A Computer Vision and AI Based Solution to Determine the Change in Water Level in Stream

Priyanjani Chandra
pchandra@niu.edu
Northern Illinois University
DeKalb, Illinois

Pratool Bharti (advisor)
pbharti@niu.edu
Northern Illinois University
DeKalb, Illinois

Michael Papka (advisor)
papka@niu.edu
Northern Illinois University
DeKalb, Illinois
Argonne National Laboratory
Lemont, Illinois

ABSTRACT

Flooding is one of the most dangerous weather events today. Between 2015 – 2019 on average, it has caused more than 130 deaths every year in the USA alone. World Health Organization has reported that, between 1998 – 2017, floods have affected more than 2 billion people worldwide. The devastating nature of flood necessitates the continuous monitoring of water level in the rivers and streams in flood-prone areas to detect the incoming flood. In this study, we have designed and implemented a computer vision and AI-based system that continuously detects the water level in the creek. Our solution employs an effective template matching algorithm on edge map images to find the water level coordinates. Next, a linear regression based model finds a straight line through these coordinates, that represents the water level. We evaluated our algorithms on 200 images across several days and achieved 0.949 R^2 score. We also tested our model on 68,890 images from 96 days and it exhibited similar trend when compared to the corresponding data from USGS.

CCS CONCEPTS

• **Computing methodologies → Machine learning**; **Computer vision**.

KEYWORDS

computer vision, edge detection, template matching, linear regression

ACM Reference Format:
Priyanjani Chandra, Pratool Bharti (advisor), and Michael Papka (advisor). 2020. A Computer Vision and AI Based Solution to Determine the Change in Water Level in Stream. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Due to the devastating nature of floods, there is a clear need for constant observation of the water level in the rivers and streams. Sensor-based and vision-based are the two common techniques used to detect the water level in the stream. Sensor-based systems[5][8] are easily affected by the environmental changes. While, few vision-based systems depend on gauge images[3][6], others on water body images with indicators[4][7][11]. These vision-based approaches require costly infrastructure setup, provide less accuracy, and require manual observations. The most adequate solution would be the one that is cost-effective, automated and requires low maintenance. In this realm, we propose a computer vision and machine learning based solution to detect the water level in the stream with high precision in real-time. The main objective of the proposed project is to leverage the image of the creek to automatically detect the change in water level in the stream with minimal error. To achieve this, we will employ edge detection, template matching, and machine learning algorithms.

2 DATA COLLECTION

The data, resources, and infrastructure required to conduct the research are provided by the data, devices, and interaction Lab (ddiLab) at Northern Illinois University (NIU). The images are taken through an Array of Things (AoT) [2] node installed on a light pole in front of the Computer Science building at NIU. AoT is an experimental system to study urban life and the environment by collecting real-time data using a sensor platform with edge computing capability. The images are collected at a frequency of 1 Hz and transmitted to the Hartley server at ddiLab via NIU WiFi for further processing. Although, AoT node is equipped with edge computing resources, all the processing and computing for this study has been done at server side on Hartley.

3 WATER LEVEL DETECTION

The water level detection system (Fig 1) presented in this work is executed in 5 steps. *First*, the images are classified as day or night based on pixel values in the images, and only the day images are further processed. *Second*, we need to identify the location of the bridge in the image. To do so, we select a template image of the bridge to search it in the new image by employing a **template matching** [9] algorithm. We used two metrics for matching - normed square difference and normed cross correlation. Both of these metric calculate a number that denotes the extent to which the template and the portion of the original are matching. While normed square difference denotes a complete match by a 0 (indicating no difference between the template and the portion of original), normed cross correlation denotes a complete match by 1. Only when both metrics found the template at same location, we considered the template is found otherwise ignored the results. Template matching doesn't work well if there is a large variation between the template image and new image which is often the case in RGB image. To mitigate this issue, we transformed the RGB image to an edge map image by leveraging the **holistically-nested edge detection(HED)** [10] model. HED is a state-of-the-art edge detection model based on a neural network algorithm. It inputs RGB image and outputs its grayscale edge map image.

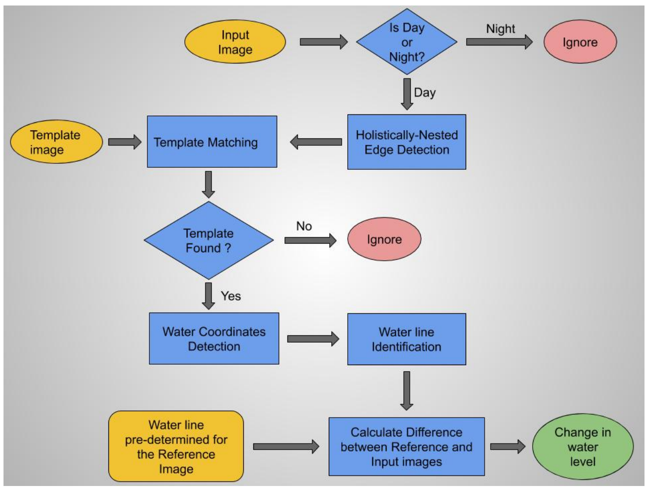

Figure 1: System Workflow

Third, once the bridge location in the new image is detected, we find the **water level coordinates** in the corresponding HED image. To accomplish it, we traversed through each column of template found area and select a coordinate in each column where the pixel value is changing drastically while proceeding from bottom to top. If the coordinates found are random and are not linear in the HED image, those images are ignored by calculating variance in the coordinates. *Fourth*, we find a **straight line** that represents the water level by passing through the water level coordinates by leveraging the reference water line. *Fifth*, finally we calculate the distance between the reference and new image water level lines to calculate the water level in the new image.

4 PERFORMANCE

We achieved high correlation with R^2 score of 0.949 when tested our results on 200 images with ground truth. We also tested our model on 68,890 images from 96 days and it exhibited similar trend when compared to the corresponding data from United States Geographical Survey[1].

5 CONCLUSION

Our water level detection system is a low-cost solution to automatically detect the water level in the stream. The computer vision and AI-based system achieved high precision in detecting the height of the water level. We validated our results and achieved a high correlation on 200 images. Our future goal is to deploy the system on the AoT node itself to reduce the communication overhead.

REFERENCES

[1] [n.d.]. USGS Current Conditions for the Nation: Build Time Series. https://waterdata.usgs.gov/nwis/uv?

[2] Pete Beckman, Rajesh Sankaran, Charlie Catlett, Nicola Ferrier, Robert Jacob, and Michael Papka. 2016. Waggle: An open sensor platform for edge computing. In *2016 IEEE SENSORS*. 1–3.

[3] Machiel Bruinink, Aswin Chandarr, Maja Rudinac, P.-J Overloop, and Pieter Jonker. 2015. Portable, automatic water level estimation using mobile phone cameras. *Proceedings of the 14th IAPR International Conference on Machine Vision Applications, MVA 2015* (07 2015), 426–429. https://doi.org/10.1109/MVA.2015.7153102

[4] Thomas Hies, Suresh Babu Parasuraman, Y. Wang, R. Duester, Hans Eikaas, and Kok Tan. 2012. Enhanced water-level detection by image processing.

[5] O. Intharasombat and P. Khoenkaw. 2015. A low-cost flash flood monitoring system. In *2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE)*. 476–479.

[6] Yan-Ting Lin, Yi-Chun Lin, and Jen-Yu Han. 2018. Automatic Water-Level Detection Using Single-Camera Images with Varied Poses. *Measurement* 127 (05 2018). https://doi.org/10.1016/j.measurement.2018.05.100

[7] Arci Jeico P. Malabanan and Vladimir Y. Mariano. 2010. Real-time Water Level Monitoring Using Image Sequence Processing.

[8] M. Mousa, E. Oudat, and C. Claudel. 2015. A Novel Dual Traffic/Flash Flood Monitoring System Using Passive Infrared/Ultrasonic Sensors. In *2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems*. 388–397.

[9] OpenCV. [n.d.]. Template Matching in OpenCV. https://docs.opencv.org/3.4/de/da9/tutorial_template_matching.html.

[10] Saining Xie and Zhuowen Tu. 2015. Holistically-Nested Edge Detection. In *2015 IEEE International Conference on Computer Vision (ICCV)*. 1395–1403.

[11] Jaehyoung Yu and Hernsoo Hahn. 2010. Remote Detection and Monitoring of a Water Level Using Narrow Band Channel. *J. Inf. Sci. Eng.* 26 (01 2010), 71–82.