Emotion Detection IoT enabled Edge-node for Citizen Security

Subhra Shankha Bhattacherjee
Department of Electrical Engineering
Indian Institute of Technology
Hyderabad, India
subhrashankha@iith.ac.in

Sanju Kumar N. T.
Department of Electrical Engineering
Indian Institute of Technology
Hyderabad, India
sanjutari@iith.ac.in

P. Rajalakshmi
Department of Electrical Engineering
Indian Institute of Technology
Hyderabad, India
raji@iith.ac.in

Abstract—In this paper, we have proposed a novel architecture for citizen security using deep learning based edge computing device. The proposed system comprises of an end-device which can be connected to every street light pole in an area. The gateway device can be connected at a large distance from the end-devices. LoRa physical layer is used as the main communication interface which enables large distance transmission. CNN based algorithm is used to identify the expression of the person who pressed the panic button on the device to reduce false alarm. A very low complex Convolutional Neural Network (CNN) architecture has been developed, to recognize the expression on the end device itself. A data logging server is used for logging the node-id everytime it is pressed. An android application is also developed which shows the real-time location of the pressed node and provides navigation routes to reach that node.

Index Terms—LoRa, Internet of Things, convolutional neural network, emotion recognition, edge computing

I. INTRODUCTION

IoT is an emerging wave for new service development and global economy growth, driven by billions of things being connected to the Internet. IoT vision of pervasively connecting billions of things is able to interact with environment around us and receive information on its status that was previously not available by simply looking at a set of things. One of the most extensively researched topics under IoT is smart cities. Citizen security, green environment, increased operational efficiency with the help of smart sensors and actuators are considered as some of the primary aspects of smart cities. Ideally smart city should be able to smartly secure its inhabitants. According to the National Crime Records Bureau, total number of crimes has increased from 47, 10, 676 in 2015 to 48, 31, 515 in 2016, i.e., about 2.6% in India [2]. Judging by the increase in crime rate we can conclude that available security solutions are proving to be incompetent. Smartphone based emergency alert system is a good start towards reaching this goal [3]–[5]. Numerous security applications based on Android OS and iOS have also been developed but there is no noticeable decrease in crime rates. The system may be effective in a society where everyone has a smartphone and has adequate knowledge of its usage. However, in developing countries and rural areas everyone does not have a smart phone and even if they do, they are not so proficient in handling it. So, smartphone based approach is futile in such cases.

Thus, there is a need for alternative communication standard that can provide us the range needed in a cityscape. This brings us to various surveillance systems like CCTV, WiFi-based monitoring etc. Child observation system using information terminal bus stops is on the rise in developed countries like Japan [6], [7]. In India CCTV surveillance is a fast growing security measure especially in crowded places like bus stops, railway stations and airports [8]. The drawbacks of these methods are – cost of deployment and cumbersome set-up. Using WiFi for wireless surveillance, limits the range up to less than 50 meters. The probable solution to this is LoRa RF. LoRa physical layer is specially built for use in developing IoT sensor networks [9], [10]. In general LoRa RF can broadcast signal upto around 2km, but can achieve range upto 10km in line of sight (LoS) with LoRa RF receiver with the help of directional antennas, which is exactly required for a city-wise deployment [11].

A smart city must be smart, meaning that the sensor nodes should be able to connect, work together as a single unit and learn from assembled results without being explicitly programmed. This brings us to machine learning and deep learning concepts [13]–[15]. Integrating machine learning within IoT networks can be an innovative perspective for smart city security. Recently machine learning based edge computing device has been built and deployed. Installing A.I. powered crime prediction and prevention surveillance network with cloud computing is already on the roll in metropolitan cities of India [16]. Given the huge success of CNN models in complex classification problem we have proposed a LoRa based alert system with a simple CNN model classifying facial expression at the edge-device.

More precisely, our contributions in this paper are:

- Proposed a low-powered and low-cost LoRa RF based closed loop alert system.
- Designed a CNN based edge-device with LoRa RF communication interface to recognize facial expression on-the-go.
- Developed an android application to show real time status of nodes.
Analyzed the proposed system on various parameters to justify its robustness.

The remainder of this paper is structured as follows: Section II describes the proposed system, Section III deals with the evaluation of our proposed network based on certain parameters and finally conclusions are drawn in Section IV.

II. PROPOSED NETWORK

The proposed network architecture as shown in Fig 1. consists of \( n \) number of LoRa based transceiver nodes, a gateway device, a data logging server and an android application. LoRa physical layer uses chirp spread spectrum (CSS) modulation scheme which is immune to interference since information is spread across several frequency channels [17]. LoRa uses unlicensed radio spectrum in the Industrial, Scientific and Medical (ISM) band. Thus, it enables quick deployment in public places using hardware and software which is interoperable and mobile. LoRa offers three transmitting frequency, 433/866/915 MHz depending on the country it is deployed. LoRa exploits variable error correction technique that improves the robustness of the transmitted signal at the expense of redundancy. LoRa modulation bit-rate \( DR \) is defined as [17]

\[
DR = SF \cdot \frac{2^{SF}}{BW} \cdot \frac{CR}{CR + 4}
\]  

Where the parameters are,

\( SF \) = spreading factor(7 . . . 12)  

\( BW \) = bandwidth (Hz)  

\( CR \) = coding rate (1 . . . 4)  

The Noise-figure of LoRa is given as [17]

\[
NF = -174 + 10\log_{10}(B) \text{ dBm}
\]  

Here, \( NF \) is noise figure and \( B \) is bandwidth.

The real-time hardware implementation details have been discussed in the next section.

A. Field Deployment

1) Transceiver Nodes: The transceiver node as shown in Fig. 2 consists of a custom designed PCB with Raspberry Pi 3, a 5 MP RPi cam Rev.1.3. The custom designed PCB as shown in Fig. 3 consists of Arduino Pro-mini board using ATMega328p micro-controller, working at 5V 16 MHz, Semtech’s SX1276 LoRa IC working at 866 MHz. A 1.5 dBi quarter wave monopole antenna is used for signal transmission. The Arduino board is responsible for all communication operations. A Raspberry Pi 3 B+ model with 64GB high speed micro-SD card is used specifically for the purpose of running the trained CNN model on the captured images. A 5MP RPi camera version 1.3 is used to capture the image of the person who has pressed the button. A push-button switch is connected which upon pressing notifies the Raspberry Pi to capture an image of the person who has pressed the node and also wakes up the arduino to send distress signal to the gateway. The components are put in a sturdy weather-proof box to protect from harsh weather conditions as the devices are to be attached to street light poles. A 1000mAh 3.7V lithium ion battery is connected to power the node and a charging circuit is used to charge the battery whenever there is power supply at the light poles. The charging IC used is TP4056 - Micro USB 5V 1A Lithium battery charger. For connecting the device to the light poles a 5V 1A wall adapter is used. The 1N4733A zener diode is used to ensure unidirectional flow of current from either battery or the adapter.

2) Gateway Device: We have designed a gateway device as shown in Fig 4 which consists of Raspberry-Pi 3 B+ model interfaced with Adafruit 32u4 LoRa module. The Adafruit
LoRa model houses a LoRa transceiver based on Semtech SX1276 LoRa modem running in the 866 MHz ISM band. A 1.5 dBi quarter wave monopole antenna is used similar to end node devices. A 4G-LTE modem is connected with the Raspberry Pi which logs the received node-id to the server and make it available online. Since this device should be continuously operating, it is wall powered using a 5V 1A power supply. There is a small power consumption penalty required for the gateway device since the nodes can sleep when inactive and thus, results in significant overall power saving. The casing is 3D printed available in-house.

3) Data logging unit: The data logging unit consists of a computer with MySQL server 1.0.11 running on it. The gateway device pushes the received node-id to the server. The server stores the node-id along with latitude, longitude values, status of the node and timestamps it. It also makes the page available online as shown in Fig. 5. This enables full retention of data for future references. The server is also responsible for sending a notification to the android app that we have developed.

4) Graphical User Interface (GUI): As shown in Fig 6 an android application has also been developed for providing a graphical interface using Android Studio 3.0. The android app by default shows markers on the map where the nodes are placed. When someone presses the button on a certain node, it shows a navigation route from the device towards that node for the concerned security personnel to reach there at once and assist the distressed person. This application is intended to be installed on the security personnel’s phone or an android TV to serve the purpose.

The proposed network has been deployed in Indian Institute of Technology Campus for testing and all the analysis has been done on the same.
The server is connected to the android app using a back-end PHP script. When the server receives a certain node-id, it compares it with the stored node-ids and retrieve the latitude and longitude values of that node so as to get the exact location on map. Then the app shows that particular node with a navigation route to reach the node from the current location of the host device as shown in Fig 6. So this app can be installed on an android TV with a large display of the map in a security cabin for a better view.

Since the end device is very appealing to any person they might press the button just for fun irrespective of any imminent danger. This could create a problem since the authority would be alarmed unnecessarily. So to stop the misuse of the system we have introduced a trained CNN model at the end device itself to eliminate the complicacy and delayed response of cloud computing. The trained model recognises the expression of the person from the captured image and sends distress signal for specific emotion like disgust, anger, sadness and neutral. This would help identify whether the person who pressed the button was really in danger or not; thus, reducing false alarms up to a certain extent and making the device more resilient. As Arduino Pro-mini is not powerful enough to do image processing on-the-go, we have included RaspberryPi at the end node to do the job.

C. CNN based Emotion Recognition

We created a simple CNN architecture for facial expression detection and classification is shown in Fig. 7. The proposed CNN model comprises of two convolution layers, one max pooling layers, one average pooling layer and two Fully Connected (FC) layers. All layers use the activation function. Rectified linear unit (ReLU) is used as an activation function with he-normal[] as the kernel initializer for all the layers, and softmax function is employed for the final classification layer. Dropout of 0.25 and 0.4 is applied prior to FC 1 and FC 2 layers to reduce the effects of over-fitting. Ahead of training the proposed network, all the images are resized to $48 \times 48$ resolution, and the pixel value of the images are normalized [19].

To obtain the discriminative representation of the images, small convolution feature maps are employed which help in breaking the linearity of the network and reduce the computational complexity. The stride and pad length is set to one pixel for each convolution layer and a stride length of two pixels is applied to the max-pooling layer. This helps in extracting the critical features such as edges and with minimal feature vector size. In the first layer, 10 feature maps with kernel size $3 \times 3$ are employed for convolution layers while kernel with size $2 \times 2$ is used for average-pooling. In the second layer, 32 convolution feature maps with kernel size of $5 \times 5$ are used for convolution layers, and a kernel size of $2 \times 2$ is employed for max pooling layer. Zero-padding has been used inbetween the convolution layers as it improves accuracy by keeping information at borders. In first fully-connected layer 256 units are used, whereas in second fully connected layers 128 units are used. Softmax function is used as the final classification layer. We have used keras in-built categorical cross-entropy as the loss function. We have also used Adam optimizer [18] since it is an efficient stochastic optimization method that only requires first-order gradients with little memory requirement. The method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients.

III. EVALUATION OF THE PROPOSED NETWORK

We have evaluated our proposed low-power, low-cost IoT Network by measuring power consumption, operating range, algorithm accuracy and cost.

<table>
<thead>
<tr>
<th>Device</th>
<th>days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Safer smart jewellery</td>
<td>14</td>
</tr>
<tr>
<td>Athena device 3</td>
<td>14</td>
</tr>
<tr>
<td>Nimb device</td>
<td>14</td>
</tr>
<tr>
<td><strong>Our end-node</strong></td>
<td><strong>30</strong></td>
</tr>
</tbody>
</table>

TABLE I: Battery life analysis of proposed network

A. Power Analysis

For evaluating power consumption we have considered the current consumption of the end node during transmission and sleep instant. As mentioned before, the end node is programmed to be in sleep mode when not using and the push button upon pressing activates the arduino. So we measure current consumption at sleep $I_{sleep}$ and transmission instant $I_{transmission}$.

The end node has $I_{sleep}$ of 30.2 mA and $I_{transmission}$ of 47 mA. We were able to achieve such a low current consumption by removing the on-board LED and the power regulator of the Arduino pro-mini. The onborad HDMI and LEDs were turned off in the Raspberry Pi thus reducing current consumption up to 30 mA. We also programmed to turn off the brown-out detector and watchdog-timer while in sleep mode. A comparison of our end node battery life expectancy with some of the commercially available devices has been shown in table I. Thus, it consumes significantly less power than available wearable devices. This leads to enhanced battery life. This justifies the low power nature of the end node.

B. Operating Range

In range evaluation, we observed the Received Signal Strength Indicator (RSSI) values of the transmitted packet.
from the end node to the gateway device. The end node is placed at different locations across the IITH campus and the gateway device is placed on the rooftop of Academic Block A which is around 50 meters high. The result is shown as a MATLAB plot in Fig 8. It is observed that the RSSI value decreases as distance is increased which is expected. When the node was placed 1.5 km away from the gateway device there was still reception and acknowledgement. We used noise figure as the metric to justify our claim that the signal is still well above the specified noise floor which is \( 122.2 \text{ dBm} \) using equation (2) and setting \( B = 125 \text{kHz} \). This helps us conclude that the nodes and gateway can be as far as 1.5 km apart and still operate flawlessly which justifies our idea of large-scale deployment. The sudden distortions is due to the presence of buildings and trees in the campus.

Thus, our network can be deployed in a citywide scape. But the penetration of signal through buildings and structures might be a problem in LoRa RF, which can be further solved with more concentrated gateways. Still, this would provide better scalability in terms of range than existing solutions like WiFi and bluetooth.

### C. Accuracy

We evaluate our CNN architecture for expression recognition on the well established JAFFE dataset [19]. The database contains 213 images of 7 facial expressions (6 basic facial expressions + 1 neutral) posed by 10 Japanese female models. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects. For each combination of subject and facial expression, there are between two and four examples present in the dataset. The sample images of each emotion are shown in Fig. 9. We have used two letter code for emotion labeling, for example, NE for neutral, AN for anger etc. To obtain sufficient samples for training the proposed CNN, data augmentation techniques such as image transformation (horizontal flip, vertical flip, image rotation, shear range and zoom range) are employed for generating new images. By using the image transformation, a total of 2048 images were obtained for training the network. We have selected \( 48 \times 48 \) patches of each image corresponding to their labelled emotions. We have divided the dataset into a 80 – 20% split for training and testing respectively. The training is performed for 50 epochs on cross-validation with the training and testing dataset chosen randomly each time from the complete dataset. Batch size of 64 is selected to decrease the computation time while training.

We trained the model for 50 epochs and got an accuracy of 86%. The plot of accuracy as a function of epochs has been shown in Fig. 10. The blue line and the orange line corresponds to training and validation accuracy respectively. It is observed that there is little difference between training and testing accuracy which shows that our model is not overfitting. The CNN model was created with Keras library of Python 2.7 using tensorflow backend. As shown in Fig 11 we ran the saved model on Raspberry Pi and it recognised the expressions correctly with 2 seconds of inference time. Thus, our model is suitable for emotion recognition at the end node.

### D. Cost Estimation

We estimated the cost of the end node and gateway device as shown in Table II and Table III

It is observed from Table II that our end node costs \( 51 \) which is substantially lower than commercially available AI
TABLE I: Cost Estimation of gateway device

<table>
<thead>
<tr>
<th>Components</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino Pro mini</td>
<td>3.55</td>
</tr>
<tr>
<td>Sx1276 LoRa IC and Antenna</td>
<td>7.565</td>
</tr>
<tr>
<td>Raspberry Pi 3</td>
<td>3.55</td>
</tr>
<tr>
<td>Rpi cam</td>
<td>1.25</td>
</tr>
<tr>
<td>Charging IC TP4056</td>
<td>0.565</td>
</tr>
<tr>
<td>1000mAh Li Ion Battery</td>
<td>1.68</td>
</tr>
<tr>
<td>Weather-proof box</td>
<td>25</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>51.45</strong></td>
</tr>
</tbody>
</table>

TABLE II: Cost Estimation of end node

<table>
<thead>
<tr>
<th>Components</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino Pro mini</td>
<td>3.55</td>
</tr>
<tr>
<td>Sx1276 LoRa IC and Antenna</td>
<td>7.565</td>
</tr>
<tr>
<td>Raspberry Pi 3</td>
<td>3.55</td>
</tr>
<tr>
<td>4G LTE modem</td>
<td>305</td>
</tr>
<tr>
<td>3D Printed box</td>
<td>1.55</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>77.565</strong></td>
</tr>
</tbody>
</table>

IV. CONCLUSION

We have proposed, implemented and evaluated a long-range IoT network for citizen security. We designed a custom PCB with Arduino and LoRa IC which acts as our end node a Raspberry-Pi based 4G gateway. A data logging server is created which logs all incoming node-ids from the gateway and is made available online. We have also developed an android application which shows which node-id is pressed in real time and navigates the concerned security personnel to that node. We have incorporated a simple low complex CNN based emotion recognition algorithm at the end node itself to minimize false alarms. Cost estimation of our end node and gateway device has also been provided which shows that our proposed network is of significantly low cost compared to existing solutions. Since all communication interface is LoRa RF based, our proposed network supports long-range which is also shown graphically.

REFERENCES


