An Experimental Approach on Detecting and Measuring Waterbody through Image Processing Techniques

Beau Gray M. Habal, Elisa V. Malasaga, and Abraham T. Magpantay
FEU Institute of Technology, Manila, Philippines
Email: {bmhabal, evmalasaga, atmagpantay}@feutech.edu.ph

***Abstract*—Flood is imminent when heavy rain occurs, identifying the level of water in plain sight is difficult to achieve. There are currently available ways to detect flood water but usually are very expensive and needs a huge equipment with sensors. The research has proposed an alternative solution to expensive ways on detecting flood and water levels. The study created an application to detect body of water by using image processing technique called Region-based segmentation algorithm to detect water on the image and Canny Edge Detection with computation using Pixel Ratio on a selected water region to determine the height of the water or flood. A CCTV camera was used to capture the image and was fed on the application through the network infrastructure. Once captured, the image was processed to detect the body of water and measurement of its level. The testing of the application was done on a controlled environment and the application was able to detect the water body on the picture. It was able to detect the edge of the water based on a selected region where the water is found. The measurement of the actual height of the water, closely matches the height of stated in the application. Thus, the research has found a way to detect body of water and gauge its water level using image processing, in which, have found a way to detect and measure water affordably. This research can be a step, in future research like monitoring the streets' flood level when heavy rains occurs. This is a much more safe and affordable way to monitoring the increase and decrease of flood.**

***Index Terms*—waterbody detection, image processing, water level measurement, CCTV IP camera**

I. Introduction

Technology today is rapidly growing and inevitably changing. But in applying development of technology, human aspect of life should also be considered. Flooding is due to heavy storms, which damages different properties and causes water to overflow from the riverbanks or the sea, or even man-made which causes loss of life and property [1]. The need for new innovations and technologies is concurrent in today's generation. The use of computer technology in detecting flood can help fix flood problems or help detect early signs of flood. Measuring water level is widely used in hydraulic studies, agriculture, petroleum, chemical, food processing and many other fields [2].

There are many research and studies about flood and uses technology to help solve problems related to this. Based on this journal Prototype of Water Level Detection System with Wireless, their study focuses on water level detection system which is designed to facilitate the people in collecting water level data that can be performed in real-time. Ping sensor used as a distance sensor for detecting water level by measuring distance between sensor and water surfaces. The system consists of two modules, transmitter and receiver. Transmitter module performs water level detection and transmits it to the receiver module as a data collector. Receiver module then displays the data on the screen [3].

Another research called Design of Early Warning Flood Detection Systems for Developing Countries, discussed that in developing countries, flooding is due to natural disasters such as hurricanes and earthquakes results in massive loss of life and property. Warning communities of the incoming flood provides an effective solution to this by giving people enough time to evacuate and protect their property. However, the range of early warning system solutions introduces a tangle of conflicting requirements including cost and reliability. It has created several interesting problems from factors as diverse as technological, social, and political. The complexity of these systems and need for autonomy within the context of a developing country while remaining maintainable and accessible by nontechnical personnel provides a challenge not often solved within developed countries, much less the developing. After describing this problem, the paper discusses a proposed solution for the problem, initial experiments in implementing the solution, and lessons learned through that work [4].

A study also stated that flooding is a great threat towards mankind as it is also considered one of the most devastating natural disasters in the world. Flooding is not any abnormal scenario worldwide, since flooding results in great damages to agriculture land, residential area and even cities with high cost in lives and towards the economy of the country. The government has to spend tons of money in flood mitigation plans in afford to help the victims and also to reduce the number in the long run.

Manuscript received June 9, 2020; revised November 17, 2020.

doi: 10.12720/jait.12.1.45-50

Most of the flood mitigating plans has high cost and only can be implemented base on priority. Baring the cost and safety measures, the researchers focus on using Flood Observatory System (FOS) as a warning and alert system to efficiently monitor the critical flood prone areas in real time basis. FOS can be deployed in flood prone areas in afford to create a well-used standard for remote flood observation systems. The ability to receive real time information on flood level empowers both government and private organizations to react to imminent danger in an effective manner. With the real-time flood information, allows public safety organizations and other emergency managers to effectively plan their resource deployment within the limited time of alert. Warning as flood rises could be used to save life's and properties. The simple and practicality of a system should be useful in all means towards mankind [5].

As a known fact, flood is a noticeable disaster occurring not only in poor countries, but also in developed countries. One of the nature's most turbulent years in flooding was 1998. Floods wreaked havoc in North America, South America, Asia and Europe. Though the El Nino and La Nina were thought to be responsible for these floods, scientists have now found out that these factors combined with global warming caused by pollution and made the disasters more severe [1].

The common denominator of these studies are uncovering ways to address the problem of flooding in our environment, and the use of technology to help reduce the calamities caused by flood. Expensive equipment and tools to overcome flooding in poor countries is difficult because of expensive cost and limited resources. The government of these countries are incapacitated to supply these kinds of equipment because of lack of funding. With this context, it is eminent to look for ways to detect water and its measurement affordably.

II. Objectives

The researchers firstly aim to create an application that captures images from a CCTV camera and would detect a waterbody in the picture and measure it. Secondly, to find a way to detect waterbody and measure the level of water that is more affordable compared to mostly used approach. And lastly, to test the consistency of the detection of the waterbody and its measurement found on the picture against the actual area.

III. Image Processing Techniques

In this research, the use of image processing techniques is applied to determine the water body and measure its level. The following are the techniques used:

A. Region Based Segmentation

In order to detect the body of water in an image, an algorithm called Region-based Segmentation was used. The goal of this algorithm is to partition the original image into regions. Grayscale thresholding was used to look for the boundaries among the regions based on discontinuities of the grayscale properties [6]. Basic formulation of region-based segmentation is

$$(a) \bigcup_{i=1}^{n} (Ri = R) \tag{1}$$

(b) Ri is connected region, i=1, 2, …,n

(c) $Ri \cap Rj = \emptyset$ for all i = 1, 2, …n.

(d) P (Ri)=$TRUE$ for i=1, 2, …,n.

(e) $P(Ri \cup Rj)$=$FALSE$ for any adjacent region Ri and Rj.

$P(Ri)$ is a logical predicate defined over the points in set Ri and $\emptyset$ is the null set.

B. Canny Edge Detection Algorithm

This algorithm was used to detect the edge of the water on the captured image. Five processes are included on this algorithm, namely noise reduction, finding gradients, non-maximum suppression, hysteresis, and edge tracking.

The first step of the process is noise reduction. This is where the edges are smoothed out to precisely detect the edges of the content of the image. Gaussian filter is applied to make the image smooth. The kernel of a Gaussian filter with a standard deviation of σ = 1.4 as shown [7].

$$B = \frac{1}{159} \cdot \begin{matrix} 1 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{matrix} \tag{2}$$

The second step is called Finding gradients. This algorithm basically finds edges where the grayscale intensity of the image changes the most. These areas are found by determining gradients of the image [7]. Smoothened image is then filtered with a Sobel kernel [8] or Sobel-Operator [7] in both horizontal and vertical direction to get first derivative in horizontal direction (G_x) and vertical direction (G_y). From these two images, we can find edge gradient and direction for each pixel as

$$Edge\_Gradient\ (G) = \sqrt{G2x + G2y}$$

$$Angle\ (\theta) = \tan{-1}\left(\frac{Gy}{Gx}\right) \tag{3}$$

Gradient direction is always perpendicular to edges. It is rounded to one of four angles representing vertical, horizontal and two diagonal directions [8].

The third process is non-maximum suppression. Once the gradient magnitude and direction is determined, the image will go through a full scan to remove unwanted pixel that is not included as an edge. Each one of the pixels is checked if it is a local maximum in its neighborhood in the direction of the gradient [8].

The fourth process is called double thresholding or Hysteresis thresholding. This stage decides which are all edges are really edges and which are not. For this, we need two threshold values, minimum value and maximum value. Any edges with intensity gradient more than maximum value are sure to be edges and those below minimum value are sure to be non-edges, so it is discarded. Those who lie between these two thresholds are classified edges or non-edges based on their connectivity. If they are connected to "sure-edge" pixels, they are part of edges [8]. On the application, a value between 120-255 is used for the thresholding value of Canny Edge.

The last part of the process is Edge tracking. Strong edges are interpreted as "certain edges" and can immediately be included in the final edge image. Weak edges are included if and only if they are connected to strong edges. The logic is of course that noise and other small variations are unlikely to result in a strong edge (with proper adjustment of the threshold levels). Thus, strong edges will (almost) only be due to true edges in the original image. The weak edges can either be due to true edges or noise/color variations. The latter type will probably be distributed independently of edges on the entire image, and thus only a small amount will be located adjacent to strong edges. Weak edges due to true edges are much more likely to be connected directly to strong edges [8].

Edge tracking can be implemented by BLOB-analysis (Binary Large OBject). The edge pixels are divided into connected BLOB's using 8-connected neighborhood. BLOB's containing at least one strong edge pixel are then preserved, while another BLOB's are suppressed. [8]

IV. The Proposed System

This research is in two separate parts. The first part is to test the precision and accuracy of application given that the data sets are in a controlled environment or readily available on the Web. The second part is to test the application on the actual streets where there is flood water found. Currently, this is the first part of the research. In line with the methodology, Fig. 1 shows the process of the method done in this research.

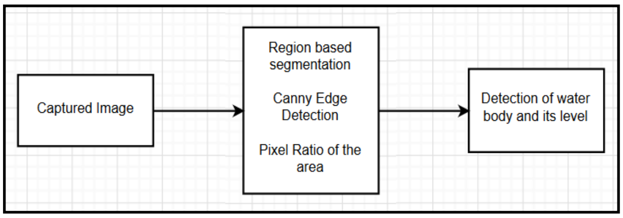


Figure 1. Input process output

Based on Fig. 1, the image was captured using CCTV camera. The captured image goes through image processing techniques, that is expected to give result to the detection of water and level of water found on the image.

A. *Initial Test*

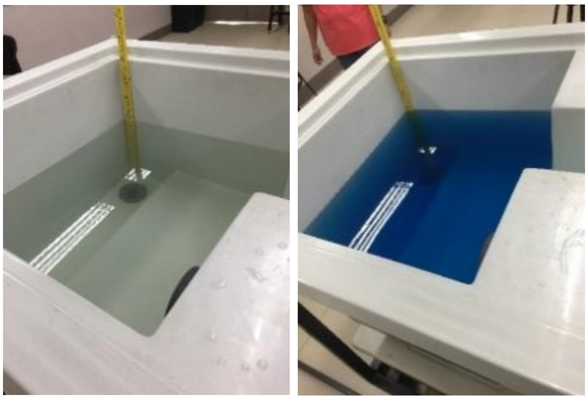
To test the algorithms stated, the application made was tested on a controlled environment that mimics the flood water.

1) Capturing the image

A Closed-Circuit Television (CCTV) captured image was used as an input to detect the body of water in an image. The capturing of the image was done at the Hydraulics room. A sample amount of water was placed on a HM150 Hydraulics Bench with a meter stick placed vertically inside the bench. The meterstick is measured in centimeters. The CCTV camera was attached on a tall board and was also connected to the network's infrastructure. The videos at pictures was were captured can be accessed on a computer found at the research area. See Fig. 2.

(a) (b)

Figure 2. Setup of CCTV in hydraulics room (a) and research area (b)



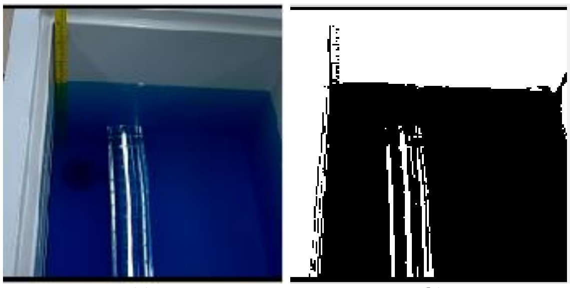
(a) (b)

Figure 3. Clear water setup (a) and dyed water setup (b)

The images taken from the CCTV was both clear water and dyed water to mimic the clear water and flooded water, respectively. Based on the actual measurement, the water's height is 35cm from the base of the hydraulics bench using the meter stick. Fig. 3 shows the level of the clear water (a) and dyed (b) taken using a mobile phone camera.

2) Detecting the body of water and determining the level

Once the image was captured, it goes through image processing techniques to find the water area on an image. Using Region-based Segmentation, the body of water was detected on the picture. Fig. 4 shows the image of dyed water (a) and its binarized result image (b). The area that is colored black indicates that the image has water in it. In order to detect the topmost edge of the water, canny-edge detection algorithm image processing technique was applied on the binarized image. Fig. 5 shows the step by step process of image from the original image to detection of the level of water.



(a) (b)

Figure 4. Image of dyed water (a) and its binarized equivalent (b)

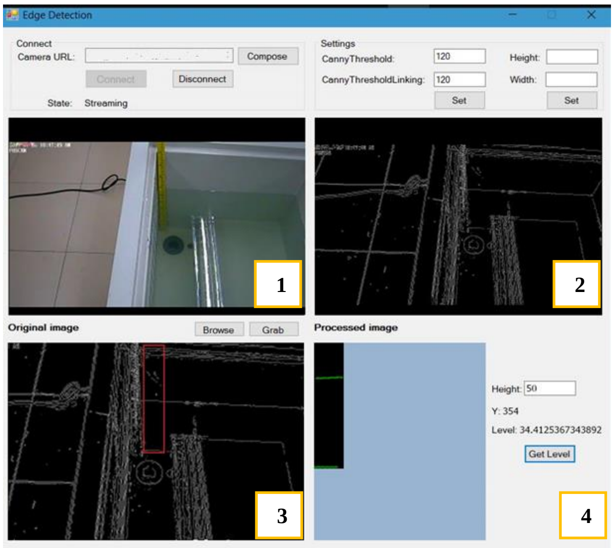


Figure 5. Processing of image from original image to detection of water and its level

The first window on the upper right is the actual CCTV footage or where the original image comes from. The second window, that is found at the upper right of the figure, is the processed image. In this window, it detects any edges it can find. The Canny Threshold and Canny Threshold Linking were tweaked to get the best possible detection of edges.

Once satisfied with the processed image, the image was grabbed by clicking on the ‘grab’ button and automatically appears on the third window. Using that window, the selection of the region to process should be where the water is found during detection. As you can see in the figure, the red vertical line indicates the selected region. The selection is from the top of the image up to the bottom of the hydraulics bench. The topmost edge of the rectangle is the topmost part of the selected region and the foot of the rectangle is the bottom of the selected region. The value of the height of the selected region should be inputted based on the actual measurement, as shown on the window. Thus, measuring the height of the area, where the viewing area of the CCTV, is required before hand.

Once the actual measurement is inputted, the program will calculate the level of water based on the height of the selected region. From the area of the region, the program will detect which line is crossing horizontally. This can be seen on the lower right corner window of the application (number 4 window). Using the Pixel ratio computation, the application calculates the number of pixels found in that region. Based on the values of the number of pixels and the height given, the application can compute what is the height of the water. The formula used in computing the height of the water is given by:

$$Height_{water} = \frac{no.\ of\ pixels}{Height_{given}} \left(\frac{y_1 - y_2}{2}\right) \qquad (4)$$

The value of y_1 and y_2 are both the value of the edges of the horizontal line. The Y value that is found on the application is the average value between y_1 and y_2. Using the formula, we can generate the height of the water through the selected region. The application should generate the measurement value that almost matches the value inputted from the actual pre-measurement.

V. Analysis and Results

Using 300 images as a dataset, which were taken from either a mobile phone’s camera and some images taken from the Web, the system was used to test the accuracy and precision of system. The data set consists of 150 images that contains water body, and 150 images that does not contain water body. Table I shows the confusion matrix used to test the precision of the system. The formula below was used to test the precision of the system

$$Accuracy = \frac{(TP+TN)}{N} \qquad (5)$$

$$Precision = \frac{TP}{Predicted\ water\ body\ (yes)} \qquad (6)$$

$$Misclassification\ Rate = \frac{(FP+FN)}{N} \qquad (7)$$

Aside from the confusion matrix table presented, the formula above was used to test the accuracy, misclassification rate and precision of the data set. True Positive (TP) shows that the picture used contains a body of water in it and was predicted by the system as positive. True Negative (TN) shows that the picture used does not contain water body in it and was predicted by the system as negative also. False Negative (FN) shows the number of pictures that has actual water body in it, but the system predicted it as negative. False Positive (FP) is the number of pictures that does not contain actual water body in it, but the system predicted it as positive. Based on the calculation, the system has achieved an 85% accuracy, 84% precision and 15% misclassification rate.

The same set of images was used to determine the level of water in an image. Only the images that are identified as True Positive where used to test the level of water. Table II shows the confusion matrix used to test the accuracy and precision of the system in detecting the height of the water.

TABLE I. Confusion Matrix of the Prediction of Water Body in an Image

N = 300	Predicted water body (YES)	Predicted water body (NO)	
Actual water body (YES)	TP = 130	FN = 20	150
Actual water body (NO)	FP = 25	TN = 125	150
	155	145	

TABLE II. CONFUSION MATRIX OF THE PREDICTION OF WATER LEVEL IN AN IMAGE

N = 130	Predicted water level (correct water level)	Predicted water level (incorrect water level)	
Actual water level (correct water level)	TP = 85	FN = 27	112
Actual water level (incorrect water level)	FP = 8	TN = 10	18
	93	37	

The same formula is used to get the accuracy, precision and misclassification rate as with Table I. Based on the confusion matrix in Table II and the formula above, the system has achieved an accuracy of 73%, a precision of 91% and a misclassification rate of 26%.

VI. CONCLUSION

With region-based segmentation, it helps highlight different segments found in the picture and thus the water segment was detected among other objects found in the picture. The water area was detected even if water is clear or dyed, but detection is solely dependent on the colors of the objects that is near the water. The closer the color is to the color of the water, the less chances of accurate segmentation. Detection of water is much more prominent if the color of the objects around it are darker than the color of the water.

The proponents made sure that when the images were captured, it was on a controlled environment wherein, the bin that was used to contain the water, may be disrupted or not, to imitate the behavior of water. Even though the water was disrupted, the algorithm can still detect the segments of the picture and thus still can determine the water area.

In detecting the top edge of the water, the Canny Edge detection algorithm was able to deliver the result. The algorithm itself is better compared to other edge detection algorithm because it shows concrete lines compared to others. Using the pixel ratio computation and with the help height value that was set on to the application, the pre-measurement of the area beforehand, and the 'y' value, it was able to compute the level of water.

When images are taken at nighttime, there should be enough lighting to detect the edges of the object especially the edge of the water.

In conclusion, the researchers have successfully created an application that was able to capture images from CCTV camera for image processing. The region-based algorithm used was able to determine the area of water on the picture. The canny edge detection and pixel ratio computation was able to detect the edge of the water based on a selected region where the water is found, and the measurement of the actual height of the water, closely matches the height stated in the application.

Thus, the researchers have found a way to detect a body of water and gauge its water level using image processing techniques, that is much more affordable compared to existing tools or machines.

VII. RECOMMENDATION

The researchers recommend that the application and the algorithm be tested on the actual streets where flooded water is found. The CCTV should be placed on an area wherein its viewing angle is parallel or the like, to flooded street to maximize accuracy of the result. An algorithm is needed to compensate the detection of water when the image is blurred or taken during nighttime is needing to increase the accuracy of the result too.

This study is also recommended for researchers who finding solutions to detect flood in areas that are remote using CCTV camera, or any researchers that looks for ways to detect water without human intervention.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

All the authors were involved in the development of this study. The third author wrote the program that implements the discussed techniques; all authors had approved the final version.

ACKNOWLEDGMENT

The authors would like to thank all their colleagues who provided support throughout this study and to FEU Institute of Technology for providing development resources.

REFERENCES

[1] M. J. Manashti and M. A. Azimi, "Water level measurement using image processing," presented at the Second International Conference on Agriculture and Natural Resources, Iraq, December 25-26, 2013.

[2] R. C. Gonzalez and R. E. Woods. (2007). Region growing. [Online]. Available: https://en.wikipedia.org/wiki/Region_growing#Region-based_segmentation

[3] S. K. Subraniam, V. R. Gannapathy, S. Subramonian, and A. H. Hamidon, "Flood level indicator and risk warning system for remote location monitoring using flood observatory system," *WSEAS Transactions on Systems and Control*, vol. 5, no. 3, March 2010.

[4] C. R. Kothari, *Research Methodology Methods and Techniques*, 2nd revised edition, New Delhi: New Age International (P) Limited, 2004, ch. 6, p. 96.

[5] S. Park, N. Lee, Y. Han, and H. Hahn, "The water level detection algorithm using accumulated histogram with bass pass filter," *World Academy of Science, Engineering and Technology*, vol. 3, no. 8, 2009.

[6] S. Jatmiko, A. B. Mutiara, and M. Indriati, "Prototype of water level detection system with wireless," *Journal of Theoretical and Applied Information Technology*, vol. 37, no. 1, 2012.

[7] E. Basha and D. Rus, "Design of early warning flood detection systems for developing countries," master thesis, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, 2014.

[8] P. K. Kalra. (March 2009). Canny edge detection. [Online]. Available: http://www.cse.iitd.ernet.in/~pkalra/col783/canny.pdf

Beau Gray M. Habal was born in Bacolod City, Philippines, on December 17, 1983. She is a graduate of Bachelor of Science in Computer Science at the University of St. La Salle-Bacolod in the year 2004 and has graduated as a Master of Science in Computer Science at the University of Negros Occidental – Recoletos in the year 2006. She is currently studying her Doctorate degree in Information Technology at the University of the East – Manila. She is currently working at Far Eastern University – Institute of Technology, Manila, Philippines as an Associate Professor. She has presented a few researches that is currently under proceedings.
Ms. Habal member of the Philippine Society of Information Technology Educators.

Elisa V. Malasaga was born in Valenzuela City, Philippines, on October 29, 1970. She received the Bachelor of Science in Information Science last 1991 and Bachelor of Science in Computer Science in the year 1992 both at AMA University and has graduated last 2010 with Master’s degree as a Master of Science in Computer Science at AMA University – Main Campus, Quezon City. She passed certification from IBM – IBM Certified RAD 7 and IBM Certified DB2 last 2010. She is currently employed at Far Eastern University – Institute of Technology, Manila, Philippines as Director, CS Department. She has presented and published a few researches under her name.

Abraham T. Magpantay was born in Makati City, Philippines, on September 5, 1991. He graduated as Cum Laude from FEU Institute of Technology (Formerly FEU-East Asia College) with a Bachelor of Science in Computer Science with specialization in Software Engineering in the year 2015. Currently he is pursuing his graduate studies in Master of Science in Computer Science at the Ateneo de Manila University. He is currently working as a fulltime Computer Science instructor from his Alma matter. He is a professional member of the Association for Computer Machinery (ACM) and also, the adviser of ACM Student Chapter at the FEU Institute of Technology.