Towards Smart Mobility During Flooding Events in Urban Areas using Crowdsourced Information

Ubaldo Castro
Department of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
ucastro@miners.utep.edu

Jonathan Avila
Department of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
jeavila6@miners.utep.edu

Cynthia V. Sustaita
Department of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
cvsustaita@miners.utep.edu

M. Armando Hernandez
Department of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
mahernandez46@miners.utep.edu

Victor M. Larios
Smart Cities Innovation Center
CUCEA Universidad de Guadalajara
Guadalajara, Jalisco, Mexico
victor.m.lariosrosillo@ieee.org

Natalia Villanueva-Rosales
Department of Computer Science
Cyber-ShARE
The University of Texas at El Paso
El Paso, Texas, USA
nvillanuevarosales@utep.edu

Oscar Mondragon
Industrial, Manufacturing &
Systems Engineering Department
The University of Texas at El Paso
El Paso, Texas, USA
oamondragon@utep.edu

Ruey Long Cheu
Department of Civil Engineering
The University of Texas at El Paso
El Paso, Texas, USA
rcheu@utep.edu

Rocio Maciel
Smart Cities Innovation Center
CUCEA Universidad de Guadalajara
Guadalajara, Jalisco, Mexico
ma.maciel@academicos.udg.mx

***Abstract*—Flooding hinders the ability of residents to move through-out urban areas. Current weather information systems provide weather forecasts with a granularity too coarse to inform city residents of floods at a street level. Reliable real-time data is also limited in some cities and only a few systems enable users to report flooding events. This research project aims to inform residents about flooding events through the Flooding Alert System (FAS), which seamlessly integrates real-time data, crowdsourced data, historical data and provides timely and precise information to users so that they can make informed decisions during these events. The proposed Smart Cities solution, uses Internet of Things sensors equipped with an image processing algorithm for approximating the water level at at-risk areas. In adittion, a mobile application allows users to report and receive alerts of nearby flooding events as they occur. The expected products will be piloted in Guadalajara, JAL, MX. Transferability of our solution is planned to be evaluated through the application of these products in El Paso, TX, USA.**

***Index Terms*—Smart City, Crowdsourcing, Transportation & Mobility, Internet of Things, Safety, Smart Infrastructure, Flooding Algorithms**

I. Introduction

Flooding is a natural disaster that affects millions of people every year. A report by United Nations reveals that flooding alone accounted for approximately 47 percent of all weather disasters worldwide from 1995 to 2015 [1]. Flooded streets affect the mobility and safety of resident in urban areas. Residents must change their daily routine and failing to evacuate flooded areas can result in their injury or death. Flooding also has major impacts in mobility of cities causing stranded vehicles in traffic for several hours and making complete neighborhoods areas inaccessible. Furthermore, flooding can damage a city's infrastructure and the financial implications could be significantly negative for the communities involved by decreasing business sales and interrupting private and government services.

As part of the NSF-funded IRES: US-Mexico Interdisciplinary Research Collaboration for Smart Cities, The University of Texas at El Paso (UTEP) and The University of Guadalajara (UdeG) are working towards creating relevant Smart Cities solutions. This work aligns to the following Smart Cities definition: "A smart city is characterized by its ability to integrate people, technology, and information to create a sustainable and resilient infrastructure that provides high-quality services for residents" [2]. This paper describes the development of a Flood Alert System (FAS) that aims to improve mobility and safety of residents in urban areas affected by flooding, thus improving resident's quality of life.

We postulate that by providing timely and adequate information to residents before flooding events, residents can take preventative measures when traveling during heavy rainfall. The objective for this work is to develop an user-friendly

This work was supported by the IRES:US-Mexico Interdisciplinary Research Collaboration program funded by the National Science Foundation (NSF) Award #1658733. This work used resources from Cyber-ShARE Center of Excellence, which is supported by NSF Award #HRD-0734825.

technology to inform of nearby flooding events as soon as reliable information is available. The FAS technical infrastructure is compose by Internet of Things (IoT) sensors that will be strategically placed across urban areas to collect real-time data about water levels. FAS also enables the crowdsourcing of data including pictures and individual perceptions of the severity of the flooding event. FAS mobile application alerts users when flooding events occurs in their nearby area. The proposed Smart City solutions integrates a human and technological infrastructure to facilitates mobility and foster safety of residents during flooding events.

The remaining of the paper will focus on urban flooding and the development of FAS. Section II reviews related work on flood information systems that have been implemented and their purpose. Section III explains the methodology up to the development of the first prototype for this work. Section IV describes the algorithm used to approximate the water level from an image entry. Section V discusses the results obtained from the current work and future tasks and objectives. Section VI describes the conclusions obtained from this research.

II. Related work

Current flooding information systems provide information with specific purposes. WaterWatch, developed by U.S. Geological Survey (USGS), is a website that displays real-time streamflow conditions using maps based on over 3,000 long-term USGS stream gauges [3]. FloodWatch, developed by D5G Technology, LLC, is a mobile application that monitors rivers and streams, leveraging real-time stream gauge data from USGS and National Weather Service [4]. Flood, developed by American Red Cross, is a mobile application that offers an alert feed and push notifications on flood events and a map with weather-related layers [5]. These maps are static and do not allow navigation.

Flooding information systems have traditionally used water gauges to determine the water level in an area. Water gauges work well when placed in a river or stream, however they can be easily damaged or stolen when placed in a populated city. More recent solutions aim to provide timely floods detection systems for Smart Cities. A research team from University of Franca in Brazil has built a flood warning system that uses ultrasonic sensors placed above a body of water to calculate water level by measuring the distance from the waters surface to the sensor [6]. The system would alert authorities and residents through telephone calls and SMS messages about risky events during the rainy season. Another research team from the University of Tenaga Nasional (UNITEN) in Malaysia has built Flood Management Emer-gency Response (FMER) system, an application for simulating the occurrence of floods [7]. The proposed FAS framework, is an early flood detection system to alert communities during flood events.

FAS will approximate the water level in an area and measure with a granularity that is fine enough for residents to use when considering traveling in their city. FAS uses crowdsourced data via user reports to increase the accuracy and reliability of historical data, useful for both residents and authorities.

III. Methodology

The following subsections describe the work towards the creation of the FAS system which is currently in the prototyping phase.

A. Testbed Location

Historically, the cities of El Paso and Guadalajara have been prone to flooding. The main cause for flooding in these cities are storm-water systems incapable of handling large amounts of water due to heavy rains. Solid waste carried by the water prevents the immediate flow of water. Both cities have experienced damages and losses due to heavy rains. El Paso experienced torrential rains during the summer of 2006 causing an overflow of the Rio Grande [8] with approximate damages of $450 million in damages. In Guadalajara, one of most flood affected zones is the Arcos del Milenio underpass, where water level reached a height of 1.5 meters and left 83 vehicles stranded in 2015 and multiple injuries. This situation repeats every year [9].

Current experimental tesbed for verification and validation of the Smart City solution proposed is located at UdeG CUCEA campus in Guadalajara, MX. Firsts testing prototypes for data collection have been placed in prone to flood zones within the campus.

B. Flood Alert System

Components of the proposed system architecture are shown in Fig. 1. Cameras will be installed in strategic locations to create an IoT sensor network throughout the city. Locations will be selected by analyzing historical data to find the areas that are most prone to flooding. As part of our IoT design, a Raspberry Pi was equipped with a No InfraRed (NoIR) camera and a Wi-Fi module. Raspberry Pi was chosen because it is a small, readily available computer capable of running a Linux operative system, suitable for carrying out software development tasks such executing computer vision algorithms. It is important to highlight that the current prototype runs the image processing algorithm locally on the Raspberry Pi. In the future, it is planned to run it in a remote server as a back-end service. The Raspbian operating system was installed as it has been optimized to run on the Raspberry Pi hardware and comes with precompiled software version for easy configuration and use.

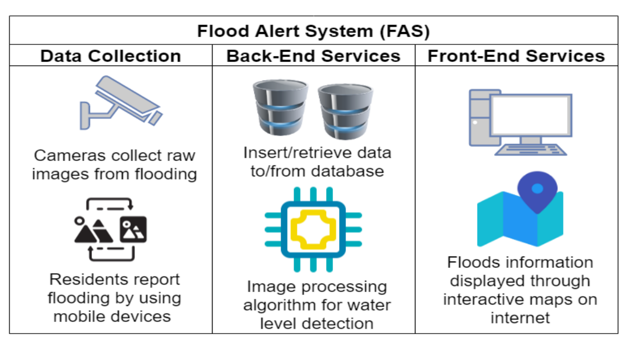

Fig. 1. Components of the FAS prototype.

As shown in Fig. 2, a series of reflective tapes were placed on a nearby surface visible by the camera. It is intended that these reflective tapes cover a height from 0 to up to 1.5 meters since floods in the testbed location have been reported to reach this height as stated in section III-A. Spaced out distance between each tape mark may help increase the FAS water level detection and consequently, obtain a better approximation for the water level severity. FAS selected a reflective material as water level marker since is designed to be highly visible in both day and night and facilitate object detection from the background. FAS captures a constant video stream input and applies image processing algorithm techniques to obtain flooding severity results in real time.

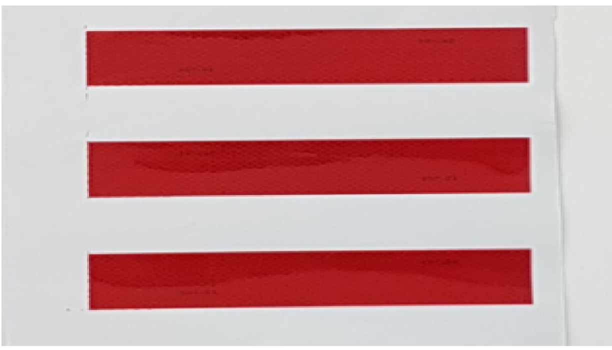

Fig. 2. Reflective tape used for the water level markers.

C. Data Model

Each sensor in the network will periodically process image captures on-site, approximate the water level in the area, and upload the water level to a database. A mobile application will read the data from the database. Fig. 3 shows the specific data fields that are stored in the database: the location of the sensor, the severity of the flooding event, and a timestamp of the report. This data will be used to inform users of current flooding events through maps on the FAS mobile application, including the exact location and severity of the flooding depending on the water level.

```
_id: ObjectId("5d51b59cde13e935041d0da8")
severity: "Medium"
description: "Test"
location: Object
    lat: 31.766
    lon: -106.5022
date: "2019-08-12 12:53:16"
imageName: "1565635971226.jpg"
```

Fig. 3. A sample of a document stored in our database

D. Use Case Diagram

A use case diagram is a high level abstraction used to document the functionality provided by a system and to indicate which external entities are permitted to use which functionality [10]. Fig. 4 shows FAS's services (known as use cases) and the actors that interact with each service.

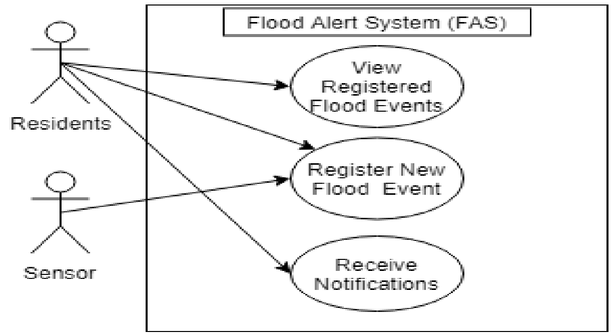

Fig. 4. Flood Alert System's use case diagram

FAS currently has two actors: residents and sensors. An actor is an external entity that represents a role played by the individuals or devices interacting with the system. Each actor has a relation with one or more use cases, which are the services the systems provides (displayed as ovals). The relation represents interactions between an actor and the system. The first actor represents residents from Guadalajara JAL, and El Paso, TX, and the second actor represents a device that will contain an image processing algorithm in order to analyze flooding behavior.

The sensors in FAS constantly collect data, analyze it and upload it to the database in real-time. Residents will act like 'sensors' by providing flooding data as described in the following subsection.

The current functional FAS prototype allows users to view floods events though a website that displays an interactive map with markers placed in the location of such events (Fig. 5). Depending of the severity, the color of the circle within the marker changes: green for low severity, orange for medium severity and red for a high severity. Users need a device with Internet access to display the map. If a marker is clicked by the user, a window appears on top of the map showing the report information as address, severity, date, description and the report image.

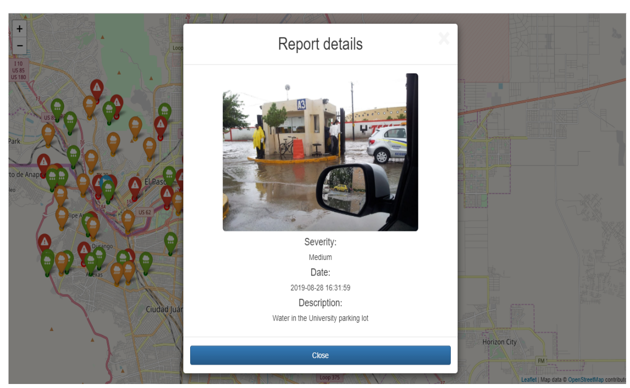

Fig. 5. Map displaying a report for a flooding event.

E. FAS Mobile Application

FAS includes a mobile application to provide residents with the ability of accessing or contributing with flooding data. The FAS App mobile application displays a feed of recent flooding events that includes an image, location, date-time, and the severity of the flooding (Fig. 6). FAS will send notifications to users who are located nearby flooding events or have a route that covers a flooded area. Users will be provided with information about past incidents in nearby flooded areas to foster their responsive actions to ensure their safety.

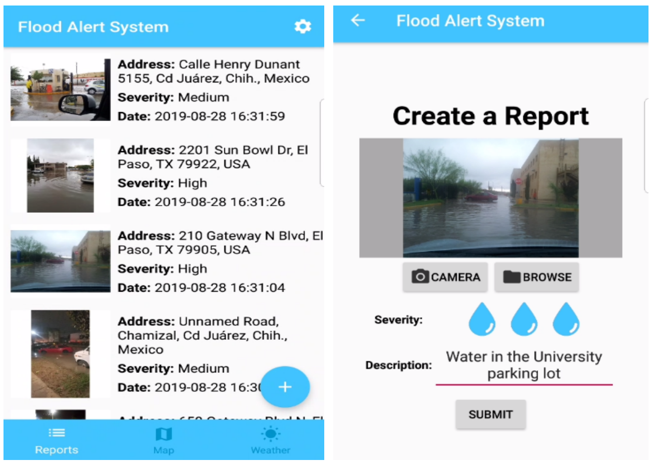

Fig. 6. FAS app showing last reported flooding events (left) and user created reports (right).

FAS App is currently under development using the Flutter framework. Flutter is a UI toolkit developed by Google for creating native applications for the Android and iOS operating systems using a single code base [11]. This work is part of a bi-national research collaboration between UTEP and UdeG, thus multilingual support have been added for including translations in both English and Spanish languages depending on user mobile device default language.

F. Smart Cities Features

FAS development strategy adheres to the five principles of Smart cities solutions: scalability, interoperability, modularity, resiliency, and security [12].

Scalability refers to the performance of the system when the number of users and data being collected and transferred grows. As more people find value in the system's services, more people will want to use it. A city has thousands of potential users, creating the need for a robust back-end services that can handle many requests to resources. The entire back-end of the system is based on Googles Firebase, which can handle thousands of concurrent users and is thus used by large applications like Shazam and Todoist [13].

Interoperability refers to how well the system can work with other systems. Our system uses a variety of technologies, so it is important that information can be exchanged within our system and among other systems if needed. For example, a navigation application may want to display flooding events as part of their cautions and reroute drivers. To do this, they would only need to communicate with the database, as the sensors and application are not required to access the data. FAS has been designed using best practices and current technology for IoT which increases the ability of the system to interoperate with similar systems and/or exchange data.

Modularity refers to how the system is broken down into specialize components and how these components are inter-related. A good system modularity enable the replacement of one of the components without affecting the functionality of the whole system. FAS contains a single database, which the sensors write to, and an application that writes to the database when creating reports and reads from the database when displaying events. The sensors software and the FAS mobile application can be modified independently, without affecting each other. Using the Flutter toolkit, only one change has to be done to a single source code that address Android and IOS operative systems.

Resiliency refers to how well the system can recover from failure. Data fed into the system comes from two sources: sensors and users of the mobile application. Both type of data collection means are independent and large in size. Our design supports the failure of sensors and the remote management of sensors (sensors can be controlled via remote connection to update code). Similarly, FAS will check whether there is or not WiFi connection in order to decide between local data storage or sending data for external database storage through the back-end services. Finally, the application is intended to be distributed through the app stores of their respective OS, so updates can be pushed there.

Security refers to the measure of the system to prevent unauthorized access. To address the security aspect in FAS, a procedure to authenticate users will be put in place so that only legitimate users are allowed to read and write into the database. This will prevent hackers to access the information that they are not allowed to see or set false alarms that could cause panic.

IV. Flooding Water Level Detection Algorithm

A flooding water level detection algorithm consisting of image processing stages was developed as shown in Fig. 7.

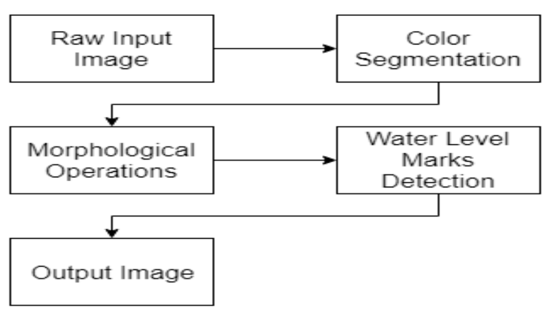

Fig. 7. Water level detection algorithm diagram.

FAS has an advantage over other systems that uses ultra-sonic sensors as the current system is not affected by factors as the temperature and humidity of the air, as well as objects that can absorb wave sounds and therefore affect the accuracy of the results obtained. Additionally, visual feedback provided by cameras allow improvement of the current solution by implementing new data extraction algorithms. A raw input image collected using FAS is shown in Fig. 8, although it is important to remark that FAS works as a real time system which senses and processes a constant video stream.

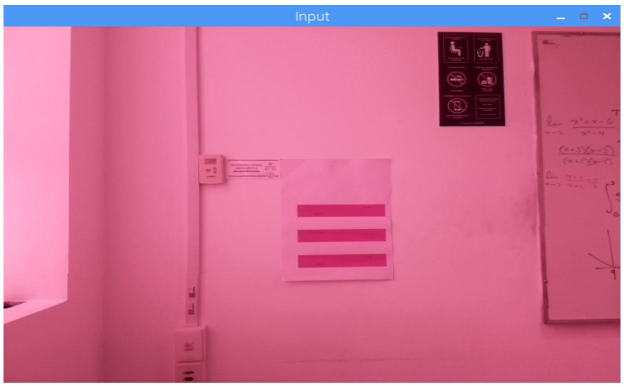

Fig. 8. Raw input image sample.

FAS algorithm's first stage is a color segmentation in which an input image is transformed from the RGB (red, green, blue) color model to the HSV (hue, saturation, and value) color model. The color transformation is done since the RGB color model mixes color (chrominance) and intensity (luminance) information and has non uniform characteristics [14] which complicates the color segmentation process. Color segmentation using the HSV color model is simple since color information is contained within a single dimension (hue dimension). After the color transformation, a threshold mask is applied to determine the minimum and maximum threshold value for hue, saturation, and value. Any pixel value that lays within this minimum and maximum threshold acquires a value equal to 1, otherwise a value equal to 0. Fig. 9 shows the results obtained after the first image processing algorithm stage.

Fig. 9. Color segmentation processing algorithm stage.

The second stage in the algorithm performs morphological operations based on the binary image obtained from the first stage to smooth the contours of the image and reduce the noise in the image. In particular, the algorithm uses opening which is erosion followed by dilation [15]. Fig. 10 shows the results obtained after the algorithms second stage.

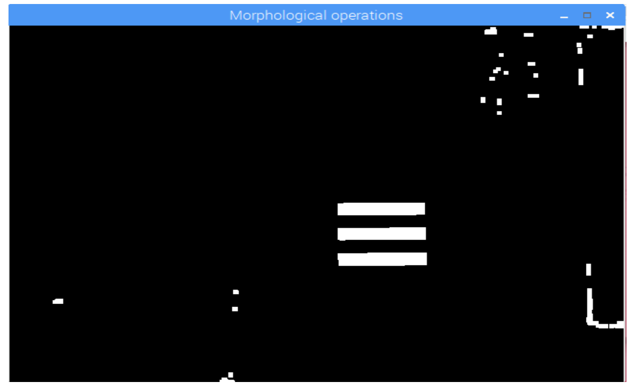

Fig. 10. Morphological operations processing algorithm stage.

The algorithm's third stage uses shape detection to detect the water level markers, which have a rectangular shape. At this point, some of the objects masked could be erroneously identified as water level markers. To help avoid this problem, we use another contour property called aspect ratio, which is the ratio between the width and the height of a bounding rectangle of the object [16]. The aspect ratio for each detected object within the masked image is calculated and is used to more accurately identify the water level markers. Since aspect ratios are prone to vary depending on external factors, such as illumination or motion, a +/- 10% tolerance is applied to the aspect ratio constant value. Fig. 11 shows the results obtained after the algorithm's third stage. In order to reduce the number of false positives during shape detection, 'x' center position of each rectangle is used as a parameter for distinguish between water level marks and possible rectangle shape objects that might appear on the image background. This design decision is based on the fact that water level marks are placed in stack position and that these marks should be placed within a similar interval around an 'x' center value position.

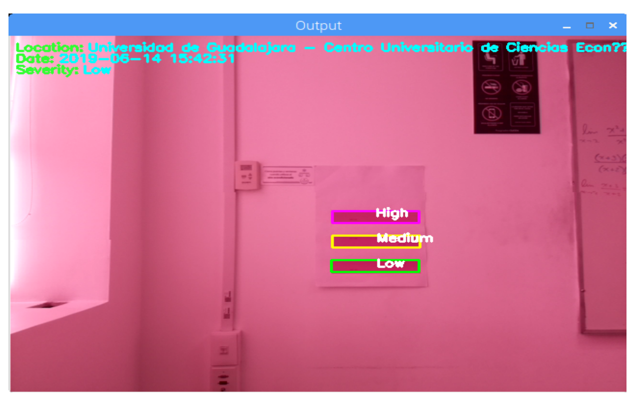

Fig. 11. Water Level marks detection processing algorithm stage.

V. Discussions and Future Work

Developing inclusive Smart Cities solutions should take into account the digital divide, which whether economical, geographical or generational is a challenge that needs to be addressed for improving services accessibility for users. The current version of the FAS system collects data from two different sources, crowdsourcing and IoT devices. Input data collected from crowdsourcing will be limited to users with mobile devices that have the capability to install the mobile app and make use of WiFi/mobile data for downloading flooding reports. Part of the future work evaluates the inclusion of SMS messaging services for receiving and sending flooding events info to users that do not have access to data usage. In the other hand, to improve IoT devices resilience, a wireless connectivity card can be included in the IoT device to send low frequency messages in case internet service falls during a flooding incident. Additionally, it is also contemplated to include a portable battery by using a power board for uninterruptible continuous power supply in power-shortage events.

At the time of submitting this paper, the image processing algorithm is being evaluated, both in a controlled environment and outdoors where noise is introduced to images. It is aimed to validate the algorithm by analyzing the flooding water level on streets by using marks as references for flooding level/severity. Once the algorithm is validated, a further step to scale will be to use the surveillance cameras already mounted in the tesbed city (Guadalajara is approx. 8000 at the moment) by collaborating with city authorities. One of the main challenges when applying crowdsourcing is related to the veracity of the information collected. One approach proposed for future work is to use data collected from the image processing algorithm to verify the veracity of the information collected from users through crowdsourcing. The mobile application will be evaluated by presenting a single question to users when displaying flooding reports made by other users. By asking users if they find reports helpful, we can identify how well the user reports are performing.

Other tasks will include data analysis. By storing the timestamp, flood location of the events along with flood severity levels, it is proposed to develop a predictive model to warn users before flooding occurs. The model would be trained by using image captures from historical data, including the change in flooding water level over time in a single event.

VI. Conclusion

Current relevant initiatives have focused on either crowdsourcing or image processing to develop flooding alert systems. The proposed FAS combines these two approaches to collect flooding data that is both accurate and reliable, with the overarching goal of improving the quality of life for residents of El Paso and Guadalajara. Informing residents about current flooding risks may assist them in mobility through at-risk areas, increasing safety and reducing damage to the city infrastructure. FAS makes possible to estimate the water level in an at-risk area with the usage of images and data captured by a user through a mobile application and an image processing algorithm. Crowdsourcing via user reports assists our cause in notifying users about the flooded zones and how severe the case is. The presented work contributes to providing insights in the use of IoT devices along with crowdsourcing models for users in urban areas at risk of natural disasters.

VII. Acknowledgements

The authors would like to thank members of the previous cohort of the IRES: US-Mexico Interdisciplinary Collaboration for Smart Cities. In particular, Dania Orozco and Elias Castillo for their initial conceptualization of the project, Sebastian Fuentes for helping with the development of the first prototype, as well as members of the Universidad de Guadalajara including Dr. Jesus Raul Beltran Ramirez, Dr. Jose Antonio Orizaga Trejo and Emmanuel Aceves for their insightful feedback.

References

[1] UNISDR and CRED, “The Human Cost of Weather-Related Disasters 1995-2015,” p. 30.

[2] N. Villanueva-Rosales, R. L. Cheu, A. Gates, N. Rivera, O. Mondragon, S. C. C. Ferregut, C. Carrasco, S. Nazarian, H. Taboada, V. M. Larios, L. Barbosa-Santillan, M. Svitek, O. Pribyl, T. Horak, and D. Prochazkova, “A Collaborative, Interdisciplinary Initiative for a Smart Cities Innovation Network,” in *2015 IEEE First International Smart Cities Conference (ISC2)*. Guadalajara, Mexico: IEEE, Oct. 2015, pp. 1–2.

[3] “USGS WaterWatch – Streamflow conditions,” https://waterwatch.usgs.gov/.

[4] D. T. LLC, “FloodWatch,” https://www.floodwatchapp.com/.

[5] “Mobile Apps,” https://www.redcross.org/get-help/how-to-prepare-for-emergencies/mobile-apps.html.

[6] A. Silva Souza, A. M. de Lima Curvello, F. L. d. S. de Souza, and H. J. da Silva, “A flood warning system to critical region,” *Procedia Computer Science*, vol. 109, pp. 1104–1109, 2017.

[7] A. Yusoff, I. S. Mustafa, S. Yussof, and N. M. Din, “Green cloud platform for flood early detection warning system in smart city,” in *2015 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW)*. Riyadh, Saudi Arabia: IEEE, Feb. 2015, pp. 1–6.

[8] J. Rogash, M. Hardiman, D. Novlan, T. Brice, and V. MacBlain, “Meteorological Aspects of the 2006 El Paso Texas Metropolitan Area Floods,” p. 25, 2006.

[9] “Un muerto y 60 lesionados deja tormenta en Guadalajara,” https://www.excelsior.com.mx/nacional/2015/06/12/1029130.

[10] M. Seidl, M. Brandsteidl, C. Huemer, and G. Kappel, Eds., *UML @ Classroom: Eine Einführung in Die Objektorientierte Modellierung*, 1st ed. Heidelberg: dpunkt-Verl, 2012, oCLC: 806979979.

[11] “Flutter - Beautiful native apps in record time,” https://flutter.dev/.

[12] V. M. Larios, L. Gomez, O. B. Mora, R. Maciel, and N. Villanueva-Rosales, “Living labs for smart cities: A use case in Guadalajara city to foster innovation and develop citizen-centered solutions,” in *2016 IEEE International Smart Cities Conference (ISC2)*. Trento, Italy: IEEE, Sep. 2016, pp. 1–6.

[13] “Firebase,” https://firebase.google.com/use-cases.

[14] K. B. Shaik, P. Ganesan, V. Kalist, B. Sathish, and J. M. M. Jenitha, “Comparative Study of Skin Color Detection and Segmentation in HSV and YCbCr Color Space,” *Procedia Computer Science*, vol. 57, pp. 41–48, 2015.

[15] Zhao Yu-qian, Gui Wei-hua, Chen Zhen-cheng, Tang Jing-tian, and Li Ling-yun, “Medical Images Edge Detection Based on Mathematical Morphology,” in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*. Shanghai, China: IEEE, 2005, pp. 6492–6495.

[16] “OpenCV: Contour Properties,” https://docs.opencv.org/trunk/d1/d32/tutorial_py_contour_properties.html.