Edge AI in Smart Farming IoT: CNNs at the Edge and Fog Computing with LoRa

T. Nguyen Gia[1], L. Qingqing[1,2], J. Peña Queralta[1], Z. Zou[2], H. Tenhunen[3] and T. Westerlund[1]

[1] Department of Future Technologies, University of Turku, Finland
[2] School of Information Science and Technology, Fudan Universtiy, China
[3] Department of Electronics, KTH Royal Institute of Technology, Sweden
Emails: [1]{tunggi, jopequ, tovewe}@utu.fi, [2]{qingqingli16, zhuo}@fudan.edu.cn, [3]hannu@kth.se

***Abstract*—The agricultural and farming industries have been widely influenced by the disruption of the Internet of Things. The impact of the IoT is more limited in countries with less penetration of mobile internet such as sub-Saharan countries, where agriculture commonly accounts for 10 to 50% of their GPD. The boom of low-power wide-area networks (LPWAN) in the last decade, with technologies such as LoRa or NB-IoT, has mitigated this providing a relatively cheap infrastructure that enables low-power and long-range transmissions. Nonetheless, the benefits that LPWAN technologies enable have the disadvantage of low-bandwidth transmissions. Therefore, the integration of Edge and Fog computing, moving data analytics and compression near end devices, is key in order to extend functionality. By integrating artificial intelligence at the local network layer, or Edge AI, we present a system architecture and implementation that expands the possibilities of smart agriculture and farming applications with Edge and Fog computing and LPWAN technology for large area coverage. We propose and implement a system consisting on a sensor node, an Edge gateway, LoRa repeaters, Fog gateway, cloud servers and end-user terminal application. At the Edge layer, we propose the implementation of a CNN-based image compression method in order to send in a single message information about hundreds or thousands of sensor nodes within the gateway's range. We use advanced compression techniques to reduce the size of data up to 67% with a decompression error below 5%, within a novel scheme for IoT data.**

***Index Terms*—IoT; Internet of Things; Smart Agriculture; Edge Computing; Fog Computing; Edge AI; LoRa; LPWAN; Low Power Wide Area Networks; CCN; Deep Learning; ML;**

I. INTRODUCTION

In order to perform activities such as watering or fertilizing, farmers need to visit their plants frequently (e.g., every day or every few days depending on the plant and trees). In some cases, farmers need to stay close to their remote farms in order to protect the crop and their resources. When the farmed areas are large, it becomes increasingly difficult and more human resources are required to perform these tasks. This can cause a significant increase in operational costs with a limited impact on productivity. In the era of the Internet of Things (IoT), a solution is to deploy remote monitoring and management systems for these remote farms. Farms adopting IoT technologies are often referred to as smart farms. However, remote farm monitoring and controlling have limited impact in areas with unstable or poor Internet connectivity. This occurs not only in developing but also in developed countries [1].

The Internet of Things (IoT) can be defined as a platform where virtual and physical objects are interconnected and communicate with each other. IoT systems consist of different technologies such as wireless sensor networks, cloud computing and embedded intelligence. These systems offer advanced services such as real-time remote monitoring, online analytics and remote management. IoT is applied in many remote monitoring applications in vast domains from healthcare to smart factories, and including smart homes, smart cities and smart farming, improving productivity and reducing costs [2]–[6]. Some of the benefits of the IoT can be utilized to improve the quality of services for automated and remote farming systems. However, relying only on traditional cloud-centric IoT architectures for remote farm monitoring and management cannot guarantee that the systems work properly because the IoT still presents several challenges. For instance, cloud-centric IoT applications cannot be deployed in remote areas where the Internet is not stable or coverage is limited. In such cases, data cannot be real-time monitored and actions toward abnormalities might not be executed on time. For instance, if a sudden fire occurs or a group of wild animals raid the crops, the system cannot react on time.

Edge and Fog computing can be illustrated as a mini cloud which is closer to the edge of the network. In other words, Edge and Fog computing represent the convergence of different network layers into interconnected smart gateways. Edge and Fog computing can help to overcome some limitations of the traditional could-centric IoT systems. For instance, Edge and Fog computing offer many advantages such as energy efficiency, distributed local storage, interoperability and enhanced security. In more detail, Edge and Fog computing can help to reduce the network load and the computational and storage burden of cloud servers. This is done by moving many computationally intensive processes from the cloud to the Edge and Fog layers and gateways, at the same time enabling more power-efficient sensor nodes as they rely more on local network smart gateways. Compared to traditional IoT applications which often rely on a 3-layer architecture (sensor-cloud-terminal), a Fog-assisted IoT application has extra layers between the sensor nodes and the cloud. Depending on the application and type of acquired data, a different number of Edge/Fog layers can be deployed [7]–[9].

Even though Edge and Fog computing can provide many advanced services, Fog-based systems still cannot work properly in remote areas where the Internet is not stable or covered, as they often rely on high-speed local networks for real-time processing and latency-critical applications. A solution in these scenarios is to deploy low-power wide-area network technologies, such as LoRa, which enable long-range transmissions with the drawback of reduced data rates. LoRa is one of the most popular LPWAN protocols for the physical layer [10], offering low-power and long-range communication up to 10 or 20 km in open and line-of-sight transmissions [11]. However, LoRa cannot be used to send data with high data rate due to local regulations and limitations to the transmission duty cycle in most regions of the world of 0.1%, 1% or 10%. Therefore, LoRa alone cannot help to solve the existing problems of IoT applications in remote areas.

In this paper, we propose an advanced Edge/Fog assisted LoRa-based system for remote farming. The system targets remote areas in developing countries where the need is evident. We propose to leverage Edge and Fog computing, together with LoRa communication to back the limitations of each other while still providing smart services and more efficient computation distribution when compared to cloud-centric solutions. In addition, the integration of Edge and Fog computing with LoRa into IoT systems can help to achieve a high level of energy efficiency for sensor nodes. When some abnormalities occur, Fog-assisted IoT systems can provide support for latency-critical applications.

As LoRa has limited transmission speed, and there are strict regulations controlling the duty cycle, we propose a novel data compression technique integrating the spatial distribution of sensor nodes, which can be densely deployed in smart farms, with image compression methods. We have simulated the output of thousands of sensor nodes and applied different image compression techniques. We compare our proposed method with standard JPEG compression and show how can we achieve up to 67% reduction of data size while keeping decompression error under 5%. This is a threshold often found in many inexpensive sensors. For instance, it is equivalent to having a 1 degree accuracy in a temperature sensor measuring around 20^o C. The main contributions of the paper are:

- Advanced architecture for monitoring and controlling remote farms where Internet connectivity is not reliable.
- Integration of image compression techniques with convolutional neural networks to compress data from multiple sensor nodes at once exploiting their spatial distribution.
- Enhancement of edge gateway for pre-processing data and considerably reducing the amount of data to be transmitted over the LoRa link.

The remainder of this paper is structured as follows: Section II presents related work in remote monitoring and management systems for smart farming. Section III describes the architecture of a Fog-assisted IoT system with the integration of Lora. Section IV includes the system implementation, edge layer analytics and discusses experimental results, while section VI concludes the work.

II. Related Work

Many efforts have been devoted to proposing smart and remote farm monitoring. Maia *et al.* [12] present a remote monitoring system for precision agriculture. The system consists of monitoring nodes, central nodes, and node in cloud. Data collected from sensors (e.g., soil temperature and humidity) integrated into a monitoring node is sent to a central node via ZigBee. The central node then forwards the received data to nodes in cloud via the Internet.

Ragai *et al.* [13] present are a remote control and monitoring system for fish farms by using wireless sensor networks. The wireless sensor node collects different data such as temperature, pH, dissolved Oxygen and sends the collected data via ZigBee to gateways forwarding to cloud servers via the Internet. When cloud servers detect abnormalities such as too low or too high value of temperature and pH, they send an alarm to farmers.

Saraf *et al.* [14] propose an IoT-based system for smart irrigation monitoring and controlling. The system consists of wireless sensor nodes, a center node connected to cloud and an Android application. The sensor node acquires temperature, humidity, soil moisture and water level of the tank and transmits the data to a center node via Zigbee. The center node then forwards the collected data to cloud which runs data processing algorithms and do actions e.g., sending commands to actuator nodes to control water.

Shaout *et al.* [15] present an embedded system for remote agricultural areas monitoring. The system consists of sensor nodes which are based on Arduino board, Bluetooth and sensors. The sensor node can collect temperature, humidity, soil moisture and temperature, and transmit to Data Droid modem for further processing.

Yang *et al.* [16] introduce a remote farm monitoring system which consists of client block, server block and end-user client. Client block is made from an LPC11C14 microcontroller, fan, light sensor, temperature and humidity sensor, buzzer, LED and OLED display screen. The data collected from a client block is sent to server block via ZigBee. A server block comprises of a camera, Wi-Fi, GPRS, and ZigBee module. The acquired data is forwarded by the server block to an end-user client via Wi-Fi.

Ezhilazhahi *et al.* [17] propose an IoT-based system for plant soil moisture monitoring. Sensor nodes of the system collect soil moisture data and transmit the data via ZigBee to a gateway for data aggregation and transmission. The data then is sent to a remote server. End-users can use their mobile phone to access the server to achieve real-time soil moisture. Similarly, James *et al.* [18] present a remote monitoring system for plant growth. The system built from Rasberry Pi collects relative humidity, atmospheric temperature and soil moisture and sends the data via Bluetooth to a mobile application of a farmer.

Gayatri *et al.* [19] introduce a remote monitoring system for agriculture by using a combination of IoT and cloud computing. The system relies on Wi-Fi/4G networks for making the interconnection between sensors, gateways and cloud servers. End-users such as farmers can use a farmer console to control or give an instruction to sensor devices in their farms remotely.

Channe *et al.* [20] propose a smart system for remote monitoring agriculture. The system is based on IoT, cloud computing, mobile computing and bid-data analysis. The system senses soil and environmental properties, and then stores in cloud servers. In addition, the system processes and analyzes the data to extra useful information such as fertilizer requirements and best crop sequences. End-users such as farmers can access the raw data and extracted information stored in cloud servers remotely.

Although the mentioned works show benefits of remote monitoring, they still have many disadvantages (e.g., energy inefficiency) and cannot be suitable for remote monitoring farms in remote areas where the Internet connectivity is not covered or stable. Therefore, we propose an enhanced architecture utilizing Edge and Fog computing, IoT, and LoRa to overcome the mentioned problems and offer many advantages services such as data processing at the edge and push notification.

III. System Architecture

The system architecture is illustrated in Fig. 1. The proposed architecture consists of 5 layers, namely sensor layer, Edge layer, Fog layer, cloud layer, and terminal layer. The sensor and edge layers are connected via nRF in our proposed system design, but other wireless solutions such as Wi-Fi or Bluetooth 5 can be utilized if they provide enough bandwidth and range. The Edge and Fog layer are connected via LoRa, which in open areas can enable transmissions of over 10 or even 20 km with a low data rate. The Fog gateways are in turn connected to cloud servers via wired Ethernet or wireless solutions with high throughput such as Wi-Fi or mobile 4G/5G. The final layer consists of the front-end user application and interfaces.

A. Sensor layer

The sensor layer consists of several groups of sensor nodes. Sensor nodes and actuator nodes are part of this layer, and can be deployed to different areas of the farm. The main difference between these nodes is that the sensing node mainly collects and sends the data to Edge gateways in Edge layer while the actuating node primarily receives commands from Edge gateway to control actuators such as turning water system. A prototype sensor node is shown in Fig. 2, equipped with a micro-controller, nRF wireless module, and different sensors. Depending on the sensor node location, some of the sensor nodes are equipped with solar panels. Most of the hardware components of a sensor node are connected via an SPI wire communication protocol as SPI supports high data rate with low energy consumption.

B. Edge layer

The Edge layer is formed by Edge gateways which are responsible for receiving data from sensor nodes via nRF, processing data and sending the processed and compressed data to Fog gateways via LoRa. This intermediate step helps to significantly reduce the amount of data transmitted over the LoRa links, and fulfill the strict requirements of LoRa's duty cycle. An Edge gateway mainly consists of an embedded board with nRF and LoRa wireless modules. Some gateways can be equipped with sensors or cameras. These gateways are equipped with Linux because Linux offers many advantages such as free to use, lightweight, enhanced security, customization, reliability and high performance while Linux does not require powerful hardware.

Edge gateway offers many advantages services such as push notification, channel categorization and security. For instance, when an edge gateway detects abnormalities such as hardware failure in sensor nodes, the gateway can send an instant message to end-users such as farmers or system administrators to inform about the problem. The possibilities of data processing in the Edge layer range from multi-robot mapping [9], to bio-signal analysis [2].

An nRF protocol supports more than 100 channels in which each channel has its own corresponding frequency (e.g.,). This creates a premise to utilize these channels in the most optimized way to reduce error rate or bandwidth limitations. Some channels are more interfered by noise surrounding or are busier due to a large amount of data transmitted by a vast number of sensor nodes. It is recommended to use a suitable channel or a group of channels in a specific moment. This service can be implemented in Edge gateways.

In order to provide some levels of security, Edge gateway can implement some algorithms including encryption algorithms such as AES-128, AES-256, or cryptography algorithms such as ECDSA. Although running these algorithms can increase latency and an Edge gateway's energy consumption, the overhead is not significant because an Edge gateway is often equipped with powerful hardware and uses power-line from a power socket.

Edge gateway can run data compression or data processing methods to reduce a large amount of data transmitted over a LoRa network. Depending on data or the applications, lossless or lossy data compression can be applied at Edge gateways. Lossy compression can provide a high compression rate e.g., 50:1 while lossless compression often offers a low compression rate e.g., 10:1 For example, temperature data can be applied with lossy compression as temperature is often collected by several sensors in an area and the changing rate of temperature is not fast in terms of minutes. Therefore, losing or missing a sample or several samples cannot reduce a system's reliability. Besides the mentioned services, Edge gateway can offer other advanced services such as interoperability, data fusion and mobility support.

In this paper, we propose a novel integration of image compression techniques with accumulated data from multiple sensor nodes. Rather than using standard JPEG or other lossy compression techniques, we have implemented a compression solution based on deep learning. Convolutional neural networks have been increasingly used in recent years for

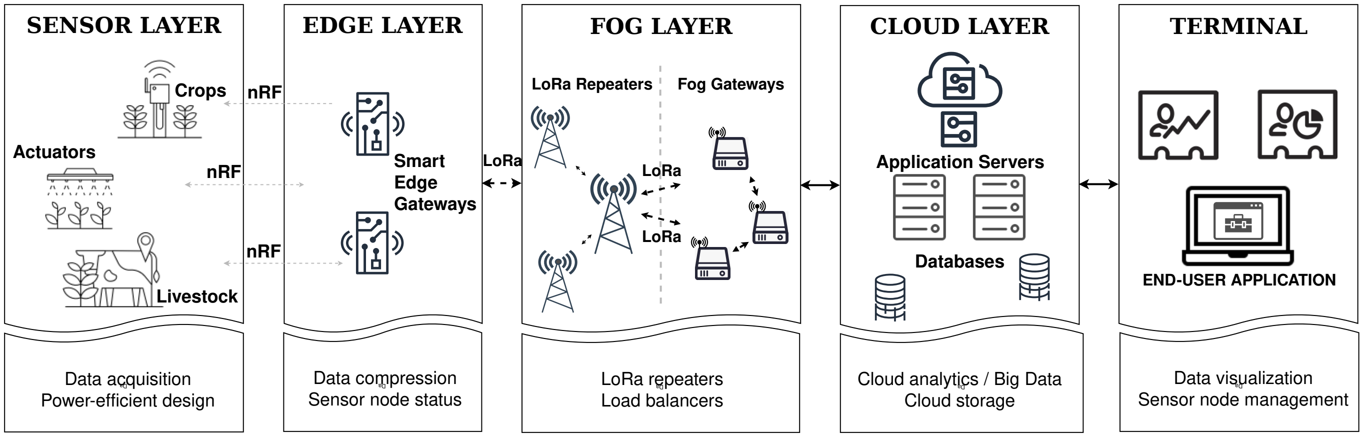


Fig. 1. Proposed 5-layer Sensor-Edge-Fog-Cloud-Terminal system architecture with LoRa wireless link used as the Edge-Fog bridge

lossy compression of images [21]–[23]. Convolutional neural networks provide robust methods for compression with less error after decompression when compared to traditional methods such as JPEG. We utilize a semantic perceptual image compression method proposed by Prakash *et al.* [24]. Their method is able to highlight semantically-salient regions, which are in turn encoded with higher accuracy. We have selected this method for IoT sensor data as it can increase the compression rate when the data is very homogeneous, while preserving regions of interest with proper accuracy. In order to use this method, edge gateways first gather data from multiple sensor nodes, and this data is encoded in the pixels of an image. For instance, the three channels of an RGB image can be used to encode three different variables, and the position of pixels can represent the spatial distribution of sensor nodes.

C. Fog layer

There are two tiers in the Fog layer. The first tier consists of an optional group of repeaters which mainly receive the data transmitted from Edge gateways and forward the data to Fog gateways via LoRa. The use of this tier and the specific number of repeaters depend on the distance between the deployment location, such as a remote farm, and the nearest Fog gateway. For instance, according to a work from Petäjäjärvi *et al.*, about 20 repeaters are required to maintain a stable LoRa communication link over a distance of 300 km, in near line-of-sign transmission [25]. In the same work, the authors warn that even though LoRa can provide long-range transmissions (e.g. 25 to 30 km), packet loss data increases dramatically after the 10-15 km threshold.

The second tier in the Fog layer consists of a series of interconnected Fog gateways which are communicated with each other and share information about sensor nodes, whether they are in a near location or not. If fog gateways are not connected through a local area network, then VPNs or other virtual network solutions can be utilized. Fog gateways are located in areas where there exist stale and reliable Internet connectivity. Fog gateways are connected to cloud servers via Wi-Fi, Ethernet or 4G. Similar to Edge gateways, Fog

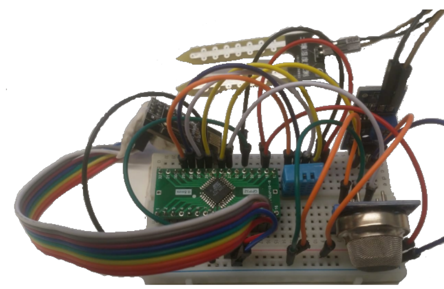


Fig. 2. Sensor Node Prototype

gateways can offer advanced services such as push notification, distributed data storage, security, data fusion, and data processing. More detailed information of Fog services and are discussed in our previous articles [7], [8], [26]–[28].

D. Cloud layer and end-user terminal layer

Cloud layer consists of cloud servers and its services such as global data storage, big data analysis, push notification, data processing with complex algorithms. Depending on the application, specific cloud services can be implemented.

End-user terminal layer consists of mobile applications and web browser which are used to access real-time data and input commands to control actuators in a remote farm.

IV. Implementation and Results

In this paper, a complete system for remote monitoring and controlling a farm in a remote area is implemented. In particular, a group of sensor nodes including sensing and actuating nodes, an Edge-gateway, a repeater, a Fog gateway, cloud servers and end-user terminal are implemented.

A sensing node is implemented with an nRF module nRF24L01, a microcontroller AVR ATmega8, a group of sensors (i.e., DHT111 temperature and humidity sensor and soil sensor). Some of the sensor nodes can use the power socket while some can be equipped with solar panel and battery. Depending on the placement of the sensing nodes, one of the mentioned methods is applied for providing power for sensing nodes. The sensing node mainly senses and transmits the collected data to an Edge gateway. Therefore, it is programmed to sleep in most of the time to save power except for a moment when it senses and transmit data. When it is working, the radio receiving part is turned off to save power.

An actuating node is also implemented with an nRF module nRF24L01, a microcontroller AVR ATmega8, a relay or a controlling circuit to turn on/off or control the actuator such as a water system. Different from a sensing node, an actuating node is powered by a power socket and its radio receiving part is turned on always to wait for the command from a farmer or a system administrator.

The Edge gateway is implemented with a Rasberry Pi v3B, which has quad-core 1.4 GHz CPU and 1G RAM. The Rasberry Pi is connected with nRF24L01+PA+LNA for receiving data from sensor nodes via nRF. Comparing with an nRF24L01 module having a printed onboard antenna, an nRF24L01+ module has a long antenna. This long antenna helps to increase receiving power at the receiver end although it also increases energy consumption of the Edge gateway. The high energy consumption of an edge gateway is not a problematic issue as edge gateways are often powered by a power socket. In addition, the Rasberry Pi is connected with a LoRa chip (i.e, Dragino hat including LoRa chip and GPS) for sending the processed data to repeaters.

In this paper, a repeater is a simple LoRa gateway which receives data from several Edge-gateways and forwards the data to Fog gateways via LoRa. In future work, a repeater can be more investigated to perform some other services such as data compression, and data processing.

Fog gateway is implemented with a LoRa gateway and an Intel UP gateway. Fog gateway is equipped with Ethernet, Wi-Fi and 4G module for connecting to the Internet and interconnecting between each other. In nominal operation, Ethernet is preferable when it is available. Otherwise, Wi-Fi and 4G are used. Fog gateway is powered by a power socket. Therefore, power consumption is not an issue. In this paper, we reuse and adapt fog services, cloud applications, and end-user terminals which have been implemented in our previous papers [7], [8], [26]–[28]. The user interface has been modified for data visualization.

A. Sensor Node

The implemented sensor node collects and transmits data (i.e., temperature, humidity, or soil moisture) with a data rate of 1 sample per minute to Edge gateways which are about 200 meters. This data rate is sufficient for farm applications as temperature, humidity, soil moisture and other similar data do not change frequently in terms of minutes. The sensor node is supplied with 3.3V and its energy consumption is measured with a power monitor tool MonSoon [29]. The energy consumption of the sensor node per minute is 138.92 mJ. In addition, we also test the transmission rate of the sensor node by sending more than 1000 samples with line-of-sight. The result shows that the success rate is larger than 99% and less than 5 samples are collapsed. If we use a 2000 mAh battery (5.6mm x 49.2mm x 68.8mm), a sensor node can work up to 863 hours. When a small solar panel (145mm x 145mm x 2mm) is used at noon with Finland sunlight, we can harvest around 1650 mJ per second. This number should be much larger if the solar panel is placed at Africa where sunlight is much more powerful at noon. With the solar panel and an average of 4 hours per day in Africa, a sensor node's battery can fully be charged. Lithium batteries often allow for 400 to 500 charging cycles. Therefore, a sensor node with the solar panel can be used for the same number of days.

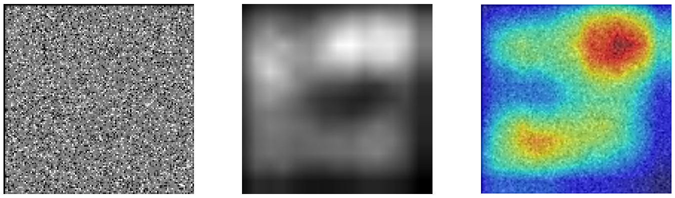
For verifying the functionality of the proposed system, we use a Web browser to access data including temperature, humidity and soil moisture. The results show that the system works properly. For example, temperature and relative humidity of the experimented land is shown (i.e., 20 degrees Celsius and 57%, respectively) in real-time. In addition, we also send some commands via a Web interface to control a water system of the experimented land.

B. Data Compression

In order to validate the proposed compression method, we have simulated the generation of 16384 samples, each with 3 8-bit channels. These channels can be used, for instance, for temperature, humidity and air quality data. The data is then saved as a 128x128 image, which is compressed using JPEG format. Data can be structured such that neighbor pixels in the image correspond to neighbor sensor nodes when taking into account their spatial distribution. Instead of sending information about individual sensor nodes, edge gateways gather data from all of them and then merge the data into an image where spatial information is encoded. The correspondence between sensor node identifiers and image pixels must be done either beforehand or through an initial message when the edge gateways are configured.

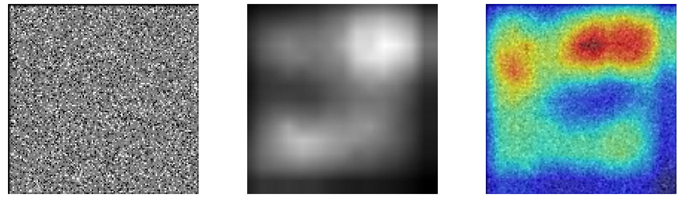
Figure 3 shows an example of image compression using the semantic perceptual image compression for simulated random data with equal mean and variance through the image. A different method has been used to generate simulated data in Figure 4, where the image has been divided in four quadrants of 64x64 pixels each and different mean and variance have been applied. In a real scenario, the first approach simulates a field with only one kind of crops and homogeneous conditions, where the second approach simulates a field with multiple crops or different soil conditions. In both cases, sub-figure (a) represents the random data, while (b) illustrates the map generated by the method from [24], and (c) shows the overlay of the original image and the map.

The compression error and reduction of data batch size is shown in Tables I and II, where we can see that for 57% compression rate the error amounts to only 2%, a reduction of over 40% when compared to standard JPEG compression. We can also see that error rates decrease when there is more information added to the image. This is particularly beneficial for detecting abnormalities with higher precision due to the semantic compression method utilized.



(a) (b) (c)

Fig. 3. Illustration of simulated data with constant distribution.



(a) (b) (c)

Fig. 4. Simulated data with four different distributions.

TABLE I
COMPRESSION COMPARISON IN RANDOM NOISE

Compression	Error	Size reduction
JPEG-50%	3.73%	57%
JPEG-30%	6.17%	67%
CNN+JPEG-50%	2.53%	58%
CNN+JPEG-30%	4.55%	67%

TABLE II
COMPRESSION COMPARISON IN 4-QUADRANT NOISE

Compression	Error	Size reduction
JPEG-50%	3.41%	56%
JPEG-30%	5.45%	67%
CNN+JPEG-50%	2.08%	57%
CNN+JPEG-30%	4.37%	67%

V. CONCLUSION AND FUTURE WORK

We have presented a hybrid 5-layer architecture for IoT systems deployed in smart farms. The proposed architecture consists of sensor, Edge, Fog, cloud and terminal layers. This architecture can be used to multiply the options of IoT deployments with LoRa or other LPWAN technologies that provide low-power and long-range transmission but limited data rate. We propose different methods for compressing data and reducing the LPWAN link load, minimizing the amount of data transmitted and stored in cloud servers while maintaining a similar user experience. In particular, we propose a novel integration of image compression algorithms to be applied in IoT data from the agricultural and farming industries, such as temperature, humidity, air quality or soil properties. In future work, we will focus on deploying multiple sensor nodes in a real scenario and provide a wider set of compression and analytics algorithms for the edge layer.

REFERENCES

[1] S. Tibken. In farm country, forget broadband. you might not have internet at all. Accessed: Jun, 2019, Updated: Oct, 2018, Available: https://cnet.co/2AmEWgL/.

[2] J. Peña Queralta *et al.* EdgeAI in LoRabased healthcare monitoring: A case study on fall detection system with LSTM Recurrent Neural Networks. In *IEEE 42nd TSP*, 2019.

[3] T. N. Gia *et al.* Low-cost fog-assisted health-care iot system with energy-efficient sensor nodes. In *2017 IWCMC*, pages 1765–1770. IEEE, 2017.

[4] I. Tcarenko *et al.* Energy-efficient iot-enabled fall detection system with messenger-based notification. In *International Conference on Wireless Mobile Communication and Healthcare*, pages 19–26. Springer, 2016.

[5] M. Jiang *et al.* Iot-based remote facial expression monitoring system with semg signal. In *2016 IEEE SAS*. IEEE, 2016.

[6] T. N. Gia *et al.* Iot-based continuous glucose monitoring system: A feasibility study. *Procedia Computer Science*, 109:327–334, 2017.

[7] T. N. Gia and M. Jiang. Exploiting fog computing in health monitoring. *Fog and Edge Computing: Principles and Paradigms*, 2019.

[8] A. M. Rahmani *et al.* Exploiting smart e-health gateways at the edge of healthcare internet-of-things: A fog computing approach. *Future Generation Computer Systems*, 78:641–658, 2018.

[9] V. K. Sarker *et al.* Offloading slam for indoor mobile robots with edge-fog-cloud computing. In *ICASERT*, 2019.

[10] J. Peña Queralta *et al.* Comparative study of LPWAN technologies on unlicensed bands for M2M communication in the IoT: beyond LoRa and LoRaWAN. In *14th FNC*, 2019.

[11] V. K. Sarker *et al.* A survey on lora for iot: Integrating edge computing. In *SLICE*, 2019.

[12] R. F. Maia *et al.* Precision agriculture using remote monitoring systems in brazil. In *IEEE GHTC*. IEEE, 2017.

[13] H. F. Ragai *et al.* Remote control and monitoring of fish farms using wireless sensor networks. In *12th ICCES*. IEEE, 2017.

[14] S. B. Saraf and D. H. Gawali. Iot based smart irrigation monitoring and controlling system. In *2017 RTEICT*, pages 815–819. IEEE, 2017.

[15] A. Shaout *et al.* An embedded system for agricultural monitoring of remote areas. In *11th ICENCO*, pages 58–67. IEEE, 2015.

[16] S. Yang, S. Tong, and L. Liang. Remote farm environment monitoring system based on embedded system and zigbee technology. In *2015 ICSPCC*, pages 1–5. IEEE, 2015.

[17] A. M. Ezhilazhahi, and P. T. V. Bhuvaneswari. Iot enabled plant soil moisture monitoring using wireless sensor networks. In *2017 ICSSS*, pages 345–349. IEEE, 2017.

[18] J. James *et al.* Plant growth monitoring system, with dynamic user-interface. In *2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pages 1–5. IEEE, 2016.

[19] M. K. Gayatri *et al.* Providing smart agricultural solutions to farmers for better yielding using iot. In *2015 IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR)*. IEEE, 2015.

[20] H. Channe, S. Kothari, and D. Kadam. Multidisciplinary model for smart agriculture using internet-of-things (iot), sensors, cloud-computing, mobile-computing & big-data analysis. *Int. J. Computer Technology & Applications*, 6(3):374–382, 2015.

[21] M. Li *et al.* Learning convolutional networks for content-weighted image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3214–3223, 2018.

[22] B. Bayar, and M. C. Stamm. On the robustness of constrained convolutional neural networks to jpeg post-compression for image resampling detection. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2152–2156. IEEE, 2017.

[23] L. Cavigelli *et al.* Cas-cnn: A deep convolutional neural network for image compression artifact suppression. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 752–759. IEEE, 2017.

[24] A. Prakash *et al.* Semantic perceptual image compression using deep convolution networks. pages 250–259, 04 2017.

[25] J. Petajajarvi *et al.* On the coverage of lpwans: range evaluation and channel attenuation model for lora technology. In *2015 ITST*, pages 55–59. IEEE, 2015.

[26] B. Negash *et al.* Leveraging fog computing for healthcare iot. In *Fog Computing in the Internet of Things*, pages 145–169. Springer, 2018.

[27] T. N. Gia *et al.* Fog computing in body sensor networks: An energy efficient approach. In *Proc. IEEE BSN*, 2015.

[28] T. N. Gia *et al.* Fog computing approach for mobility support in internet-of-things systems. *IEEE Access*, 6:36064–36082, 2018.

[29] Monsson Solutions. High voltage power monitor. Accessed: June,2019, Available: https://www.msoon.com/online-store.