Design and Deployment of a Flash Flood Monitoring IoT: Challenges and Opportunities

Bipendra Basnyat
UMBC
Maryland, USA
b125@umbc.edu

Neha Singh
UMBC
Maryland, USA
nehasingh1@umbc.edu

Nirmalya Roy
UMBC
Maryland, USA
nroy@umbc.edu

Aryya Gangopadhyay
UMBC
Maryland, USA
gangopad@umbc.edu

***Abstract*—Successful implementation of the Internet of Thing (IoT) is precursory to a thriving smart city. However, the technical, physical, and environmental conditions can often pose challenges in their successful deployments. The deployment is further complicated if the time and location of implementation are amidst a natural disaster.**

In this work, we use flash flood detection as a natural hazard testbed and describe various IoT deployment, our progression, and first-hand experience from those implementations. We compare and contrast three IoTs and their performance in real-time execution. Next, we discuss systems architecture and their end-to-end design and present lessons learned from these heterogeneous deployments. Additionally, we evaluate and outline our observations, challenges, and opportunities for further improvement. We also formulate standard evaluation metrics for their scoring and document our deployment journey.

***Index Terms*—IoT, Edge and Cloud computing, Smart City, Smart Services and Computing, Flood Detection**

I. Introduction

The vision of smart cities encompasses a safe, clean, healthy, inclusive, resilient environment that can provide economic opportunity and high quality of life for their residents. When the daily chores of city inhabitants are fully integrated into successful Information and Communication Technology (ICT), one can say that Smart living is achieved. Smart cities and ICT address the quality of life by improving the health, safety, and security of their citizens [1], [2].

The concept of smart living becomes more valuable when a community is dealing with a crisis and unforeseeable situations. A city is indeed smart if it succeeds in helping the community navigate through the information overload during such emergencies. Smart IoTs can be those agents in the time of crisis. Thus, IoTs and ICT systems play a vital role in disaster management in smart cities. They enable effective communication between the first respondents, government agencies, and local inhabitants.

Internet of thing (IoT)s and ICT systems have been well-integrated in the transportation industry, mainly due to the advancements of self-driving vehicles. On the other hand, water resource management and specifically the flash flood monitoring is one of the areas within civil engineering that has lagged ICT and IoT integration. Water-induced disasters such as floods, storms, heavy rainfall, etc. are some of the most dangerous and devastating forms of calamities. Flash flood is a common occurrence and can happen on tranquil streams and creeks in the neighborhood, city streets, and highway underpasses [3]. Motivated by this, we designed various early and intermediate stage prototypes to mitigate the problem. Over the last two years, we have developed multiple prototypical systems that allow us to gather real-time data and empower decision-makers to act in advance to the disaster. Our prototype can remotely monitor/sense the flooding situation and deliver a functional cyber-physical system. We also believe that our systems can be used to create a community watchdog via social media integration in a smart city. In this work, we present our findings from the flood detection and monitoring systems, deployed in Ellicott City, Maryland, USA.

A. Objective

This work's main objective is to understand and document the complexity and challenges involved in the deployment process of a disaster risk reduction IoT systems. The underlying tasks in attaining this objective are:

- Design and deploy smart IoT systems for early warning and enhancement in the decision support system.
- Assess and validate the design, implementation, and deployment of IoT systems under harsh conditions and heterogeneity.
- Stress-test the IoT systems for their reliability during disastrous/extreme situations.

B. Contributions

The main contributions of this paper are as follows:

- This paper presents a detailed architecture and design approach used in three different kinds of flash flood detection IoTs.
- We deploy our IoTs in real-world hazardous conditions and experiment with their robustness in a heterogeneous environment.
- We frame each deployment into its generation to evaluate its performance based on the underlying technology.
- We rationalize our progression from one deployment/generation to next by documenting the actionable learning and shortcoming.

II. Related Work

Ashton [4] introduced the term Internet of Things (IoT), and it has been one of the most revolutionary technologies of the

978-1-7281-6997-2/20/$31.00 ©2020 IEEE
DOI 10.1109/SMARTCOMP50058.2020.00088

21st century. In recent times, IoT has become a common and integral part of everyone's daily lives. It has improved people's lifestyle in several domains through millions of sensors and devices which perform various tasks such as to measure, collect, store, process and transfer huge amounts of data everyday [5], [6].

The research [7] reviews various innovative applications in multiple domains such as health-care, smart buildings, smart cities, transport, agriculture mining from the utility perspective in the area of IoT and data analytics. The research [7] reviews various innovative applications in multiple domains along with the challenges they encountered while executing the whole process (development and deployment) from the utility perspective in IoT and data analytics. It is quite prominent based on the survey [7] that these applications will continue to improve our lives. However, there are some serious challenges, and researchers are working towards addressing these challenges and making it more effective and efficient. The difficulties in deployment have been well studied and can be categorized into the following major areas [7], [8].

Performance & Scalability: It is caused by the need for infrastructure scaling and real-time, fast analytic processing (accurate predictive measures), especially during disasters.

Heterogeneity in Data Sources: As we integrate multiple IoTs and sensors, we start to assimilate different types of data, for example, social media contents, audio-visual elements, satellite, and geospatial data. Heterogeneity in the data set increases the possibility of richer knowledge and insights but also deepens the complexity of the process.

Time-Space Complexity Computing: Another challenging critical component is the velocity and veracity of real-time IoT data collection, processing, and its overall management.

Device Reliability: Faulty sensors in the devices sending malicious or missing data can mislead or make the machine learning model bias resulting in less optimal results.

Individual privacy concerns: As connected IoTs integrate into people's lives as a society, the individual's fundamental privacy could be compromised. It is a challenging task to maintain the balance of privacy concerns while providing the appropriate individualistic solution of daily smart living.

III. General System Architecture

All our deployments adhere to similar technology and architecture. They follow a three-layer IoT architecture, which contains a Perception layer, a Network/Gateway layer, and an Application layer. These layers are shown in Figure 1. The communication protocols, IoTs units, target infrastructure, are different in every generation (Gen) deployment, but their overall data flow and working are the same. The target structure for Gen-1, Gen-2, and Gen-3 deployments are on-premise server, edge computing unit, and the cloud computing infrastructure, respectively.

IV. Deployment Progression

We describe these IoT's and their echo system as generations of their own. We then describe and present how we leaped into next-generation using the previous deployment as

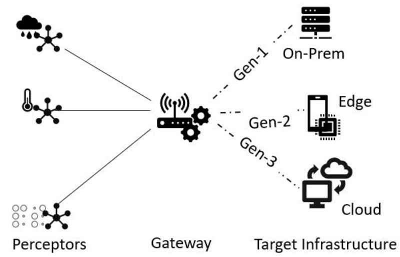


Fig. 1. General System Architecture

our stepping stone. As we progress into the next generation, we also experiment with different technology and underlying computing units such as on-premise server-based computing, edge computing, and cloud computing. The progression and primary contributing technology used in each of these deployments are summarized in Figure 2.

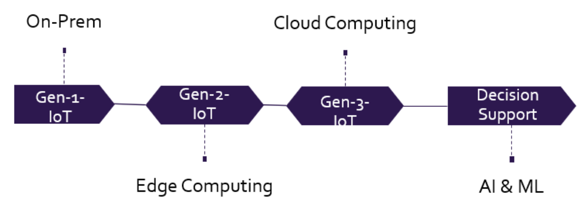


Fig. 2. Deployment & Progression Plan

5

A. Gen-1 On-Prem IoT

Our first IoT deployment is based on "On-Premise IoT Solutions, called Smart Security P&S Unit. We purchased and acquired the sensor along with the CPU from a European company named Libelium. The Waspmote Plug & Sense contains an internal SD (Secure Digital) card with up to 2 GB storage. The battery in the unit is charged through the solar panel.

(a) IoT Description (b) Field Deployment

Fig. 3. Gen-1 Deployment

1) Method and Materials: Figure 3(a) explains the functionality of Gen-1 IoT and Figure 3(b) depicts the field deployments of the same. The Gen-1 IoT's preceptor is a float switch that sends a binary signal to the CPU when the rising water level triggers and closes the circuit. Until the threshold is reached, the float switch remains deactivated. Table I shows a sample reading during flooding and non-flooding/regular scenarios, including one flood triggering event denoted by

Y in the last column. When the float switch is closed, i.e., flood level is reached, the CPU transmits a byte into our on-premise server provisioned with LAMP (Linux, Apache, MySQL, PHP). The server-side programs based on JAVA and cron are used to detect the signal and send email notifications based on the set thresholds.

TABLE I
SAMPLE READING FROM GEN-1 IOT

TimeStamp	**Temp**	**Flood Threshold**
10/11/2017 21:18	74.6	N
10/15/2017 23:42	76	**Y**
10/16/2017 9:18	77	N

2) Shortcomings and Challenges: While this deployment gave us ample learning opportunities, and we iterated multiple times within this generation, the following are some of the challenges and shortcomings of this deployment.

- *Data Limitations:* This device is only capable of sending a binary signal when the preset flood threshold is reached.
- *Limited Predictive Capability:* Real-time flash flood detection requires a predictive model which can forecast future flood level in the area. However, the data limitation capability of this system does not accommodate the predictive power well. This deployment would be able to create at most a *Rule Based* engine.
- *Cost:* Given a limited data ability for our application, the device is also cost-prohibitive. The equipment cost us almost two thousand dollars.
- *International Device:* Being an imported device connecting to the American network service providers is a challenge

Among others, this deployment possessed an unforeseen impediment for us. This is an imported device from a Spanish company called Libelium. The Libelium unit was unable to connect with any of the leading US telecommunication systems such as AT&T, Verizon, T-Mobile, etc. Finally, we had to work with the vendors from Spain to get the basic functionally to work. To mitigate the problem, the company had to develop a special device solely for our research purposes.

3) Lessons Learned: Making Gen-1 functional has been quite a difficult and time-consuming endeavor. First and foremost, this is our first cyber-physical system to be deployed and tested outside the controlled lab setting. We had to make sure that the device will be deployed in a secured area and away from vandalism or theft. Although the unit is marketed as a Plug & Play unit, it required a lot of custom coding to make it operational. We had to study and understand various hardware, software, and networking aspects of the unit.

TABLE II
GEN-1 IOT DEPLOYMENT SUMMARY

Gen-1 On-Prem IoT	
Network Protocol	3G
Perceptor	Flood Level Threshold Sensor
The application layer	Custom Code (LAMP, JAVA)
Major Capability	Binary Data and Email Trigger

Once we were able to operate the device, it was adequately reliable. We deployed the system for more than three months without any problem. The server performed the critical function of this unit, and hence the unit itself is less prone to failure at the IoT node. Table II summarizes the findings from our Gen-1 deployment. The main capability of this deployment is being able to send a binary trigger to the server when the unit would reach the preset flood threshold. As soon as the water level rose to the float sensor level, the rising water level closes the circuit, and IoT sends a binary signal. The signal then triggered a series of server deployed applications such as database inserts, SMS, and email alerts.

4) Motivation for Gen-2: Thereafter the deployment of Gen-1 and learning from them, we move to our second-generation (Gen-2) deployment. The main motivation for us to delve into this iteration is to try to build an in-house lab unit and circumvent the main challenges presented by Gen-1. In Gen-2, we strive for a more economical solution and look for a richer data set. We also lost a significant amount of time in the back and forth shipment of the devices across continents. Soon we realized that working across the globe and timezone was not a viable solution. Subsequently, we move on to the next options and explore building an in-house flash flood detection system.

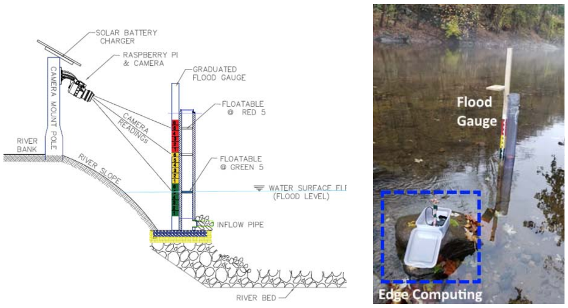
B. Gen-2 Edge Computing IoT

This deployment primarily explored the concept and working of edge computing. There are two separate units within proximity, the physical unit, a riser structure, and the computational edge unit. The riser structure is a color-coded flood gauge.

1) Method and Materials: The IoT system is shown in Figure 4(a), where the perceptor is a camera unit continuously taking a picture of the Riser Structure. The Field Deployment for Gen-2 is depicted in Figure 4(b) with RaspPi_Camera and Flood_Gauge marked on the Figure 4(b). In-depth analysis and deployment detail are discussed in our previous work [9].

Edge Computing: The unit is based on scene text recognition, which allows locating the area of interest with an image. We use various image pre-processing techniques such as background subtractions, template matching, and Region of Interest (ROI) trimming to isolate the areas to perform information extraction. We use the K-Means clustering technique to separate the most dominant colors.

We then leverage digit recognition techniques to identify the flood level in a stream. The edge computing unit is a microcomputer (Raspberry Pi), runs on the Linux operating



(a) IoT Description (b) Field Deployment

Fig. 4. Gen-2 Deployment

system. We installed the MySQL database to store and manage the meta-data such as image, time, label, size, deployment location, etc. The unit contains a pre-trained deep learning model capable of recognizing digit and color. The edge unit used K-Means clustering to identify the most dominant colors. We then used digit recognition techniques to determine the flood level in a stream. The CPU unit is a Raspberry Pi unit loaded with a pre-trained deep learning model capable of recognizing digit and color. At the end of each run, the unit takes a picture, performs image pre-processing, and prepares input for the color and digit recognition deep learning model.

2) Shortcomings and Challenges: This device is built locally, with commodities such hardware (pipes, gauges, overflow structures) and software (computing models, image data storage and processing), it has its own set of challenges and learning opportunities. Some of the major challenges and shortcomings of this deployment are listed below.

- *Stability:* The major challenge in this design and deployment is the physical stability of the unit. Figure 4(b) shows the precarious physical stability of this deployment.
- *Reliability:* Lose wire connections, dangling parts, and stand-alone camera reduced the reliability of this unit.
- *Safety:* The units will also be hard to deploy in harsh flooding or unforeseeable scenarios. It would be easily swept away during flooding and misplaced.
- *Accessibility:* The camera is unable to capture clear images in extreme/foggy weather and at night.

TABLE III
GEN-2 IOT DEPLOYMENT SUMMARY

Gen-2 Edge Computing IoT	
Networking	On Device Storage
Perceptor	Camera Unit
The application layer	Pre-Trained Deep Learning Model
Major Capability	Real time Flood Detection on Edge

3) Lessons Learned: This work is based on the challenging area of computer vision and provided us with good research opportunities. However, besides the shortcoming listed above, the main challenge of this unit is its inability to detect the flooding condition at night. The computer vision technique described above works on camera being able to read the color and digit, and hence during the night, the device would be unable to detect flood level. The unit taught us basic knowledge about the cyber-physical device's development that different yet intricate components need to work in tandem to have a successful IoT. These main components, as described, included the hardware (physical units), the IoT sensing electronic components, and the final software components. As seen, they are often the expertise of different engineering groups such as mechanical/civil, electronics, and software engineering. To that end, it would have been an uphill battle for a single lab to produce a scalable device in this generation. Nevertheless, we believe this unit has huge potential, and further data exploration and research activities are currently in progress.

4) Motivation for Gen-3: With our lessons learned from the previous generation and experimenting with them for some time, it is evident that the most time-consuming part of IoT development is its reliability, physical stability, and data granularity. We finally decided to move towards the off the shelf IoT product solutions. During our first two deployments, we learn that maintaining in house server and hardware component is yet another overhead for the main research work, i.e., to detect flash flood readily. To that end, we decided to use Software as a Service (SaaS) technology and opted to use a cloud provider for day to day data storage and application management. Our main goal in moving to the next-gen is to focus more on building a robust machine learning solution and less on physical stability and mundane tasks.

C. Gen-3 Cloud Computing IoT

In Gen-3, we attempt to solve our previous flood detection problems by selecting the ready to use IoT product. The deployment, along with sensors of this system, is shown in Figure 5. The units come with the perceptors ready to connect to the CPU that transfer data to the cloud. Once the raw data is in the cloud server, we are ready to access them and perform machine learning actitivites.

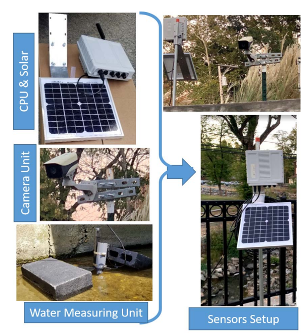


Fig. 5. Gen-3 Cloud Computing IoT

1) Method and Materials: One of the Gen-3 deployment setups is depicted in Figure 5, which displays two sensors water level measuring unit, a pressure-based water sensor, and the camera unit along with field deployment. The camera triggers when the flooding is sensed by the water measuring unit. Table IV shows sample reading from one of our deployed sensors. In this deployment, we use four such units into hydrologically significant stream locations. They are online and collecting real-time flood data, the Ellicott City, Maryland streams, which was devastated by flash in the past. Figure 6 shows the water level recorded at the same time by two different sensors located along the same tributary. It shows

TABLE IV
SAMPLE READING FROM GEN-3 IOT

RECORD_TIME	Water Level (inch)	Temp o F
10/10/2019 12:00	9.668	86.93
10/9/2019 1:00	5.828	60.67
10/12/2019 17:00	2.957	71.60

that the bottom graph is a feeder stream to the upper one and hence has a higher water reading. Gen-3 is already integrated with the social media platform **Twitter** to broadcast the sensor reading regularly via the twitter handle **umbc_floodbot** [1]. One such tweet is depicted in Figure 6.

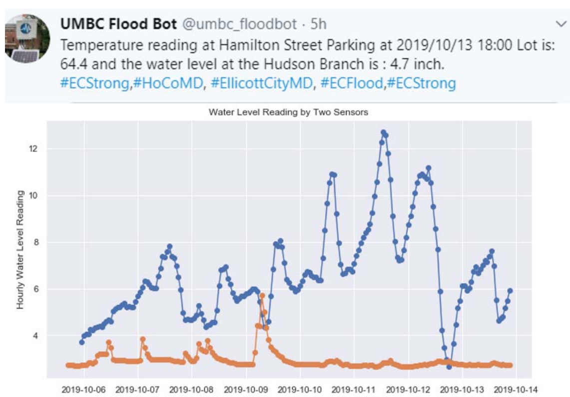


Fig. 6. Gen3-Data Dissemination

2) Shortcomings and Challenges: The systems are up and running online for a few months now. We have begun to understand the functionality of the unit and its challenges. The following are a few challenges that we have encountered so far.

- *Hardware Issues:* Once ordered, it took us a few months to get the system operational because of some internal hardware failure and network connectivity challenges.
- *Proprietary Parts:* The device and its perceptor (CPU and other units) are all proprietary to the vendor, and hence we have limited accessibility and low modifiability to the inner functionality of the unit.
- *Stability:* One recent flash flood event, washed away one of the perceptor (pressure transducer); thus, this unit is also susceptible to physical stability.
- *Clogging Debris:* The sensor unit is submerged in the water and often gets clogged by siltation and other debris. The debris also causes connectivity issues and data loss.

3) Lessons Learned: As of the writing of this paper, Gen-3 units have been live for a few months. Similar to previous deployments, this deployment gave us ample learning opportunities. As we progressed from Gen-1, Gen -2, we notice that some of the same fundamental challenges still remain. For example, the washing away of our sensor on Gen 3 or vandalism, the reliability in network connectivity are all still ongoing challenges. However, we do believe that this deployment is one of the significant achievements for us and is expected to bring multi-facet learning opportunities. We have been expanding and collaborating with other vendors to deploy more sensors in the area. The unit is fully integrated with a camera, which allows us to see and validate the data remotely.

In addition to the current research, we have another entirely different and noble research direction that has emerged from this deployment. Given the easy integration of sensor reading and social media, we are exploring various ideas to integrate the cyber-physical system with social media. The reading from this device and images captured are propagated as tweets using the **umbc_floodbot** twitter handler. On top of four sensors, we are also recording weather data (historical and foretasted) for the area, so it will be an interesting research work to see if we can predict the flow patterns based on weather and rainfall data. Similarly, ample opportunity for research lies in the integration of computer vision into a physical phenomenon captured by our deployed sensors.

[1]https://twitter.com/umbc_floodbot

TABLE V
GEN-3 IOT DEPLOYMENT SUMMARY

Gen-3 Cloud Computing IoT	
Network	Cloud Connected 3G
Perceptor	hydro-static level sensor
The application layer	Cloud hosted API
Major Capability	Water Level, Images and Social Media Posting

V. RESULT AND DISCUSSION

We evaluate the overall success of our deployments based on the following evaluation criteria.

A. System Evaluation

We score the systems and their success based on evaluation criteria discussed by Fahmideh et al. and Maoling and et al. [10], [11]. Based on their study, we have selected the four evaluation metrics to assess the quality of our deployment and system. *Performance* to measure the device capacity from parallelism, their ability to query the unit from both multiple user interfaces and Operating Systems. *Modifiability* to measure the ability and flexibility to make changes from both hardware and software in the deployment. *Reliability* to measure the overall reliability of the system. *Availability* to measure the capability of usage and execution of the software developed during intervals of time.The results are shown in Table VI. Gen-1 deployment was available for the duration of execution time despite difficult communication networks. Since the Gen-1 was primarily a plug and play device, it had less adaptable to change for scalability and customized needs. The performance for Gen-1 was limited as it could not support multiple users or distributed environments. Gen-2 deployment's availability was minimal since it was designed and installed in a precarious state and dependent on the mercy of environmental/weather conditions.

The overall performance of the Gen-2 system was medium since the data collected and analyzed using this system was not up to the mark due to the physical instability of the system and bad image data due to the night vision of the camera.

As yet, the Gen-3 deployment seems to be promising in fulfilling the gaps left out by the previous Gen deployments. Gen-3 is high in availability .We can access the data in real-time through the API call and dashboard. The sensors and devices are efficient and well connected to the network. The Gen-3 system would still face extreme physical/weather conditions but more reliable than the previous Gen systems. Gen-3 uses fully functional, pre-built, ready to use systems with data visualization and web interface. Thus, it does not provide much flexibility in modifying the system, making it less in Modifiability and more reliable than our Gen-2 deployment. The performance of the Gen-3 system is expected to be higher than the previous Gen systems because the good quality of data, system stability, network connectivity, and data analysis would be superior relatively. Gen-3 has the potential to achieve the final goal with efficiency.

TABLE VI
EVALUATION METRICS

Evaluation Report Card				
IOT	*Performance*	*Modifiability*	*Reliability*	*Availability*
Gen-1	Low	Low	High	High
Gen-2	Medium	High	Low	Limited
Gen-3	High	Low	High	High

B. Point of Failure Analysis

Table VII shows the information against our 3-layered architecture. The weakest link and lesson learned from these generations are summarized in table VII.

TABLE VII
WEAKEST LINK ANALYSIS

Point of Failure Analysis		
Deployment	*Weakest Link*	*Discussion*
Gen-1	Perceptor Gateway Application	Float switch does not give granular flood stage
Gen-2	CPU	Device lacked proper enclosure for harsh weather Image-based solution needs night-vision
Gen-3	Perceptor Gateway	Data loss issues, Cloud Connectivity

C. IoT-Service Delivery Model

We observed that the proper selection of IoT Service Delivery Model is also important in the overall success of the IoT implementation. Gen-1 and Gen-2 were based on-premise server for data analytic, and since we implemented all the server-side code, the overall process is very reliable. All the challenges that we are currently facing are mostly attributed because IoT is an evolving field and applicable to other projects as well. Hitherto, Gen 3 seems to be the best solution for scalable IoT deployment. This mode of deployment could be better used to perform more sophisticated tasks such as utilizing machine learning and Artificial Intelligence methods to find suitable smart solutions.

VI. CONCLUSION AND FUTURE WORK

In this work, we have presented our experience in creating IoT solutions around three heterogeneous environments. We have presented our learning from all of these implementations. We have intentionally left out the discussion from the data quality and analytically perspective. In our future work, we will divulge more on each of the unit's performances and the readiness of data usage for a decision support system and their machine learning potentials. IoTs play an important role in informing people and, more so, during disastrous situations. We firmly believe that the success of smart cities lies in the success of many successful IoTs. Furthermore, it has been our experience that the IoTs (hardware) and the application (software) are two different and rich research area in themselves. Looking at all three option and our experience with cloud deployment, IoTs are the perfect candidate for cloud-based solutions.

VII. ACKNOWLEDGMENT

This work is partially supported by the NSF CNS grant 1640625.

REFERENCES

[1] P. N. Andrea Caragliu, Chiara Del Bo, "An overview of smart sustainable cities and the role of information and communication technologies," *ITU-T Focus Group*, vol. 22, p. 2014, 02 2015.

[2] K. Su, J. Li, and H. Fu, "Smart city and the applications," in *2011 international conference on electronics, communications and control (ICECC)*. IEEE, 2011, pp. 1028–1031.

[3] S. A. Kumar, S. Bao, V. Singh, and J. Hallstrom, "Flooding disaster resilience information framework for smart and connected communities," *Journal of Reliable Intelligent Environments*, vol. 5, no. 1, pp. 3–15, 2019.

[4] K. Ashton *et al.*, "That 'internet of things' thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[5] R. S. Alonso, I. Sittón-Candanedo, S. Rodríguez-González, Ó. García, and J. Prieto, "A survey on software-defined networks and edge computing over iot," in *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2019, pp. 289–301.

[6] J. Dizdarević, F. Carpio, A. Jukan, and X. Masip-Bruin, "A survey of communication protocols for internet of things and related challenges of fog and cloud computing integration," *ACM Computing Surveys (CSUR)*, vol. 51, no. 6, p. 116, 2019.

[7] E. Siow, T. Tiropanis, and W. Hall, "Analytics for the internet of things: A survey," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, p. 74, 2018.

[8] S. Maitra and K. Yelamarthi, "Rapidly deployable iot architecture with data security: Implementation and experimental evaluation," *Sensors*, vol. 19, no. 11, p. 2484, 2019.

[9] N. R. Bipendra Basnyat and A. Gangopadhyay, "A flash flood categorization system using scene-text recognition, in proceedings of the 4th ieee international conference on smart computing," 06 2018.

[10] M. Fahmideh and D. Zowghi, "Iot smart city architectures: an analytical evaluation," in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 2018, pp. 709–715.

[11] M. Yan, P. Liu, R. Zhao, L. Liu, W. Chen, X. Yu, and J. Zhang, "Field microclimate monitoring system based on wireless sensor network," *Journal of Intelligent & Fuzzy Systems*, vol. 35, no. 2, pp. 1325–1337, 2018.