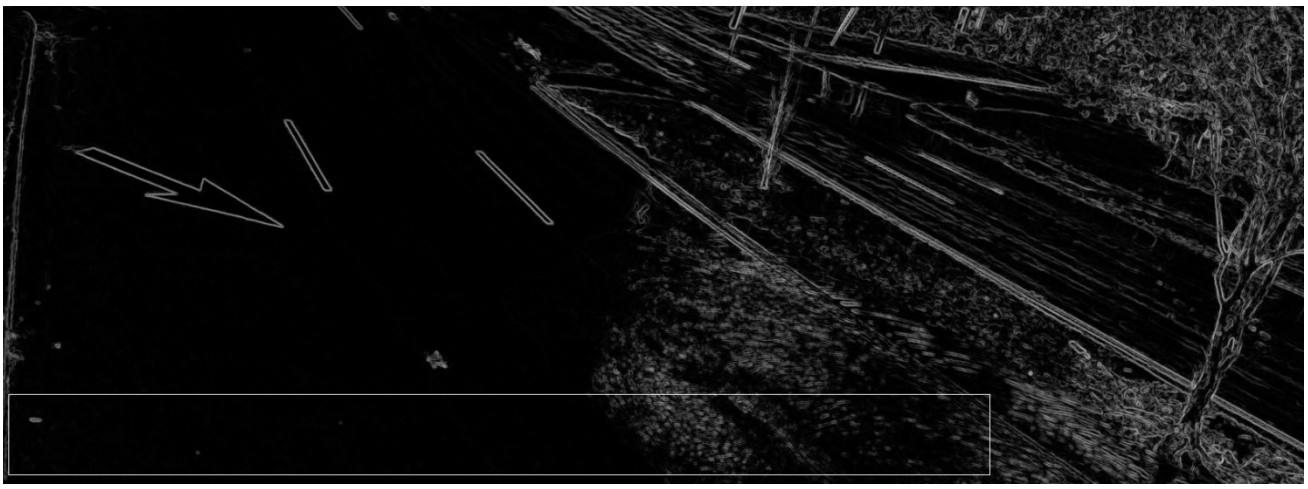

Figure 3- Edge output from Sobel algorithm with the region of interest marked by a rectangle.

The average edge gradient explained in the methodology can be described mathematically as:

$$\bar{\sigma}_x = \sum_{y=0}^{Ymax} \sigma_{x,y} \Big/ Y_{max} \qquad \textit{Equation 1}$$

Where $\bar{\sigma}_x$ is the average standard deviation over the height of the chosen rectangular section, Y_{max} is the height of the rectangle, and $\sigma_{x,y}$ is the standard deviation for each pixel in the rectangle calculated as:

$$\sigma_{x,y} = \sqrt{\sum_{n=1}^{N_{frames}} (E_{x,y}^{n} - \mu_{x,y}) \Big/ N_{frames}} \qquad \textit{Equation 2}$$

Where $E_{x,y}^{n}$ is the absolute edge value calculated from the edge detection model for nth frame, N_{frames} is the total number of frames used to calculate the standard deviation, and $\mu_{x,y}$is the average edge value for each pixel location x, y calculated by:

$$\mu_{x,y} = \frac{1}{N_{frames}} \sqrt{\sum_{n=1}^{N_{frames}} E_{x,y}^{n}} \qquad \textit{Equation 3}$$

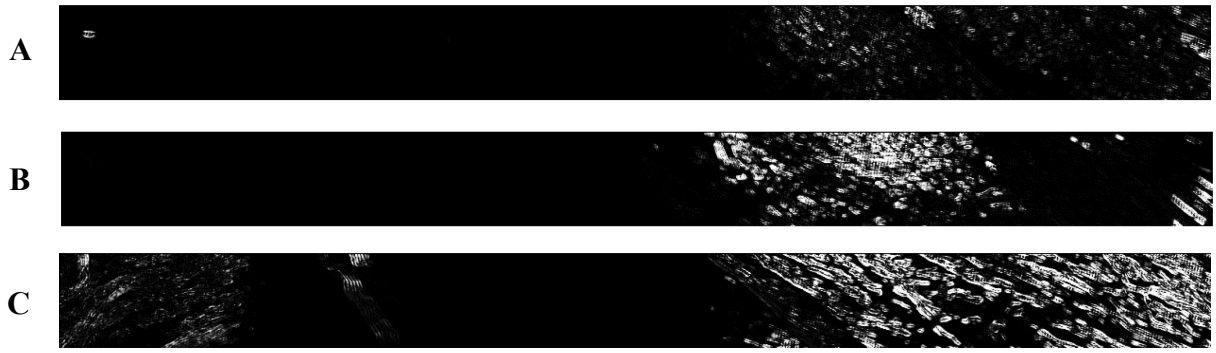

Figure 4- Standard Deviation of Edge values in the selected region
A) At the beginning of the video B) Half point of the video C) At the end of the video

RESULTS AND DISCUSSION

The proposed methodology was applied to a video capturing a flooding event for testing. Figure 5 shows the results for three different points of the video. As it can be seen in the graph, each section of the pixel locations on the graph corresponds to different parts of different lanes. Since lane 1 is the closest lane to the camera, more pixels represent this lane. The arrows on the bottom of the graph represent the ground truth location of the water body labeled manually. The green line shows the normalized average edge gradient at the beginning of the video. Since at the beginning of the video the water only exists on parts of lane 2 and lane 3, the plot peaks at around 700 pixels, corresponding to two third of lane 2 being flooded. As the flood extends and covers more of lane 2, the gradient line shifts towards the beginning of lane 2, almost covering the whole lane at the middle of the video. Finally, the red line shows the edge gradient at the end of the video when all three lanes are completely flooded. However, since the water is still shallow and hence ripple and wave heights are less pronounced in lane 1, smaller peaks are noticed in the graph. Table 1 shows the percentage error of the estimated flooded area compared to the ground truth for each lane across different sections of the video using 90 frames for each section. A threshold of 2% was used to determine the flooded regions. As it can be seen, the model is able to capture the percentage of the lane flooded with great accuracy except for lane 1 at the end of the video. As explained before, this is due to the shallow Figure 5water body which has just covered lane 1 and does not produce as many ripples and waves yet to be captured.

Table 1- Percentage Error of Flooded area detection by lane using 90 frames.

Time of Video	Estimated Percentage of Lane Flooded (Lane 1/ Lane 2/ Lane 3)	Ground Truth (Lane 1/ Lane 2/ Lane 3)	Difference Percentage
Beginning	5 / 69 / 100	0 / 66 / 100	2 / 3 / 0
Middle	0 / 88 / 100	0 / 85 / 100	0 / 3 / 0
End	75 / 100 / 100	100 / 100 / 100	25 / 0 / 0

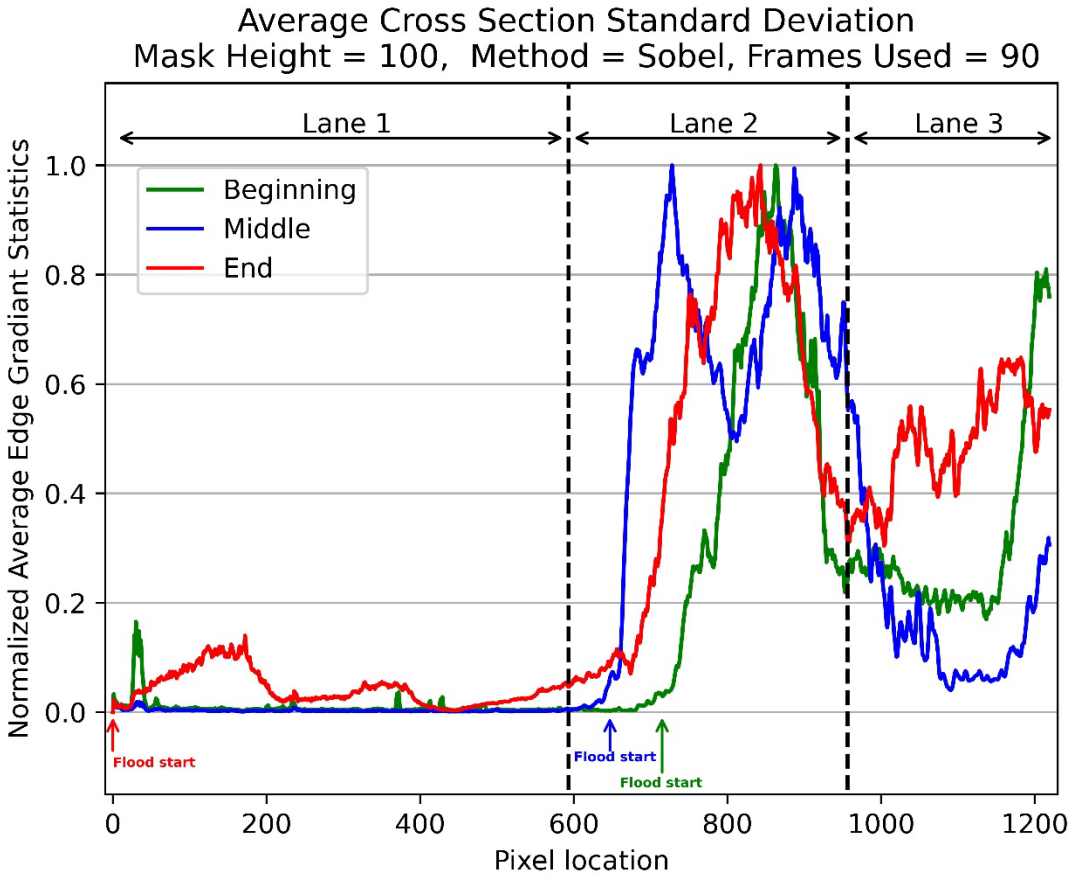

Figure 5- Normalized averaged edge gradient statistics over the selected region of interest

Effect of the Number of Frames

The number of frames used for calculating the standard deviation of edge gradient is an important factor to be considered. When few numbers of frames are used, the gradient will only capture the few ripples and waves present in those frames, hence it might miss some flooded sections specially in lower depth regions. Figure 6 shows the effect of the number of frames used for calculating the standard deviation for the end of the video when all 3 lanes are flooded. As it can be seen, when the number of frames is increased, more flooded areas of lane 1 are captured. In other words, when the gradient is calculated over a larger period, more ripples and waves can be captured specially in lower depth regions such as lane 1 of our case study here. It is worth noting that while considering more frames can be beneficial in this way, this will add to the computational

cost and will slightly delay the calculations (5 seconds in our case for using 150 frames in a 30fps video).

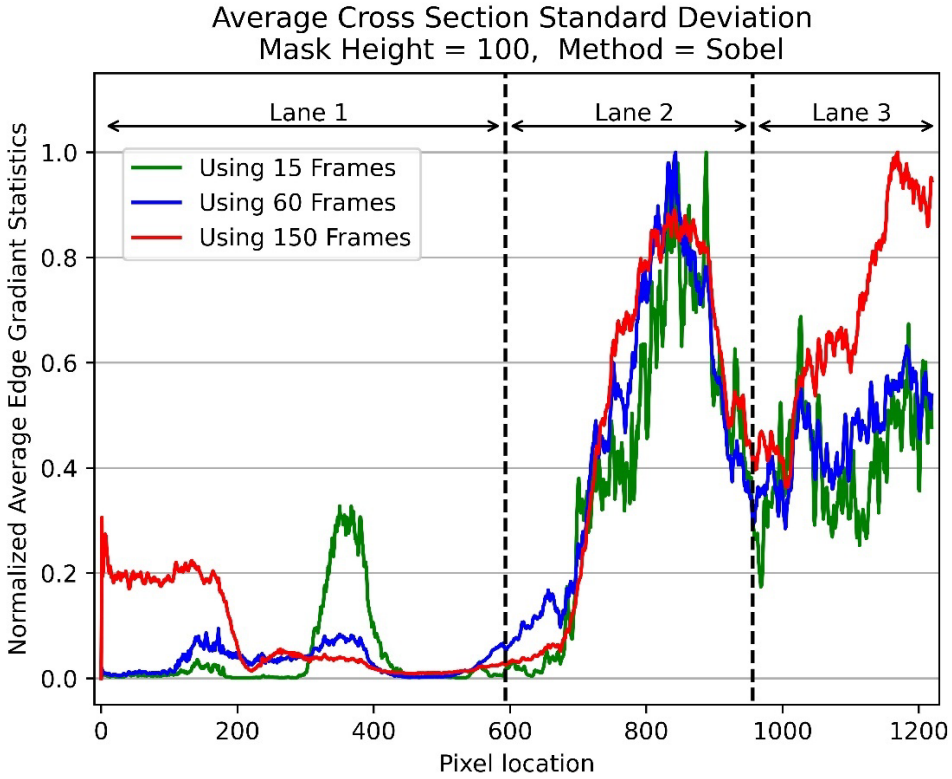

Figure 6- Effect of Number of Frames Used for Standard Deviation Calculations

Effect of ROI Size

The height of the region of interest chosen for edge calculations can potentially be a factor to consider when using this model. Figure 7 shows the effect of the height of the rectangular masked region on the average edge gradient in the middle portion of the video. As shown in this figure, the size of the ROI does not greatly affect the average standard deviation values as long as the region is well representative of all lanes. Hence, instead of using a larger ROI, it is recommended to have several ROIs in different sections of the road if needed.

Effect of Edge Detection Model

We have also explored the possible effects of the edge detection model on the results. We have chosen three famous models of Sobel, Laplace, and Prewitt for this purpose. Figure 8 shows the effect of choosing different models applied to the beginning section of the video. As it can be seen, the choice of edge detection model does not greatly affect the results. Sobel and Prewitt are almost identical while Laplace is less capable of detecting ripples on the farthest lane from the camera which has fewer representative pixels.

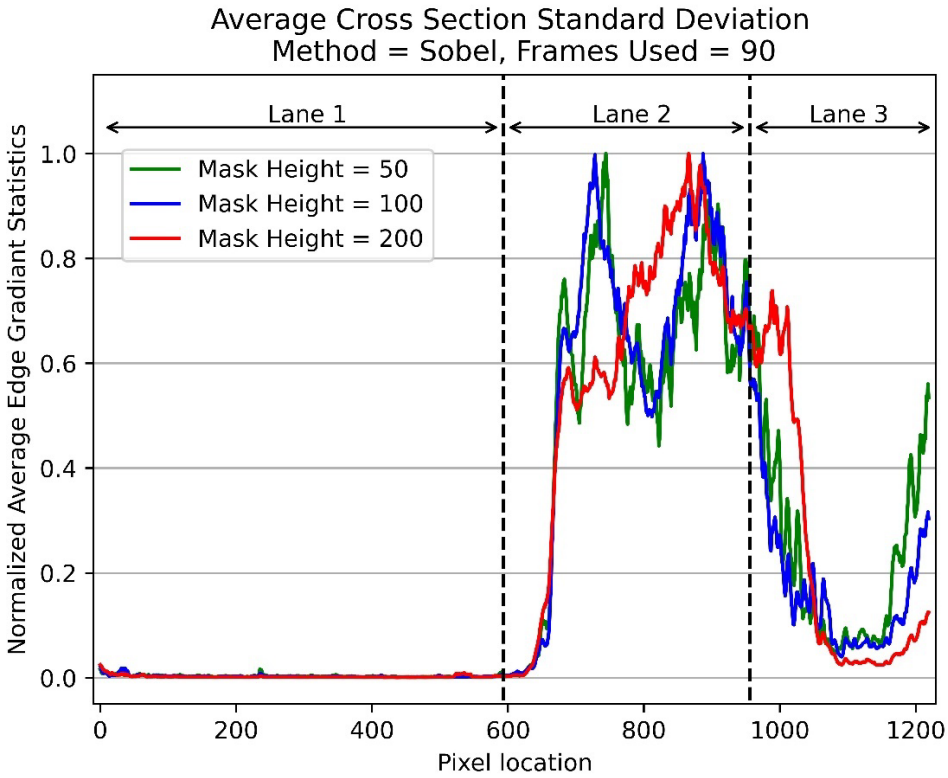

Figure 7- Effect of ROI Size

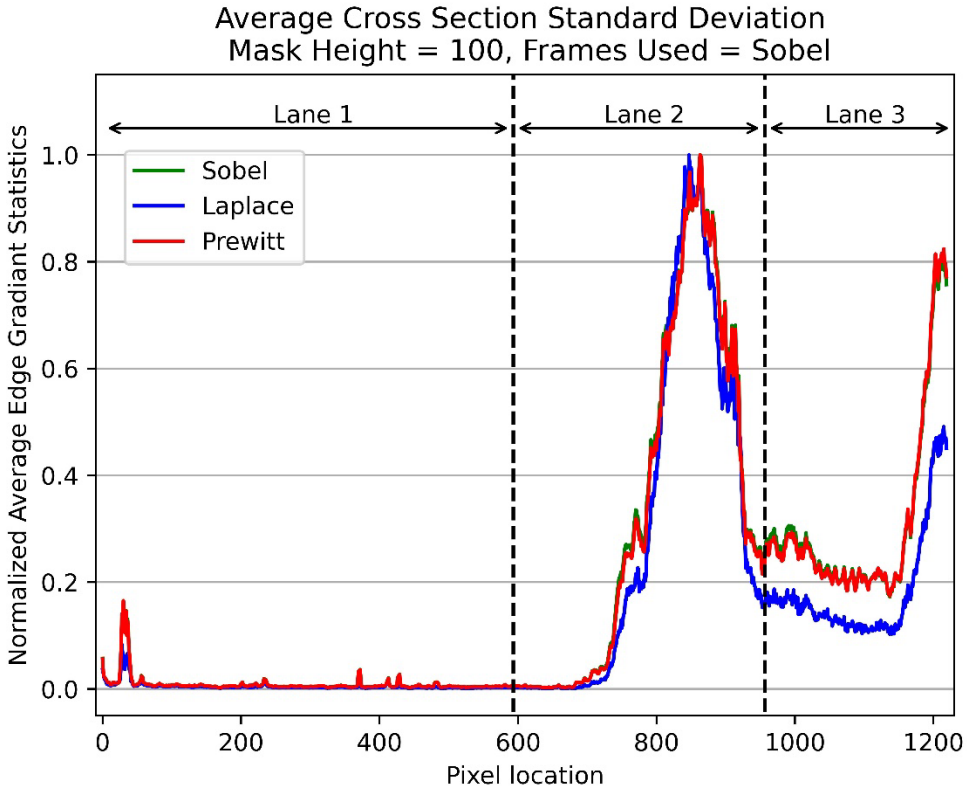

Figure 8- Effect of Edge Detection Model

CONCLUSIONS

In this paper we have proposed a 4-step methodology consisting of a deep learning model for object detection and edge detection for determining the extent of flood at street level. The standard computer vision techniques used for ripple and wave detection are efficient enough to be run in real-time and can be used for ordinary CCTV and UAV video footages. As shown, the proposed methodology can achieve good flood extent estimation despite its relative simplicity. Since the computational cost is relatively lower, this model can be combined with other flood detection models such as semantic segmentation models for verification purposes.

Currently we are using a thresholding criterion for choosing the flooded areas. In future works we will explore the use of change point detection models for detecting flooded areas more robustly. A combination of this method with optical flow analysis can also be investigated.

ACKNOWLEDGMENTS

This research was funded by the NSF Grant SCC-IRG Track 2: Scalable Modeling and Adaptive Real-time Trust-based Communication (SMARTc) System for Roadway Inundations in Flood-Prone Communities (Award number: 1951745). We would also like to mention that some of the utility functions for simple preprocessing steps such as masking the selected area using OpenCV has been facilitated with the help of ChatGPT.

AUTHOR CONTRIBUTIONS

All authors contributed to the conceptual development of the models. Salahshour processed and analyzed the video data, conducted the literature review, derived the mathematical equations, and wrote the Python code for image analysis. Cetin helped in the development of statistical models and Iftekharuddin in computer vision methods. All authors contributed to model validation and performance evaluation.

REFERENCES

1. Lorie, M., et al., *Modeling coastal flood risk and adaptation response under future climate conditions.* Climate Risk Management, 2020. **29**: p. 100233.
2. Spanger-Siegfried, E.F., Melanie; Dahl, Kristina, *Encroaching Tides: How Sea Level Rise and Tidal Flooding Threaten U.S. East and Gulf Coast Communities over the Next 30 Years*. 2014.
3. Alabbad, Y., et al., *Assessment of transportation system disruption and accessibility to critical amenities during flooding: Iowa case study.* Science of The Total Environment, 2021. **793**: p. 148476.
4. Burgos, A., et al., *Future Nuisance Flooding in Norfolk, VA, From Astronomical Tides and Annual to Decadal Internal Climate Variability.* Geophysical Research Letters, 2018. **45**.
5. Abdul Tauqeer, Haroon Stephen, , Sajjad Ahmad, *Review of Remote Sensing Techniques Used for Mapping Flood Extent, Flood Monitoring, Flood Hazard, Exposure and Damages, and Flood Resilience in Pakistan*, in *World Environmental and Water Resources Congress 2023*. 2023. p. 1215-1226.
6. Nair, B.B. and S. Rao. *Flood water depth estimation — A survey*. in *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*. 2016.
7. Tiampo, K., et al. *A Machine Learning Approach to Flood Depth and Extent Detection Using Sentinel 1A/B Synthetic Aperture Radar*. in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*. 2021.
8. Gebrehiwot, A. and L. Hashemi-Beni, *A METHOD TO GENERATE FLOOD MAPS IN 3D USING DEM AND DEEP LEARNING.* Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., 2020. **XLIV-M-2-2020**: p. 25-28.
9. Soria-Ruiz, J., et al., *Flooded Extent and Depth Analysis Using Optical and SAR Remote Sensing with Machine Learning Algorithms.* Atmosphere, 2022. **13**(11): p. 1852.
10. Martinis, S., et al. *Automatic Near-Real Time Flood Extent and Duration Mapping based On Multi-Sensor Earth Observation Data*. in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*. 2020.
11. Hamidi, E., et al., *Fast Flood Extent Monitoring With SAR Change Detection Using Google Earth Engine.* IEEE Transactions on Geoscience and Remote Sensing, 2023. **61**: p. 1-19.
12. Seydi, S.T., et al., *Fusion of the Multisource Datasets for Flood Extent Mapping Based on Ensemble Convolutional Neural Network (CNN) Model.* Journal of Sensors, 2022. **2022**: p. 2887502.
13. Hashemi Beni, L. and A. Gebrehiwot, *Flood Extent Mapping: An Integrated Method using Deep Learning and Region Growing Using UAV Optical Data.* IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2021. **PP**: p. 1-1.
14. Rahnemoonfar, M., et al., *FloodNet: A High Resolution Aerial Imagery Dataset for Post Flood Scene Understanding*. 2020.
15. Alizadeh Kharazi, B. and A.H. Behzadan, *Flood depth mapping in street photos with image processing and deep neural networks.* Computers, Environment and Urban Systems, 2021. **88**: p. 101628.

16. Sazara, C., et al., *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, 2022. **4**(1): p. 85-101.
17. Ariawan, A., et al., *Image Processing-Based Flood Detection.* Proceedings of the 10th National Technical Seminar on Underwater System Technology 2018, 2019.
18. Zhang, Q., N. Jindapetch, and D. Buranapanichkit. *Investigation of Image Edge Detection Techniques Based Flood Monitoring in Real-time*. in *2019 16th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. 2019.
19. Akbar, Y., A. Musafa, and I. Riyanto, *Image Processing-based Flood Detection for Online Flood Early Warning System*. 2014.
20. Utomo, S.B., J.F. Irawan, and R.R. Alinra, *Early warning flood detector adopting camera by Sobel Canny edge detection algorithm method.* Indonesian Journal of Electrical Engineering and Computer Science, 2021. **22**(3): p. 1796-1802.