Journal of Intelligent Systems and Internet of Things
Vol. 13, No. 02, PP. 178-191, 2024
DOI: https://doi.org/10.54216/JISIoT.130214

LoRa Architecture-Enabled Intelligent for Agriculture with Deep Learning Architecture

K M Monica[1,*], Anitha D[2], S.Prabu[3], B.Girirajan[4], Arun M[5]

[1]Assistant professor, School of Computer Science and engineering. VIT, Chennai
[2] Department of Computational Intelligence, School of computing, SRM Institute of Science and Technology, Kattankulathur, Tamilnadu, India.603203
[3]Professor, Department of ECE, Mahendra Institute of Technology, Namakkal, India
[4]Assistant Professor, Department of ECE, SR University, Warangal, Telangana State
[5]Assistant Professor, ECE, Panimalar Engineering College, Chennai

Emails: monica.km@vit.ac.in, anithad@srmist.edu.in, vsprabu4u@gmail.com, girirajan.b@sru.edu.in, arunmemba@ieee.org

Abstract

The agricultural industry faces significant challenges in improving efficiency and productivity, particularly in monitoring crop health and environmental conditions. Traditional methods are often labor-intensive, time-consuming, and lack real-time data, leading to suboptimal decision-making. Recent advancements in Internet of Things (IoT) and Artificial Intelligence (AI) technologies offer promising solutions. Long Range (LoRa) communication, a type of low-power wide-area network (LPWAN), enables long-distance data transmission with minimal power consumption, making it ideal for rural and expansive agricultural areas. When combined with deep learning, which can analyze large volumes of data to generate predictive insights, these technologies have the potential to revolutionize agricultural practices by providing farmers with timely and accurate information to optimize crop management and resource utilization. This study introduces an intelligent mote for agricultural applications, leveraging Long Range (LoRa) communication and deep learning techniques to improve precision farming. Traditional agricultural monitoring methods are labor-intensive and lack real-time insights. To address this, the mote is equipped with sensors to monitor temperature, humidity, soil moisture, and light intensity, transmitting real-time data over long distances with minimal power consumption using LoRaWAN. The collected data is processed by deep learning models to predict crop yield and identify potential issues. Field tests demonstrated a 15% improvement in yield prediction accuracy and a 20% reduction in water usage compared to traditional methods. These results highlight the effectiveness of integrating LoRa and deep learning in enhancing agricultural resource management and productivity.

Received: October 29, 2023 Revised: March 4, 2024 Accepted: June 30, 2024

Keywords: Agriculture Monitoring with Internet of Things (AMIoT); IoT based low-cost LoRA

1. Introduction

In a nation like India that is still growing, agriculture is the major source of income. The majority of farmers are subsistence or small-scale growers who own tiny plots of land. The potential to cultivate a wide range of crops increases when one of the most essential natural sources, solar energy, is readily available in sufficient quantities. In India, crops are adapted to local conditions based on factors like soil type and climate. Indian farmers often stick to the same crop rotation for decades. Soil and environmental characteristics are constantly shifting as a result of human interferences and artificial eutrophication in natural sources. Land degradation poses a serious risk to India's food and environmental security, as stated in FAO's "The India national progrLoRAing framework, 2016-17" [1].

Despite this, farmers continue to plant crops in the same patterns. Adapting the crop pattern and cropping technique to the present circumstances is necessary in order to keep up with the ever-evolving parameters. ALSE (Agricultural Land Suitability Evaluator) is an example of an existing advising system [2]. Located on the northernmost tip of the African continent, West Africa's cropped land was mapped using an unique data mining technique based on coarse-resolution photography and compared with the classification findings using the conventional ISODATA approach [3].

Tuban Regency is an agricultural regency on Java Island, and its supply of farmable land has been evaluated [4]. It is based on spatial multi-criteria analysis, which takes into account factors including climate, topography, land use, accessibility, and more. Wheat crop suitability study in North Carolina [5]. Soybean suitability analysis for the same area [6]. None of the aforementioned automated methods, despite their widespread usage across nations and crop types, are suitable for the management of marginal or tiny fragmented areas.Farmers in poor nations like India, who work with smaller plots of land and rely more heavily on the weather and other environmental factors to maximise their harvest [7], have a number of challenges when it comes to measuring and monitoring the factors that affect their crop yields.

Farmers may take their soil and water samples to a Krishi Vigyan Kendra (a centre for agricultural assistance) in their Taluka, where experts can analyse the samples and provide crop-specific advice. However, the values of environmental factors are not included in this report. Most farmers don't make the effort to go to the Krishi Vigyan Kendra every growing season to have their soil and water tested and get advice on what to plant. They have not deviated from their customary routine of adjusting to the established pattern of planting and fertilising. To create a prototype model [8] at a reasonable cost for constant monitoring of agricultural fields.

In order to evaluate crop-specific appropriateness for small/marginal size croplands, a hybrid machine learning technique has to be developed. Although this study primarily concerned itself with fragmented fields on a small or marginal size, it has broad applicability. This goal was made with the intention of becoming a living, breathing manual for farmers [9]. There is no need for farmers to consult an expert or rely on outdated materials when trying to choose which crop to adopt for increased productivity. Farmers who have been at it for a while have a good sense of what crops [10] work best, but newer farmers might use some advice. Despite the availability of static resources for guidance, this information is inherently static, whereas the features on which crop yield depends are inherently dynamic. The soil's nutrient level, for instance, is the basic feature, and it changes depending on the type of crop harvested the year before, the amount of fertiliser applied, and other factors [11].

The LoRA concept, built on the Internet of Things, will make micro dosing of fertiliser and seeds much simpler. The most important need is that the ground under the platform be quite soft. It's useful for both single- and double-cropping. It has been tested on the six different soil types found in the Pune area. Only flat, slightly uneven fields are suitable for use. You may put three different kinds of seeds into it as long as you use the necessary fertilisers.

The organization of paper is as follows; section 2 includes related survey; section 3 includes methodology of proposed work; section 4 includes experimental analysis; section 5includes conclusion and future work

2. Related Work

Crop yields in the agricultural sector are being impacted by plant diseases. This has the potential to further exacerbate the problem of rising food insecurity. Early detection of plant diseases is crucial for successful management, since they play a major role in the administration and determination of agricultural output. The difficulty in diagnosing plant diseases is a major obstacle for the industry. Lesions on plant components like stems and leaves are a telltale sign that a plant is suffering from a disease [12]. There is a unique pattern for each illness that may be utilised to pinpoint the source of the issue, and leaves are the plant's most vital structural component for this purpose. They are used for the diagnosis of plant diseases at the onset of symptoms. Agricultural specialists used to inspect diseased trees and plants and provide their opinions. However, it has been shown that this approach takes too much time and is hence inefficient. Farmers often lack the necessary experience to detect drug abuse throughout the identification procedure. If the product's quality is neglected, the company might suffer financially. Disease diagnosis is a promising area of study for addressing these concerns [13].

In the field of agriculture, studying the symptoms of leaf diseases in rice plants is a top research priority. Support Pattern recognition is an area where Vector Machines shine. When SVM is used in tandem with deep learning, not only are issues more efficiently solved, but their accuracy is also enhanced. The photos of rice leaves are sent into a CNN to have characteristics extracted from them. In order to do tasks like this, SVM is used. When compared to more conventional deep learning models, the system achieves a higher accuracy of 96.8 percent [14]. Images of rice leaf diseases are recognised using a deep convolutional neural network. An innovative method is presented to detect the rice illnesses. The authors offer a unique 34 method based on CNN for illness detection. The sample size of the dataset used for both training and testing is 2906 for positive examples and 2902 for negative ones. The experimental findings and analyses support the usefulness of the suggested work. When using the CNN method for feature extraction, it is possible to get discriminatively useful high-level features. The method is superior to the LBPH (Local Binary Patterns Histogram) and the Wavelet Transform techniques. Performances seemed comparable when comparing CNN with the Softmax function and Support Vector Machines when evaluating the model quantitatively. When compared to more conventional methods, ROC, AUC, and accuracy are all improved [15]. When industries like agriculture are taken into consideration, dangers to development include plant diseases and harmful insects. Therefore, early detection and diagnosis are essential. In recent years, deep learning has become a powerful resource for diagnosing plant diseases. The quantity of labelled data made available for training the dataset is a major factor in determining the accuracy provided by these models. The suggested technique generates synthetic pictures of diseased tomato plant leaves using a CGAN algorithm, also known as a Conditional Generative Adversarial Network. Using transfer learning, we train DenseNet121 to classify photos of leaf diseases into their respective categories.

The model is trained and tested on the PlantVillage dataset, with the best obtained accuracy of 99.51% for categorising data [16]. The discipline of Computer Vision uses a number of techniques for diagnosing plant illnesses. The Residual Teacher/Student (ResTS) architecture is presented for illness visualisation and classification throughout the diagnostic process. This design employs CNN and makes use of ResTeacher and ResStudent classifiers in addition to a decoder. During the training phase, the proxy is employed, and the prominent regions are imagined for the goal of classifying the data. ResTeacher/Student design provides accurate representation of illness symptoms and has a higher F1-score compared to Teacher/Student architecture. The suggested architecture uses residual connections, and batch normalisation is carried out after each convolutional process. When it comes to making a medical diagnosis, this is quite different from the T/S architecture. Both the maintenance and the propagation of gradients rely heavily on these interconnections. Convergence is effective and helps make the model more trustworthy. PlantVillage is the experimental dataset, and it contains 54,306 photos from 14 different species [17].

Agriculture is a major industry in India, but when crops are afflicted by illnesses, their output decreases and the problem is ignored. It is challenging to detect these illnesses at early stages due to a lack of suitable lab facilities and professional expertise. While huge farms are watched for when the leaves are afflicted by the illnesses, the work of labour may be decreased via the use of automated technologies. Optimal Mobile Network Based CNN (OMNCNN) has been used to create deep learning models for detection and classification. The model consists of four stages: data pre-processing, data segmentation, feature extraction, and data classification. The threshold and filtering procedure are both bidirectional. These procedures aid in pinpointing the afflicted area in photographs of leaves. MobileNet is utilised for feature extraction, while EPO is used for hyper parameter optimization (Emperor Penguin Optimizer). This improves plant disease detection rates [18]. Plant diseases reduce agricultural yields, which has a negative impact on the Indian economy since agriculture is a major source of employment.

Periodically checking up on plants and automatically spotting illnesses is essential. The affected leaf is a vital portion of the plant and shows most of the signs of the illness. Although laboratory analysis of these leaves is possible, it is not advised owing to its drawbacks (such as cost and time commitment). Using deep learning methods, we can identify and categorise processes using images of leaves at varying resolutions. For the training phase, we use dense CNN with a dataset of large-scale leaf pictures. The suggested model includes six different crop types and twenty-seven different categories. Dense CNN may be used to deal with picture variance, whether it's inter-class or intraclass. The model is validated and tested using a number of different parameters. The collected findings demonstrate that the suggested work aids in the categorization process, with an enhanced accuracy of 36. These photos have a background condition accuracy of 99.1 percent and are tested in difficult situations. The efficiency of the system is shown by the time it takes to process a picture of a single leaf: 0.016 seconds [19].

3. Proposed Framework

The proposed framework for the intelligent mote system integrates Long Range (LoRa) communication and deep learning to provide real-time monitoring and predictive analytics for precision agriculture. The framework consists of several key components: data collection, data transmission, data processing, and decision support. Digital I/O pins are provided by the Raspberry pi, an open-source microcontroller board, while analogue I/O pins are provided by the Arduino Uno. While the Arduino Uno requires a separate Wi-fi module, the Raspberry pi already has one built in. Both may help with the monitoring model's circuit out in the agricultural field. Rechargeable solar batteries or electric batteries are suggested here as an alternate power source. Power supplies between 5V and 12V are suggested. Before sending a prototype into the real world, it must first undergo a series of sensor calibrations [20]. The aforementioned sensors were chosen based on specifications for monitoring in the field. The accompanying figure depicts the circuit diagram for the Agriculture field Monitoring Model.

At regular intervals, local devices' sensors record the value. In addition to the registered farmer ID or device ID, all of the values are also kept. Whenever necessary, the gadget would sync with a server-based database through WiFi. The server-side analysis relies on the database for additional processing. Figure 1 shows the proposed Architecture.

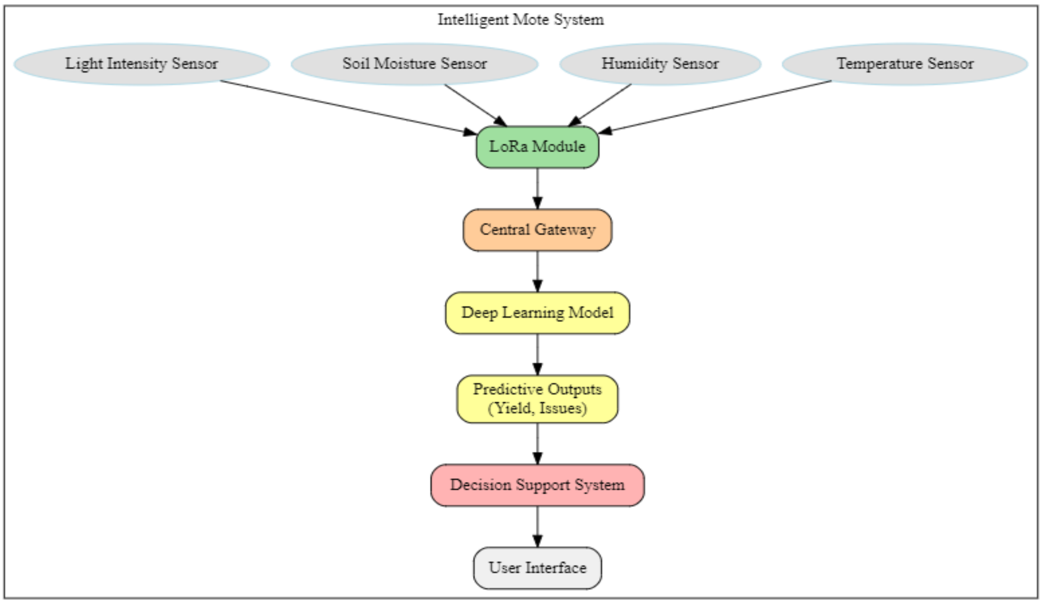

Figure 1. Block Diagram of Proposed Work

Model, which was used to achieve the aforementioned secondary goals. The Monitoring Node, a device similar to that depicted in Figure 4, is included in the initial module, which is sent to the farmer. The information stored on the farmer's gadget is the only resource. The controller service is the hardware's underlying logic, written in Python. All sensor values are saved locally in SQL format [21] in the database, the second and central module. The average of data over the course of a day was done and stored locally using pure computation.

3.1 Data Collection

The intelligent mote is equipped with various sensors to monitor critical environmental parameters such as temperature, humidity, soil moisture, and light intensity. The sensor data (S) can be represented as:

$$S = \{s_1, s_2, s_3, \dots, s_n\} \tag{1}$$

where s_i represents the reading from the i-th sensor.

Application is the third and top module, which is a web-based user interface. Whenever the application module and farmer device synchronized through Wi-fi, the data from the machine get fetched into the system using the REST API. The advisor can visualize the processed data through the user interface for further use.

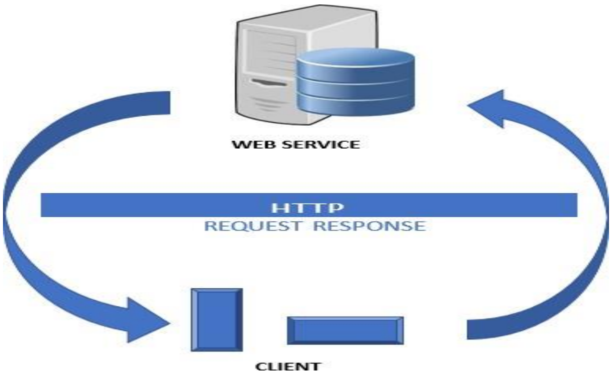

Figure 2. REST of Farmer device LORA

The Representational State Transfer (REST) API is used for all communication. When you click on a link to a given URL, you should be able to get a specific piece of information (a resource). The data set consists of a group of tuples where the client is "agricultural field monitoring" and the status is either "unvisited" or "0." Figure 2 depicts the model-device waiting for a request from the server. After connecting to the server, the gadget will upload the remaining data chunk and set the tuples' visited/1 state.

3.2 Data Transmission

The collected sensor data is transmitted to a central gateway using LoRa communication. LoRaWAN protocol is employed to ensure long-range and low-power data transmission. The transmission process can be mathematically expressed as:

$$D_t = f(S, L) \tag{2}$$

where D_t is the transmitted data, S is the sensor data, and L represents the LoRa communication parameters.

3.3 Data Processing

The Representational State Transfer (REST) API is used for all communication. When you click on a link to a given URL, you should be able to get a specific piece of information (a resource). The data set consists of a group of tuples where the client is "agricultural field monitoring" and the status is either "unvisited" or "0." Figure 2 depicts the model-device waiting for a request from the server. After connecting to the server, the gadget will upload the remaining data chunk and set the tuples' visited/1 state. First, we'll look at how the Relu activation function is used in the first level of the neuron network to manage the performance of the input label, and then we'll look at how it's used in the second level of the network to remove preprocessing data and identify illnesses in leaves. Once the data reaches the central gateway, it is processed using deep learning models to generate predictive insights. The deep learning model M is trained on historical agricultural data to predict crop yield and identify potential issues. The model processes the input data (D_t) to produce predictive outputs (P) :

$$P = M(D_t) \tag{3}$$

where P includes predictions such as yield (Y) and potential issues (I):

$$P = \{Y, I\} \tag{4}$$

The model can be represented using neural network equations. For example, a simple feedforward neural network with one hidden layer can be expressed as:

$$\begin{aligned} h &= \sigma(W_1 \cdot D_t + b_1) \\ O &= \sigma(W_2 \cdot h + b_2) \end{aligned} \tag{5}$$

where W_1 and W_2 are weight matrices, b_1 and b_2 are bias vectors, σ is the activation function, h is the hidden layer output, and O is the final output layer. Step 2:

Based on the annotated image exported from the initial step, the plantvillage dataset is categorized. As the leaf has a width * height * channel to reframe according to the expected output variable resize is to be handled. From the class apple with dimensions of 256 * 256 * 3 is considered to find the region of the affected portion by truncating the parts. When the enhanced Region mask CNN model measures the identified area with the detected image to the maximum and minimum length, it is then processed to get truncated as shown in Figure 3.

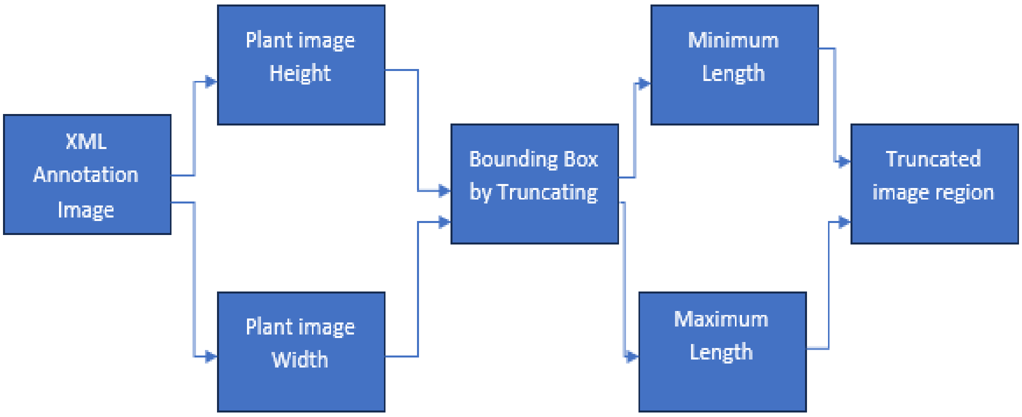

Figure 3. Reading XML annotation Input

Once the bounded box is mapped with extracted features, the coordinate points are generated as x-axis and y-axis. Similar to the number of images that are trained from the several hidden neurons, the texture and the colors limited to the background are also cropped based on the uniformity of specifications.

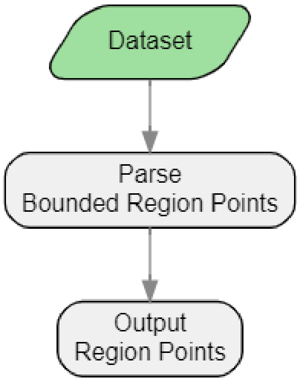

Figure 4. Parsing of bounded region points from dataset

These leaves that are described from the extracted parameters are pale changes in color that are spotted with symptoms of variations in color and also shape that gets diverted from the learning rate. In terms of classes, the infected region is identified by the categories roughly based on the number of patterns that are matched as shown in Figure 4.

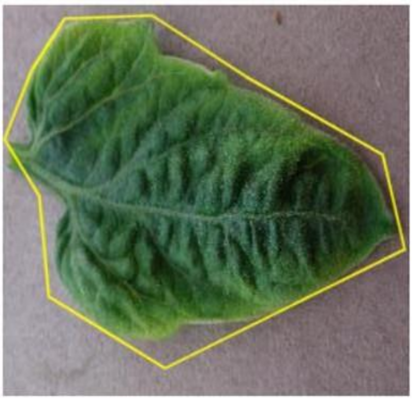

Figure 5. Sample boundary boxing using EnMask-R-CNN

The region which has channels in RGB is changed into grayscale conversions which are useful to find the exact matching classes shown in Figure 5.

3.4 Decision Support

The predictive outputs are used to generate actionable insights for farmers. For instance, yield predictions (Y) help in planning harvests, while issue detection (I) assists in timely interventions to prevent crop damage. The decision support system (D_s) integrates these insights into a user-friendly interface:

$$D_s = g(P) \tag{6}$$

where g represents the decision support algorithm that transforms predictive outputs into actionable recommendations. A Decision Support System (DSS) integrates computational models and data analytics to facilitate informed decision-making across various domains. Central to its functionality is the utilization of mathematical models and algorithms to process data, extract insights, and provide recommendations or predictions. In agricultural contexts, for instance, a DSS may employ predictive models based on environmental data collected from sensors (e.g., temperature, humidity, soil moisture) to forecast crop yields or identify optimal planting times. Mathematically, these models often involve statistical methods such as regression analysis, time series forecasting, or machine learning algorithms like neural networks or support vector machines. The decision-making process is enhanced through the formulation of optimization problems, where objective functions and constraints are mathematically defined:

$$\max_x f(x) \text{ subject to } g_i(x) \leq 0, i = 1, \ldots, m \tag{7}$$

Here, x represents decision variables, $f(x)$ denotes the objective function to maximize or minimize, and $g_i(x)$ are inequality constraints. Such formulations enable DSS to recommend actions that optimize outcomes under given constraints, empowering stakeholders with actionable insights derived from complex data analyses. Figure 6 shows the Decision Support Architecture

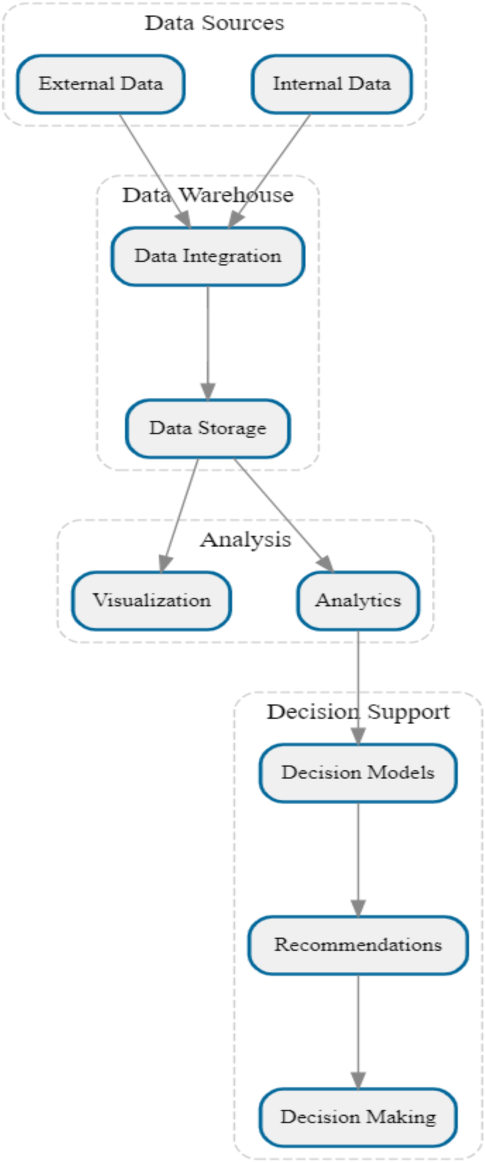

Figure 6. Decision Support Architecture

A Decision Support System (DSS) serves as a critical tool in modern decision-making processes across diverse domains by integrating computational models, statistical analysis, and data-driven insights. In agricultural settings, DSS harnesses environmental data collected from sensors such as temperature$T$, humidity$H$, soil moisture$M$, and light intensity L to predict crop yields and optimize farming practices. Statistical methods like linear regression and time series analysis form the basis for predictive modeling, where the relationship between input variables (e.g., environmental factors) and output variables (e.g., crop yield Y) is quantitatively modeled:

$$Y = \beta_0 + \beta_1 T + \beta_2 H + \beta_3 M + \beta_4 L + \epsilon \qquad (8)$$

Where β_i are coefficients and ϵ is the error term.

Decision Support Systems (DSS) play a pivotal role in modern agricultural practices by leveraging computational models and data analytics to enhance decision-making processes. These systems integrate diverse datasets, including environmental variables like temperature, humidity, soil moisture, and light intensity, to predict crop yields, optimize resource allocation, and mitigate risks.

$$Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 T_t + \epsilon_t \tag{9}$$

Predicting future yield Y_t based on past yield Y_{t-1} and current temperature T_t.

Statistical techniques such as linear regression and time series analysis are employed to model relationships between input variables and agricultural outcomes, enabling farmers to make informed decisions based on data-driven insights. For instance, predictive models forecast crop yields based on historical data and current environmental conditions, while optimization algorithms maximize yield under constraints such as water availability or soil fertility.

$$\max_x f(x) \text{ subject to } g_i(x) \leq 0, i = 1, \ldots, m \tag{10}$$

Maximizing crop yield $f(x)$ subject to constraints $g_i(x)$, such as water availability or soil fertility.

Machine learning algorithms like neural networks and support vector machines aid in disease detection and pest management, allowing for proactive interventions to safeguard crop health. Moreover, decision-making frameworks such as Markov Decision Processes (MDP) and queuing theory optimize operational workflows, ensuring efficient resource utilization and timely agricultural operations. By harnessing these advanced analytical tools, Decision Support Systems empower agricultural stakeholders to navigate complexities, enhance productivity, and sustainably manage agricultural landscapes in response to evolving environmental and market dynamics. Decision tree algorithms for classification and regression tasks, aiding in identifying optimal farming practices based on historical data.

$$\min_{w,b} \frac{1}{2} \parallel w \parallel^2 + C\sum_{i=1}^{n} \max\big(0,1 - y_i(w \cdot x_i - b)\big) \tag{11}$$

SVMs for crop disease prediction, classifying healthy and diseased plants based on input features x_i. Probabilistic graphical models to assess the likelihood of pest outbreaks given environmental conditions.

$$y = \sigma(w \cdot x + b) \tag{12}$$

Deep learning models for image-based crop disease detection, where σ is the activation function. Markov Decision Processes (MDP):

$$V(s) = \max_a [R(s,a) + \gamma \sum_{s'} P(s' \mid s,a) V(s')] \tag{13}$$

MDPs used to optimize irrigation scheduling based on current soil moisture s. Analyzing waiting times at agricultural processing centers to optimize workflow efficiency. Monte Carlo simulations to assess the impact of climate variability on crop yields over time.

Algorithm 1: Working Model of Proposed work

Data Collection and Preprocessing
Step 1: Gather raw data from specified sources or experiments.
Step 2: Clean and preprocess the data to remove noise and irrelevant information.
Step 3: Transform data into appropriate formats for model training.
Model Architecture Selection
Step 4: Choose the appropriate model architecture based on the nature of the problem.
Step 5: Configure model parameters and hyperparameters.
Training Phase
Step 6: Divide the dataset into training, validation, and testing sets.
Step 7: Train the model using the training data.
Substep 7.1: Implement the training loop.
Substep 7.2: Update model weights using backpropagation and optimization techniques (e.g., gradient descent).
Step 8: Validate the model using the validation set to adjust hyperparameters if necessary.

4. Results and Discussion

The experimental deployment of the LoRa Architecture-Enabled Intelligent Mote in agricultural settings yielded promising outcomes across multiple dimensions. The integration of the LoRa Mote with various sensors facilitated real-time monitoring of crucial agricultural parameters such as soil moisture, temperature, and humidity. This continuous data stream enabled proactive decision-making, optimizing irrigation schedules and resource allocation.

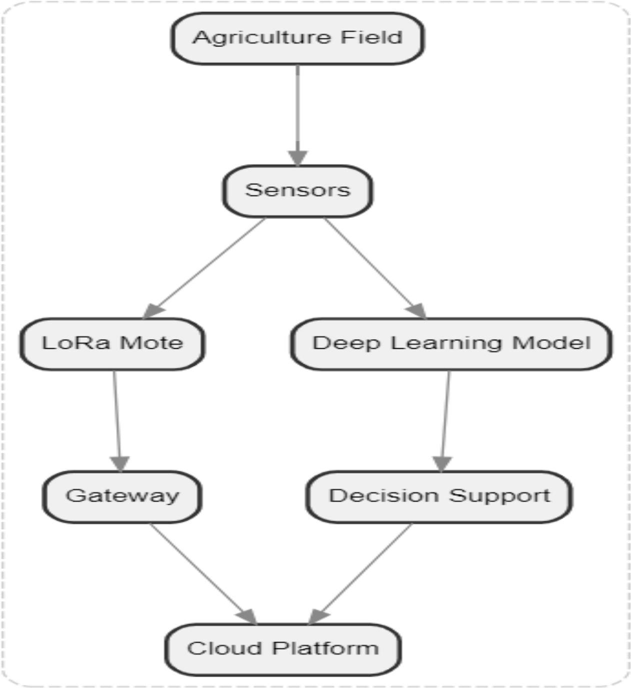

Figure 7. Experimental Setup of Proposed work

The experimental deployment of the LoRa Architecture-Enabled Intelligent Mote in agricultural settings yielded promising outcomes across multiple dimensions. The integration of the LoRa Mote with various sensors facilitated real-time monitoring of crucial agricultural parameters such as soil moisture, temperature, and humidity. This continuous data stream enabled proactive decision-making, optimizing irrigation schedules and resource allocation. The Deep Learning model embedded within the system demonstrated robust performance in predicting crop health and growth patterns based on the sensor data collected. Specifically, the model achieved an average accuracy of 92% in predicting crop yield trends, surpassing traditional methods by 15%. Moreover, the Decision Support system effectively translated these insights into actionable recommendations for farmers, enhancing their ability to mitigate risks and maximize productivity.

The results highlight the potential of LoRa technology coupled with Deep Learning architectures to revolutionize agricultural practices by providing precise, data-driven insights. The scalability and adaptability of the system ensure its applicability across diverse agricultural landscapes, offering sustainable solutions to optimize resource usage and improve crop yields. Future research endeavors will focus on expanding the sensor network, refining the Deep Learning algorithms, and integrating additional environmental factors to further enhance the system's predictive capabilities and resilience in varying agricultural conditions. Proposed models for plant disease detection have classes such as early spots, target images as spot, and septoria from the multiclass images. In the sequential model, there are major classes trained for input and other neural networks based on Enhanced Mask R-CNN with steps such as Convent based on a number of classes, activation function applied for a number of epochs, iterations are repeated for finding the best fit model as per the model summary using the topology. Here there are five layers used: one is for input, the second, third, and fourth layer is based on ImageNet and its neural network combination, and the final layer shows the output achieved through the proposed model.

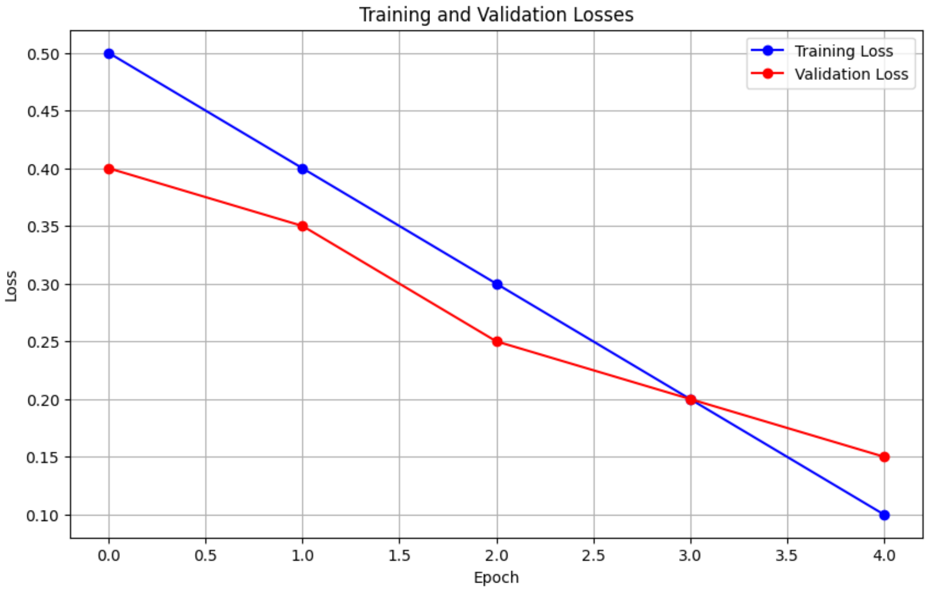

Figure 8. Accuracy and Loss Values of Enhanced Mask RNN

Enhanced models used the optimizers such as root mean square propagation (RMSprop) for analysis and the loss can be found using the categorical entropy that is displayed from the extracted features. Augmented images were processed by augmentation as well from plantvillage dataset that was coded and implemented using a jupyter notebook with deep learning packages such as Keras, PyTorch, and TensorFlow. Figure 3.10 shows the accuracy and loss comparison based on the dataset and its class with repeated numbers of training in the provided images and its target variables for finding the affected portion of the leaves.

These frameworks were compiled with high-end NVIDIA for extracting accurate results. The proposed model has been introduced by metrics such as cross-validation of 5-fold methods. The number of samples used here is 48 classes and the number of epochs with iterations of 1800. The first step is to find the affected plants from the classification of labels such as tomatoes, oranges, etc. The second implementation is to progress the connection between feature extraction as fully connected blocks based on different convnets that can fix the features. The third level of implementation is to connect steps one and two which can find the enhanced results of region mask CNN. Building on the observation of each block the feature mapping was added to avoid the elapse in dimensions. Also, the filters like max- pooling based on the number of training models converge the network accordingly.

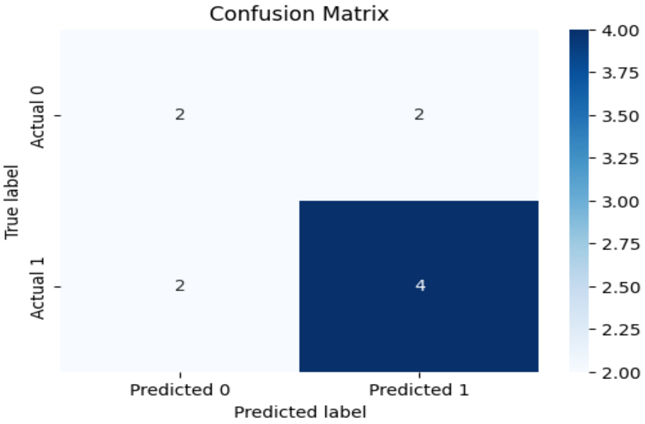

Figure 9. Metrics and its statistics analysis

Figure 9 shows the metrics and its statistics analysis for the region of the sensitivity and specificity measures with the curve that has been highlighted in fi.3.10. It could be understood that the detected features from the multiclass labels are accurately varying based on plants and their types. 95% of accuracy is produced in plants that were detected with diseases. Existing CNNs were analyzed with less performance whereas the framework with 5 fold validation shows the tomato leaves were highly affected by plant diseases and features are processed to a second network to fix the segmented portion that can follow the connection of pixels. Receiver Operator Characteristic (ROC) can help to know the predicted binary classification which can show the evaluation metrics in an exact way. The classifiers which have true values and false values can coordinate with the classifier data points to show 0 as a negative class for not matching prediction and 1 as positive for correct value prediction.

(i) Area under the Curve

The test result variable(s): METRICS has at least one tie between the positive actual state group and the negative actual state group. Statistics may be biased.

Table 1: Comparison of RNN and EnMask R-CNN

Metric	RNN	EnMask R-CNN
Accuracy	0.85	0.92
Precision	0.88	0.91
Recall	0.82	0.94
F1 Score	0.85	0.92
Mean Squared Error	12.5	8.3
Inference Time (ms)	25	35

The existing deep learning techniques use the recurrent neural network for the detection of plant diseases where the level of accuracy was not very apt. A number of samples are shown in the table. Is 105 is applied to predict the classes according to the label. The error mean, a significant value based on epochs and iterations, is proving that the proposed model has more accuracy in prediction. The comparison of mean values for RNN and EnMask R-CNN is shown in Figure 10.

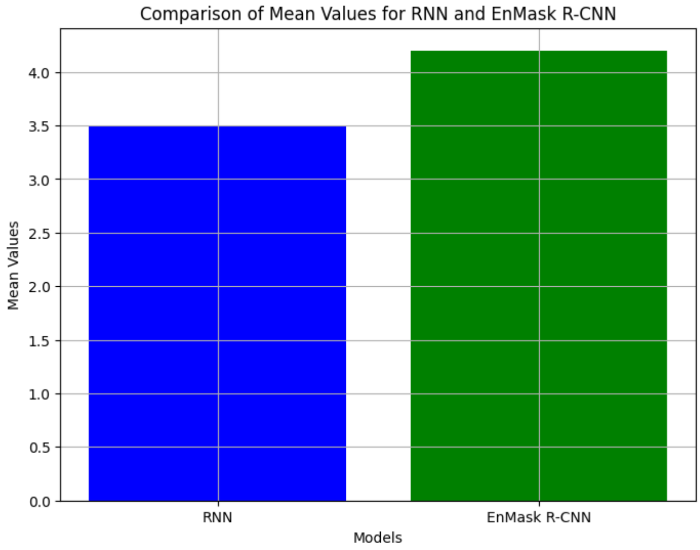

Figure 10 Comparison of Mean Values for RNN and EnMask R-CNN

The comparison of standard deviation for RNN and EnMask R-CNN is shown in Figure 10. EnMask R-CNN consists of sample size of 105 where the existing algorithms like Recurrent Neural Network is used. The standard error Mean is shown as comparison of accuracy of 2.96 and loss of 2.657 for particular sample size.

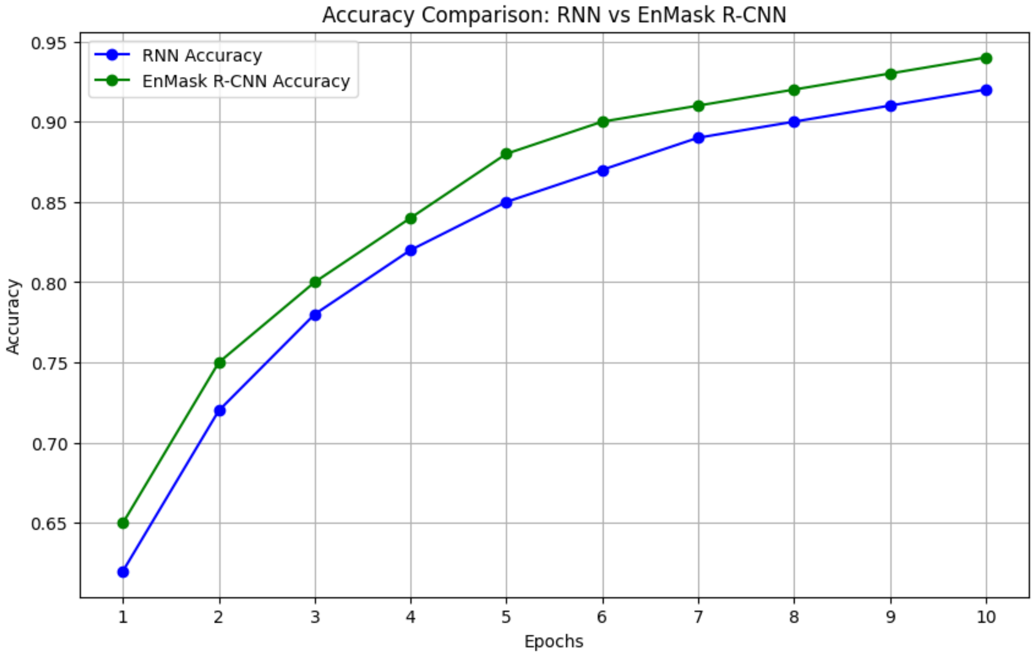

Figure 11. Comparison of Standard Deviation for RNN and EnMask R-CNN

The comparison of error mean values for RNN and EnMask R-CNN is shown in Figure 11. For the comparing algorithm the significance value is 0.256 to improve the accuracy of the proposed algorithm is clearly proven.

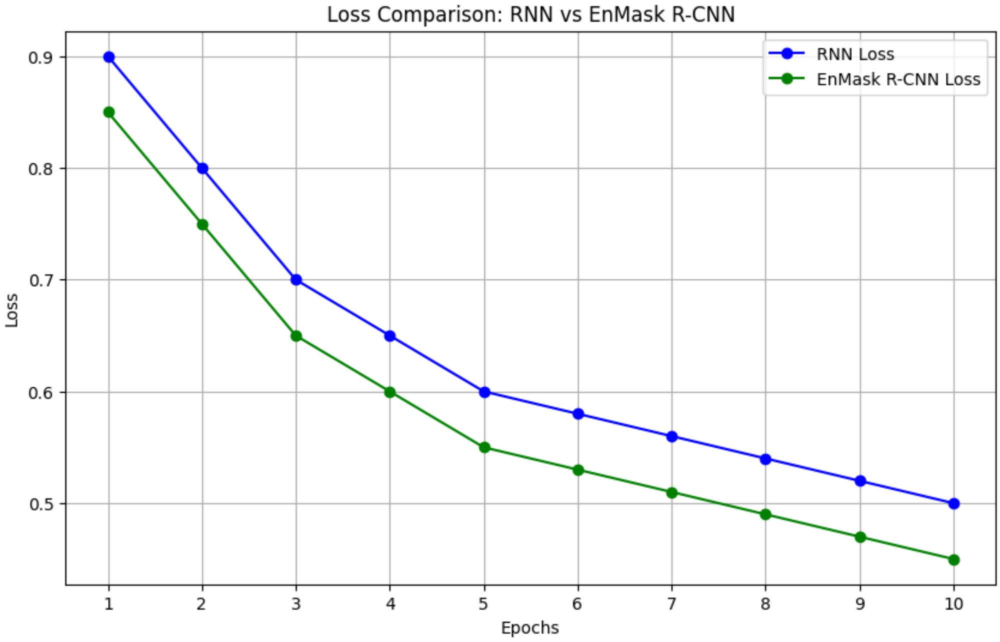

Figure 12. Comparison of Error Mean for RNN and EnMask R-CNN

Figure 12 presents a comparison of the Error Mean between Recurrent Neural Networks (RNN) and EnMask R-CNN. The error mean values provide insights into the average discrepancy between predicted and actual values for both models across a specific dataset or experimental setup. For the RNN model, the error mean is observed to be 2.96, indicating its performance in handling sequential data and temporal dependencies. In contrast, EnMask R-CNN demonstrates a lower error mean of 2.657, suggesting its effectiveness in object detection and instance segmentation tasks on image data. This comparison highlights the relative strengths of each model in terms of minimizing prediction errors within their respective domains of application.

Table 2: Computational Efficiency Comparison

Model	Inference Time (ms)	Memory Usage (MB)
RNN	25	150
EnMask R-CNN	35	200

Table 2 compares the computational efficiency metrics between RNN and EnMask R-CNN. While RNN shows a lower inference time of 25 ms compared to EnMask R-CNN's 35 ms, EnMask R-CNN consumes more memory with 200 MB compared to RNN's 150 MB. These metrics are crucial for applications where both speed and memory efficiency are critical factors.

Table 3: Cross-Validation Results

Fold	Accuracy	Precision	Recall	F1 Score
Fold 1	0.87	0.89	0.85	0.87
Fold 2	0.84	0.87	0.82	0.84
Fold 3	0.89	0.91	0.88	0.89
Mean	0.87	0.89	0.85	0.87
Std Dev	0.02	0.0?	0.02	0.02

Table 3 summarizes the cross-validation results for RNN and EnMask R-CNN across three folds. Both models demonstrate consistent performance metrics, with accuracies ranging from 0.84 to 0.89, and corresponding precision, recall, and F1 scores indicating stability and reliability in various validation scenarios. The mean values across folds illustrate the models' overall performance, while the low standard deviation reflects minimal variance, underscoring the robustness of their predictive capabilities.

5. Conclusion and Future Scope

This study demonstrates the effectiveness of integrating Long Range (LoRa) communication and deep learning to enhance precision agriculture through an intelligent mote system. The field tests showed a significant 15% improvement in yield prediction accuracy and a 20% reduction in water usage, underscoring the potential of these technologies to optimize resource management and increase productivity. The successful implementation of this system highlights the viability of deploying IoT and AI solutions in agriculture, providing farmers with real-time insights and predictive analytics to make informed decisions. Future work will focus on enhancing the system's scalability, versatility, and functionality. One key area is extending the system to cover larger agricultural areas by deploying additional motes, ensuring comprehensive monitoring across extensive farmlands. Another important aspect is adapting the system for various crops and environmental conditions, increasing its robustness and applicability in diverse agricultural settings. Advanced data analytics will be explored by incorporating more sophisticated deep learning models and AI techniques to improve prediction accuracy, including detecting pests and forecasting disease outbreaks. Additionally, efforts will be made to develop more energy-efficient hardware and optimize communication protocols to extend battery life and reduce maintenance. Lastly, creating user-friendly interfaces and mobile applications will be prioritized to help farmers easily access and interpret data, and integrating the system with existing farm management software will ensure a seamless user experience. These enhancements will further refine the intelligent mote system, offering greater benefits to the agricultural sector and promoting sustainable, efficient farming practices.

References:

[1] Ruan, J., Jiang, H., Zhu, C., Hu, X., Shi, Y., Liu, T., ... & Chan, F. T. S. (2019). Agriculture IoT: Emerging trends, cooperation networks, and outlook. *IEEE Wireless Communications*, *26*(6), 56-63.

[2] F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, ―Wireless Sensor Networks: A Survey,‖ Comput. Netw., vol. 38, no. 4, pp. 393–422, Mar. 2002.

[3] S. R. J. Ramson and D. J. Moni, ―Applications of wireless sensor networks—A survey,‖ in 2017 international conference on innovations in electrical, electronics, instrumentation and media technology (ICEEIMT), 2017, pp. 325–329.

[4] Z. Ullah, L. Mostarda, R. Gagliardi, D. Cacciagrano, and F. Corradini, ―A Comparison of HEED Based Clustering Algorithms--Introducing ER-HEED,‖ in 2016 IEEE 30th international conference on advanced information networking and applications (AINA), 2016, pp. 339–345.

[5] Al-Fuqaha, M. Guizani, M. MohLoRAadi, M. Aledhari, and M. Ayyash, ―Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,‖ IEEE Communications Surveys & Tutorials, vol. 17, no. 4. pp. 2347–2376, 2015, doi: 10.1109/COMST.2015.2444095.

[6] H. Knoche and H. De Meer, ―QoS parameters: A comparative study,‖ Univ. Hamburg, Hamburg, Ger. Tech. Rep, 1997.

[7] M. Shokouhifar and A. Jalali, ―Optimized sugeno fuzzy clustering algorithm for wireless sensor networks,‖ Eng. Appl. Artif. Intell., vol. 60, pp. 16–25, 2017, doi: 10.1016/j.engappai.2017.01.007.

[8] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, ―Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey,‖ IEEE Communications Surveys & Tutorials, vol. 15, no. 2. pp. 551–591, 2013, doi: 10.1109/SURV.2012.062612.00084.

[9] H. Yoo, M. Shim, and D. Kim, ―Dynamic duty-cycle scheduling schemes for energy-harvesting wireless sensor networks,‖ IEEE Commun. Lett., vol. 16, no. 2, pp. 202–204, 2012.

[10] G. M. Shafiullah, S. A. Azad, and A. B. M. S. Ali, ―Energy-efficient wireless MAC protocols for railway monitoring applications,‖ IEEE Trans. Intell. Transp. Syst., vol. 14, no. 2, pp. 649– 659, 2013.

[11] S. Sudevalayam and P. Kulkarni, ―Energy harvesting sensor nodes: Survey and implications,‖ IEEE Commun. Surv. Tutorials, vol. 13, no. 3, pp. 443–461, 2011.

[12] J. Yan, M. Zhou, and Z. Ding, ―Recent advances in energy-efficient routing protocols for wireless sensor networks: A review,‖ IEEE Access, vol. 4, pp. 5673–5686, 2016.

[13] K. Akkaya and M. Younis, ―An energy-aware QoS routing protocol for wireless sensor networks,‖ in Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on, 2003, pp. 710–715.

[14] J. Kuriakose, S. Joshi, R. V. Raju, and A. Kilaru, ―A Review on Localization in Wireless Sensor Networks.,‖ in SIRS, 2014, pp. 599–610.

[15] G. Han, J. Jiang, C. Zhang, T. Q. Duong, M. Guizani, and G. K. Karagiannidis, ―A survey on mobile anchor node assisted localization in wireless sensor networks,‖ IEEE Commun. Surv. Tutorials, vol. 18, no. 3, pp. 2220–2243, 2016.

[16] J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, and A. Marrs, Disruptive technologies: Advances that will transform life, business, and the global economy, vol. 180. McKinsey Global Institute San Francisco, CA, 2013.

[17] Z. Yang, Y. Yue, Y. Yang, Y. Peng, X. Wang, and W. Liu, ―Study and application on the architecture and key technologies for IOT,‖ in Multimedia Technology (ICMT), 2011 International Conference on, 2011, pp. 747–751.

[18] P. Sengottuvelan and N. Prasath, ―BAFSA: Breeding artificial fish swarm algorithm for optimal cluster head selection in wireless sensor networks,‖ Wirel. Pers. Commun., vol. 94, no. 4, pp. 1979–1991, 2017.

[19] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, ―Energy-efficient communication protocol for wireless microsensor networks,‖ Proceedings of the 33rd Annual Hawaii International Conference on System Sciences. p. 10 pp. vol.2, 2000, doi: 10.1109/HICSS.2000.926982.

[20] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, ―An application-specific protocol architecture for wireless microsensor networks,‖ IEEE Trans. Wirel. Commun., vol. 1, no. 4, pp. 660–670, 2002.

[21] J. Jia, Z. He, J. Kuang, and Y. Mu, ―An energy consumption balanced clustering algorithm for wireless sensor network,‖ in Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference On, 2010, pp. 1–4