Internet of things-drone trajectory planning model with edge computing based on long range payload in rural areas

Eddy Prasetyo Nugroho[1,2], Taufik Djatna[3], Imas Sukaesih Sitanggang[1], Irman Hermadi[1]
[1]Department of Computer Science, Faculty of Mathematics and Natural Sciences, IPB University, Bogor, Indonesia
[2]Department of Computer Science Education, Faculty of Mathematics and Natural Sciences Education, Universitas Pendidikan Indonesia, Bandung, Indonesia
[3]Department of Agroindustrial Technology, Faculty of Agricultural Engineering and Technology, IPB University, Bogor, Indonesia

Article Info

Article history:

Received May 31, 2024
Revised Oct 14, 2024
Accepted Nov 19, 2024

Keywords:

Edge computing
Internet of things-drone trajectory
Long range
Precision agriculture
Rural areas

ABSTRACT

The integration of internet of things (IoT) with unmanned aerial vehicle (UAV) or drone, for precision agriculture (PA) in rural tea plantations is required to ensure optimal outcomes. However, rural settings presents exceptional challenges for data transmission, particularly in maintaining effective communication between drone and ground control stations (GCS). Therefore, this research aimed to develop a payload metadata identification model using long range (LoRa) technology, known for robust IoT capabilities of the model. LoRa was used to transmit drone data packets to GCS, including image data computations and onboard sensor information. Additionally, the research proposed IoT-drone trajectory planning model, specifically designed for PA in rural tea plantations. This model incorporated LoRa technology for data transmission, leveraging the effectiveness of the model in remote areas. Edge computing was also integrated into model to classify the suitability of tea plantation picking areas based on image captured with drone. An important component of the research was trajectory planning system, which optimized drone flight paths by considering location data, throughput data, battery energy consumption, and the computation of suitable picking locations. Finally, experimental results showed the effectiveness of the proposed model in identifying payload metadata, monitoring drone trajectory, and optimizing picking location paths in rural tea plantations.

Corresponding Author:

Taufik Djatna
Department of Agroindustrial Technology, Faculty of Agricultural Engineering and Technology
IPB University
Raya Dramaga Street, Campus IPB Dramaga Bogor, Bogor, 16680 West Java, Indonesia
Email: taufikdjatna@apps.ipb.ac.id

1. INTRODUCTION

The use of internet of things (IoT) technology in the agricultural sector along with physical objects such as sensors and IoT communication networks is in support of potential development of precision agriculture (PA) technology [1], [2]. Precision monitoring aims to identify plantation areas with potential for high-quality tea leaves, thereby increasing tea production. Moreover, the implementation of this PA includes an unmanned aerial vehicle (UAV) or drone, which can be used to monitor plants from the air [1], [3]-[5]. Drone is capable of continuously and reliably performing environmental monitoring tasks for remote sensing. Additionally, the device has a camera installed and provides an image of the environment in the aimed areas [6].

Implementation of PA on large, hilly tea plantations is typical of rural areas with agricultural landscape topography [7], which require network technology devices that can transmit reliable data. The definition of rural areas generally refers to classifications based on topography, access, or distance to (at least) facilities, such as urban areas and agricultural landscapes [7]. Moreover, research has proven the use of long range (LoRa) technology and its frequency channels for communication in rural environments, including hectares of tea plantations [8]. LoRa is a communication protocol that governs the transmission of data packets using a LoRa data frame, which has a maximum size of 256 bytes. This protocol can transmit data packets up to 20 km line of sight (LoS) in rural areas and 5 km in urban areas [9]-[11]. Additionally, data transformation is based on LoRa protocol, which limits data transmitted from drone to ground control station (GCS) for remote monitoring in the field. Identifying drone monitoring locations, collecting throughput data, and gathering information on the energy consumption of the device batteries are necessary to ensure the viability of tea plantation picking areas, shaped similarly to hills with plantation blocks. GCS receives this information from drone data package, which then informs trajectory planning of the device, a crucial factor to consider during PA implementation.

Drone trajectory planning includes designing trajectory to collect data from drone sensors, considering specific requirements such as complete data acquisition and efficient battery energy consumption [12]. The device sensors will provide information on feasible tea picking location points, throughput data, timestamps, and drone battery usage during flights from identified location points to monitor the viability of tea picking in this plantation areas. In addition, the device will determine an optimal location by calculating the shortest distance between identified location points to gather path planning data. To perform this process, the device applies traveling salesman problem (TSP) method combined with simulated annealing (SA) algorithm, which ensures each location point is visited once, and returns to the starting point [13], [14]. SA algorithm is a stochastic meta-heuristic method that returns an effective solution search for TSP problem [13]-[15]. Following the discussion, effective calculations for drone usage during flight require information related to throughput data during the device data transmission to GCS, as well as the optimal power of battery energy consumption by the device.

Using drone intelligence in determining the suitability of tea-picking areas requires edge computing system [16]. The term "edge computing" refers to a technological concept that enables computation at edge of a cloud service network and facilitates data transmission to IoT services [17]. A classification process using a machine learning method with artificial neural network (ANN) for visual computation [18] is applied to drone image received by OpenMV camera sensor to determine which tea plantation blocks are ready for harvest [19]. During this research, the classification process begins with preprocessing, which considers the level of vegetation (vegetation index) according to red-green-blue (RGB) data from drone image. Moreover, formulas that focus on RGB in regional imagery from drone are visible atmospherically resistant index (VARI), green leaf index (GLI), and visible atmospherically resistant indices green (VIGreen). Based on the performance of the three formulas in both urban and forest (rural) areas, the method sensitive to leaf greenness is GLI vegetation index [20]. Following this process, classification begins with a dataset derived from GLI calculations on each drone image pixel. Before transmitting data to GCS, drone will integrate the classification results from edge computing with onboard sensors. This process allows metadata identification necessary in forming IoT-drone payload data packets based on LoRa protocol.

Previous investigations, which serves as the foundation for this research, focus on drone payload metadata and edge computing in rural tea plantation areas to meet PA requirements. The research by Silvagniet *et al.* [21], Bejiga *et al.* [22] used the device with infrared (IR) and thermal camera sensors to help with search and rescue (SAR) in mountainous rural areas. This method does not use the technology and cannot record GPS location data [21], [22]. PA research, which uses drone for smart city air pollution monitoring, has successfully transmitted sensor data, including optical camera sensors, to LoRa-based GCS. However, the investigation does not apply edge computing for sensor data processing [23]. Research in the agriculture sector focuses on crop monitoring, using drone and wireless sensor network (WSN) spectral cameras for data acquisition. This research succeeded in sending all the sensor data and has used edge computing, at edge node with a WSN connection on the ground [12]. Consequently, several research on drone trajectory view SAR planning [24], organizing edge terrain nodes for crop monitoring [12], and using the device orchestrators to design transmissions [25]. These three investigations monitored GPS sensor location and energy consumption but did not optimally use information from edge computing in drone with cheap, ordinary RGB cameras. The device trajectory planning requires precision due to the limited resources of IoT drone, which include computing power, energy, bandwidth, and storage [26]. In addition, limitations of the data transmission technology used by the device include WSN or LoRa technology.

This research aimed to develop a mechanism that identifies metadata from IoT drone, crucial for PA applications affecting rural tea plantation areas, thereby improving the feasibility of LoRa-based tea picking. Model combines edge computing classification process with ANN machine learning method, based on GLI vegetation index from tea plantation image captured by OpenMV drone camera. Following the discussion, this method aimed to manufacture more intelligent drone. The research focuses on the communication of the device payload data transmission to GCS via LoRa technology in rural tea garden areas in drone trajectory planning. The final output of the research is a device trajectory model specifically designed for rural areas. This model, known as drone flight location trajectory, is used to determine the shortest path to tea plantation blocks where harvesting is feasible by applying TSP method with SA algorithm. Moreover, the device trajectory efficiently computes throughput data and battery energy consumption of drone payload.

2. METHOD

The research comprised three main stages which included identifying payload metadata from LoRa-based IoT-drone sensor data using edge computing, creating a reference dataset for tea leaf picking needs, and using IoT-drone payload metadata information to design trajectory planning model that located tea plants ready for picking. The first step in Figure 1 showed data collection or acquisition process, which started with capturing data from drone flying over the tea plantation [27]. The device sensors captured this data from image and onboard sensors specifically designed for PA in expansive rural tea plantation areas. The need for edge computing to improve the intelligence of drone necessitated the processing of sensor data from IoT drone to determine the feasibility of tea plantation harvesting. Moreover, the technology process used computer vision methods in OpenMV to process image based on GLI [20] of the leaves (Figure 2). The technology also used machine learning with a sequential ANN model from TensorFlow to sort the image captured by drone (Figure 3).

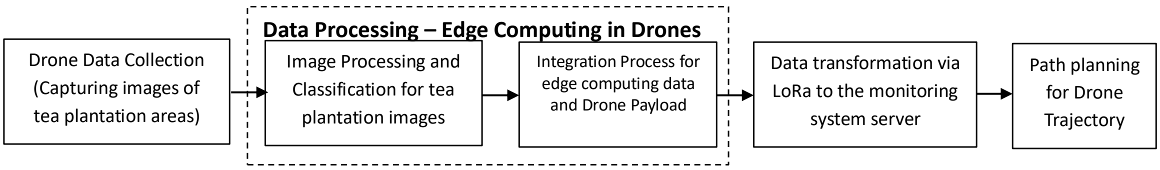


Figure 1. The research stages

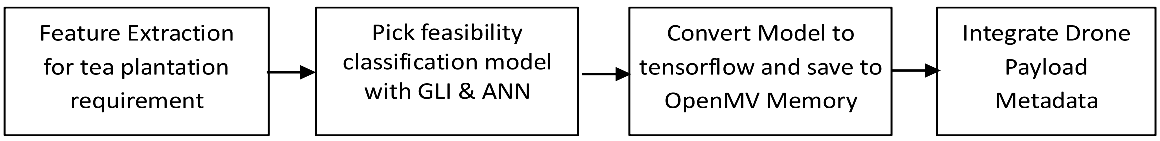


Figure 2. Edge computing of drone image data to payload integration process

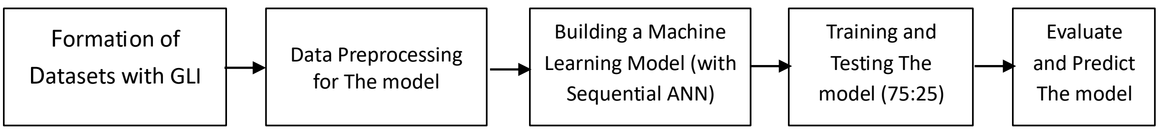


Figure 3. A classification model for picking tea plants using GLI and ANN

ANN-based classification model for drone image began with the design of a tea plantation dataset with GLI calculations from RGB color values [20] using (1):

$$GLI = \frac{(2G-R-B)}{(2G+R+B)} \quad (1)$$

The device payload data incorporation model in the results and discussion chapter provided a clearer description of combination process for designing LoRa-based drone payload metadata frames. The device communication protocol in tea garden of rural areas limited data frame size and space characteristics. Additionally, the data transformation process transmitted the information from the device to GCS via LoRa and generated payload metadata.

The process of forming path planning for drone relied on receiving LoRa-based device payload data packets from the monitoring server or GCS. The method served as a tool for monitoring information related to location, time, throughput, edge computing result of tea picking suitability, and battery energy consumption of the drone. This information was valuable for managing tea plantations, particularly in assessing the quality of picking in the rural tea plantation areas. Figure 4 showed the various stages of the process concerning the research.

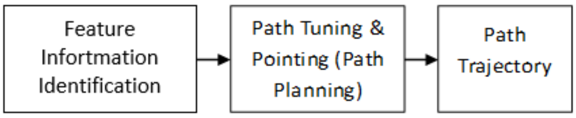


Figure 4. The formation process of trajectory planning

The figure started the process of identifying feature information from telemetry data received by the GCS server from the drone. This process was conducted to monitor the feasibility of tea picking, focusing on location data, drone flight paths, timestamps, throughput, edge computing results, picking feasibility classification, and battery energy consumption. The steps in path tuning and pointing process included turning data patterns into paths based on the characteristics of trajectory data, creating the data, and finally finding the best results that fit the needs as well as limitations of using the data.

In trajectory planning concerning the appropriate location for tea plantation picking, calculating the distance (d) between location points ($v1, v2, \ldots, vn$) to obtain trajectory was necessary. Moreover, the distance of each point to other points was calculated using the following haversine formula (2) [28]:

$$d = 2 * R * asin \sqrt{\sin^2(\frac{\delta lat}{2}) + \cos(lat1) * \cos(lat2) * \sin^2(\frac{\delta long}{2}))} \quad (2)$$

where: d is distance, R is earth radius (~6,371 km), $\delta lat = (lat2 - lat1)$, and $\delta long = (long2 - long1)$. Converting degrees to radians for longitude ($long$) and latitude (lat) values using (3) was important.

$$radian = degree \ * \left(\frac{\pi}{180}\right) \quad (3)$$

Processing the number of telemetry data packets successfully transmitted by drone to GCS via LoRa in the transmission time limit at each drone position in bytes/s determined trajectory throughput of the device. Meanwhile, trajectory was related to battery energy (power) consumption by multiplying the voltage (V) and current (I) of the battery energy of drone at each movement location.

Aggarwal and Kumar [26] argued that path planning method aimed to optimize the path for processed data, thereby generating optimum results. In this scenario, the shortest or minimum distance was achieved, minimized battery energy consumption, and maximized data throughput. The location trajectory drone was proposed to optimize the shortest distance using TSP method with (4). Additionally, SA algorithm was proposed to optimize the shortest distance between locations:

$$min \sum_{i=1}^{n} \sum_{j=1}^{n} d(i,j).x_{ij} \quad (4)$$

where $\sum_{j=1, j\neq i}^{n} x_{ij} = 1, \forall i \in \{1,2,\ldots,n\}$ and $\sum_{i=1, i\neq j}^{n} x_{ij} = 1, \forall i \in \{1,2,\ldots,n\}$, each point was visited once, $d(i,j) =$ the distance between vi and vj points, and $x_{ij} = 1$ when the drone moved from vi to vj.

Meanwhile, the (5) and (6) calculated minimum battery energy consumption and maximum data throughput, respectively:

$$min \sum_{i=1}^{n} \sum_{j=1}^{n} P_{ij}.x_{ij} \quad (5)$$

$$max \sum_{i=1}^{n} \sum_{j=1}^{n} t_{ij}.x_{ij} \quad (6)$$

where P_{ij} is battery energy or power consumption during the device movement x_{ij} , t_{ij} are the throughput between vi and vj, and $x_{ij} = 1$ when the drone moved from vi to vj.

3. RESULTS AND DISCUSSION

Testing IoT-drone for remote sensing needed in rural environments was conducted at PPTK Gambung and PTPN 1 tea plantations, Ciwidey, Bandung, Indonesia. Flight pattern followed natural conditions and directions from partner stakeholders for ready-picked gardens (fit for picking) and already picked (not fit for picking).

3.1. A classification model for picking tea plants with drone edge computing

The drone was previously trained with data from the two gardens of the partners, PPTK Gambung and RancaBali Gardens, Ciwidey, Bandung. During the research, the training conducted edge computing experiments on three flights using GLI vegetation index and sequential ANN model, as shown in Figure 2. Figure 3 showed the image classification model that determined the feasibility of collecting each piece of data received by the drone. Following this discussion, the stages during the training were described as follows.

– The process of forming a dataset from image by calculating the vegetation index for each image pixel using (1) [20]. The level of greenness obtained from the image was a vegetation index value from -1 to 1 which was scaled to a value range of 0 to 1 with the visualization in Figure 5.

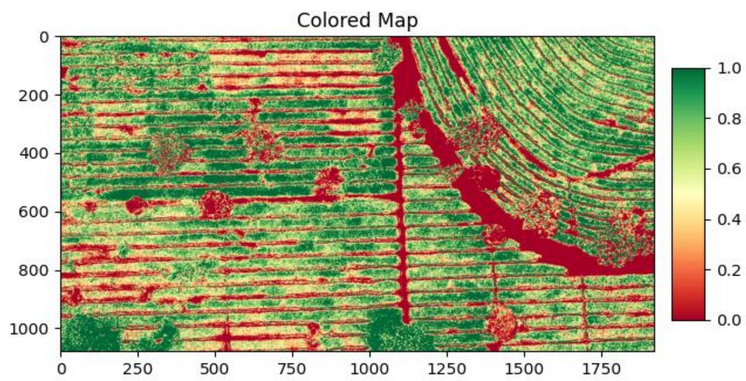


Figure 5. Vegetation index with GLI and a colored map

The dataset was formed from each image data which included feasibility status of the expert label, total pixels, average vegetation index value, and the total value of pixels in 10 sections of vegetation index (0-1).

– Data preprocessing included defining the training section as 75% of the data, namely 138 data, and the testing section as 25% of the data, namely 46 data. This process was shown by Python code snippet in Figure 6, which identified X_train as training data and y_train to be future aimed label.

```
….
X_train, X_test, y_train, y_test = train_test_split(df, output_rows, test_size=0.25,
random_state= 0)
…
Output:
        X_train: (138, 12)
        y_train: (138,)
        X_test: (46, 12)
        y_test: (46,)
```

Figure 6. Definition of training data and testing part of model

– The construction of the machine learning model in Figure 7 included implementing a sequential ANN model using KerasTensorflow module. This model consisted of 12 input layers, which contained feature information from the dataset, and one classification output layer. Model was then compiled using Adam optimizer, with a learning rate of 0.001 and a fit for model training with an epoch of 300.

```
……
# define the keras model
model = Sequential()
model.add(Dense(units=16, input_shape=(12,), activation='relu'))
model.add(Dense(1, activation='sigmoid'))

# compile the keras model
adam = keras.optimizers.Adam(learning_rate=0.001)
model.compile(loss='binary_crossentropy', optimizer='adam', metrics =['accuracy'])
# fit the keras model on the dataset
model.fit(x_train, y_train, epochs=300)
…….
```

Figure 7. Construction model and training model in Python code

– The evaluation and prediction model in Figure 8 provided evaluation results with an accuracy of 82% and a loss of 49%.

```
[ ] # Evaluate the restored model
    loss, acc = new_model.evaluate(X_test, y_test)
    print('Restored model, accuracy: {:5.2f}%'.format(100 * acc))

    2/2 [==============================] - 0s 8ms/step - loss: 0.4896 - accuracy: 0.8158
    Restored model, accuracy: 81.58%
```

Figure 8. Evaluation of ANN model in Python code

Model was saved with the command `basic_model.save('my_model.keras')`. Table 1 presented a confusion matrix that tested the potential of model to produce prediction results on the given data. Following this process, the result showed that model can only positively predict all data, thereby enabling the preparation of ready-to-use predictions.

Table 1. Confusion matrix from the predictions model obtained

	Predicted negative (0)	Predicted positive (1)
Actual negative (0)	27 True negative	7 False positive
Actual positive (1)	0 False negative	12 True positive

The evaluation of the classification model produced precision=0.63, recall=1.0, F1 score=0.77, and accuracy=0.82 based on the prediction results. The image data from the ANN model method was then converted into drone in OpenMV memory with h5 model allowing model to run on the device when used for realization in the field in rural areas. The conversion model stages in this research were shown in Figure 9.

```
The Conversion Process of ANN Classification Model to Tensorflowlite
Result: model.tflite
 Train model dalam Tensorflow;
 Save(model.h5);
 Tf.lite.TFliteConverter from_keras_model(model); // Convert the model to
Tensorflow lite
 Save(model.tflite)
 ModelOpenMV = model.tflite;
 Test(ModelOpenMV);
```

Figure 9. The process of converting an ANN classification model to OpenMV-drone

3.2. Drone payload metadata identification

Before sending drone data to GCS, edge computing process culminated in the incorporation of image data processing and onboard system data from the device. This process led to the device payload data, as shown in Figure 10. The process of adding data from the payload included data from OpenMV camera, which captured image of tea plantations. The image was then processed using ANN algorithm for classification into two groups including blocks ready to be picked and those not suitable for picking. These groups were already set up in edge computing during the research.

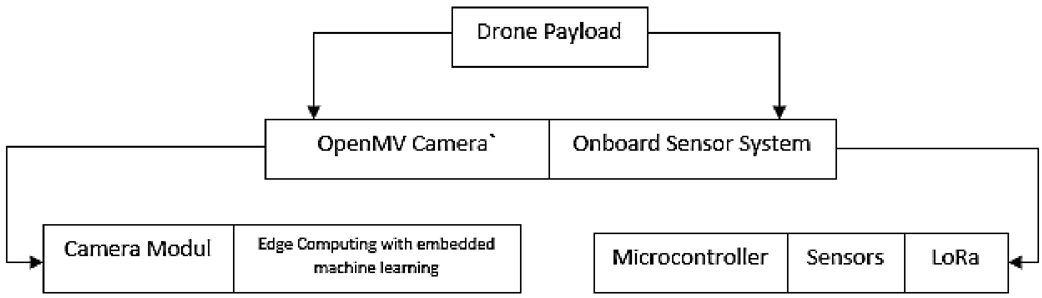


Figure 10. LoRa-based drone payload data integration process [29]

Figure 11 showed the process in OpenMV_data (i), where experts and GLI vegetation index calculations determined whether to classify each image data identifier I as 1 (ready to pick) or 0 (not ready to pick) based on the results. For the data frame section 1, the onboard sensor functioned to measure several parameters, including GPS-based drone position (latitude, longitude, altitude), speed, direction, battery voltage, and current sensors. Following this discussion, sensor_data (i) captured all this sensor data during the research.

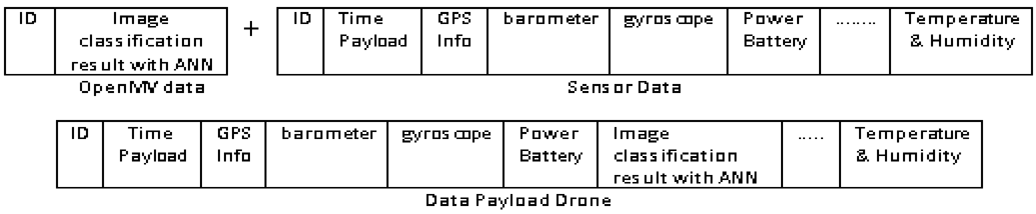


Figure 11. The drone payload data frame package was formed

The hilly nature of tea plantation block, coupled with the characteristics of rural areas, necessitates the use of an appropriate drone communication protocol, specifically LoRa protocol. Moreover, the device data payload package was designed with LoRa, and the data transformation was limited by the size of the data frame sent to the destination GCS server. The payload data package from the incorporation process, as previously shown in Figure 11, included drone package identifier information, recording time, location, barometer, gyroscope, battery energy, magneto, garden profile, harvest status (classification edge computing results), temperature, and humidity.

Algorithm 1 showed how to transform drone data by sending drone payload data packets to GCS at a frequency of 425 MHz and a size of 160 bytes each, using LoRa-based communication module. The monitoring results in the telemetry data section of Table 2 showed the results of data processing from drone computing to LoRa-based GCS. Specifically, the class column represented the outcome of edge computing, which classified tea garden device image using ANN algorithm. When the image classification produced a value of 1, it showed that the tea garden at the flying location of drone was suitable for picking, while a value of 0 signified the garden was not yet ready to be picked. This process was determined by ANN machine learning using the trained GLI vegetation index.

Algorithm 1. Drone-GCS data transformation via LoRa
```
initialization;
//variable & function declaration
//IO Digital Setup
//Yahboom IMU (JY901), GPS, Current and INA219 voltage Sensor Setup
//LoRa Ebyte 433 Module Setup and UART Communication for OpenMV Setup
Timer = $sensor_timer
While (Timer > 5 second}
 do {
 getData(Yahboom IMU,GPS, Current and INA219 voltage Sensor);
        request(OpenMV_Data);
 If (OpenMV_Data, Sensor_data)== true then //data valid
        {
        Check serial_channel to onboard;
        If (Serial_available == True)then
        {
        Serial_Connection == True;
 OpenMV_Data_Sending == True;
        Append(Append(OpenMV_data, Sensor_data),delimeter(",")) ;
        } else
        {
        Serial_Connection == False\;
        }
        }else
 {
        OpenMV_Data == False\; Sensor_data == False\;
               Append("NaN",delimeter(","))
        }
 }
Data_sending_From_Lora_to_GCS(i)= +1;
Timer=0;
 }
```

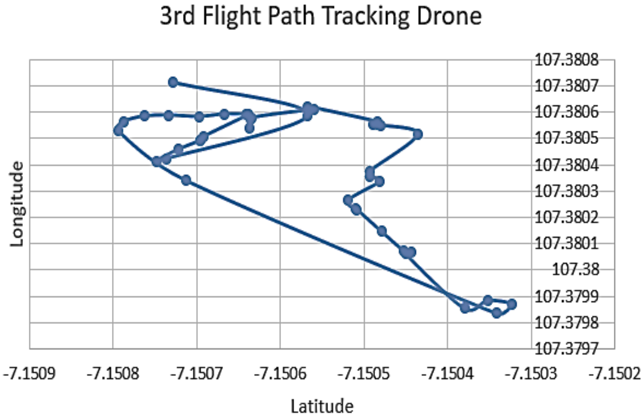
Table 2. The telemetry data review was the result of drone-GCS data transformation

ID	Time payload	Lat	Long	Bar (cmdpl)	I-Bat (mA)	Vbat (V)	P (mW)	No. of Garden	Class (bool)	..	Hum (%)
..	…	..	..	..	..	..	..	..	..	..	..
9	13:28:47	-7.17	107.367	166513	196.3	12.23	2418	11	1	..	64
10	13:28:55	-7.176	107.367	166445	206.6	12.2	2488	12	1	..	65
11	13:29:06	-7.176	107.367	166984	208.5	11.26	2360	13	1	..	66
12	13:29:22	-7.1762	107.367	166896	231.7	11.06	2350	15	0	..	68
13	13:29:32	-7.1758	107.367	166445	224	11.04	2504	16	0	..	68
14	13:29:40	-7.1757	107.367	166631	230.5	10.96	2546	17	0	..	69
..	…	…	..	…	…	..	…	..	..	..	..

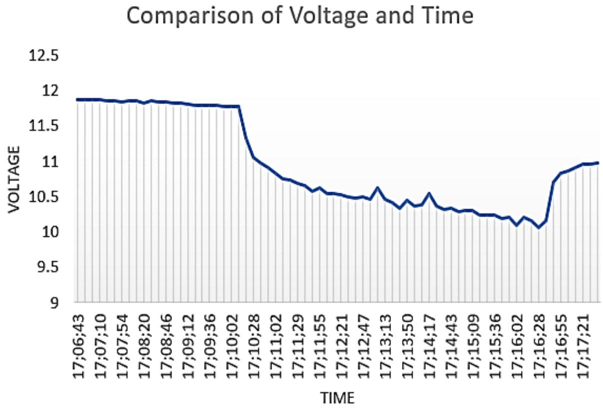
3.3. Monitoring internet of things-drone trajectory

From each drone flight simulation, the monitoring results on GCS provided information on the device flight trajectory, based on its location points. This information aided observers in determining the location of the plantation areas, as shown in Figure 12(a). Moreover, Figure 12(b) showed how the drone trajectory uncovered battery energy consumption of the device controller module during observations in the tea plantation areas.

Experimental simulations using flight data which were conducted three times with partners, led to a monitoring analysis on GCS as shown in Table 3. This analysis concluded that approximately 97% of drone payload data packets were successfully sent, including all metadata. The limited battery capacity of UAVs or the device affected flight time [30]. Therefore, gathering flight time information was necessary when drone reached all the intended locations in rural tea plantation areas. Table 3 showed that the average battery energy usage duration was 11.49 minutes. According to (5), the minimum drone battery energy consumption was an average of 0.467 Watthour (Wh), and each device flight had a maximum throughput of 16.95 bytes/s (average payload packet size is 149 bytes) at an altitude of 17.88 meters with (6). The location predictions of flight 1 were accurate for already-picked plantations, but predictions for 2 and 3 for ready-to-pick plantations showed an invalid prediction status due to a significant proportion of unpickable areas. Additionally, the uneven picking probably caused image detections of ready-to-pick plantations, or it necessitated the addition of training data from the same partner under identical conditions.



(a)



(b)

Figure 12. Trajectory of: (a) location drone and (b) battery energy

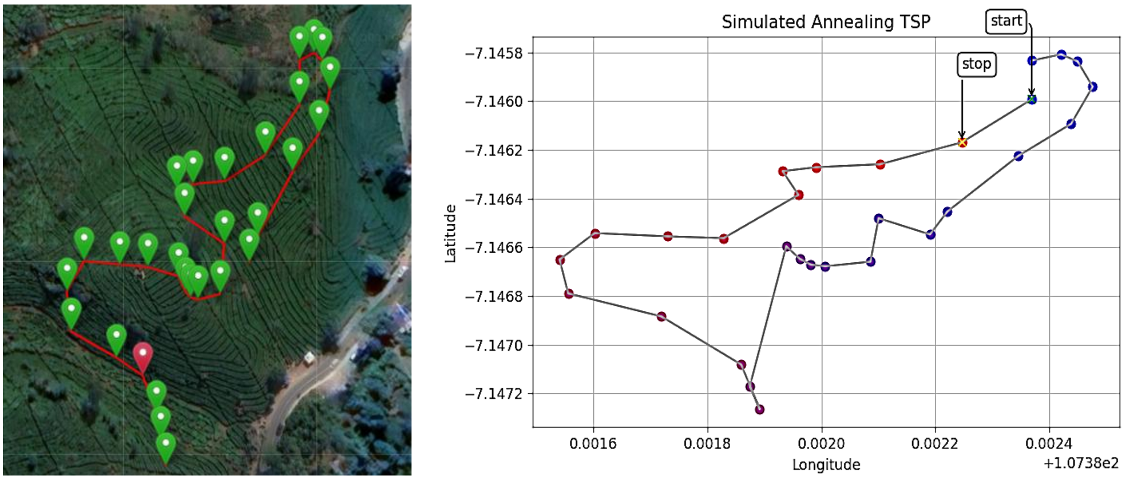
Table 3. The results of LoRa-based drone trajectory analysis

No. of Test	Time flies (minute)	Number of data packets sent (n)	Throughput (byte/s)	Payload drone battery energy (Watt)	Payload energy consumption (Wh)	Height (m)	Predictions ready to be picked (%)	Predictions are not ready to be picked (%)
1	10.47	75	17.224	2.433	0.4246	12.94	4	96
2	13.44	90	17.203	2.459	0.5509	30.28	47	53
3	10.56	69	16.413	2.419	0.4258	10.42	39	61
Average	11.49	78	16.947	2.437	0.4671	17.88	30	70

4.4. Optimizing location trajectory of the picking tea plantation in rural areas

The marked locations of the tea plantation blocks that were ready to be picked were optimized to find the shortest path to be used by pickers from IoT-drone trajectory generated in rural areas, concerning the location trajectory using (2) and (3) to calculate the distance. This process occurred to reduce time moving to the tea plantation picking location using SA algorithm for TSP method with (4). This SA algorithm considered temperature changes and associated costs to determine the possibility of a new optimal solution when comparing the shortest distance results.

Figure 13(a) showed the visualization results of GCS server monitoring, as drone movement trajectory results were marked with a red line. The locations of the plantation areas in greened serve as the basis for calculating the shortest distance between the random starting point of the picking location and the final location. Moreover, Figure 13(b) showed the application of TSP method, coupled with SA algorithm to determine the shortest distance. This process was achieved by simulating the formation of a new trajectory and comparing the route to the optimal shortest distance to the randomly marked picking location [13], [15], [31].



(a) (b)

Figure 13. The SA algorithm for the optimum shortest path from the picker based on location marks: (a) GCS visualization and (b) optimal shortest path with the SA algorithm

4. CONCLUSION

In conclusion, implementing smart drone in PA in rural tea plantation areas necessitated a mechanism for developing IoT-drone metadata, which consisted of a data incorporation process with edge computing results from the device image classification and onboard drone sensor data. Sequential ANN (keras.TensorFlow) model was successfully applied based on GLI calculation dataset on each image pixel, which included 12 dataset input layers, an activation function with rectified linear unit (ReLU), and one classification output layer in the device image classification process to determine the suitability of the tea plantation picking areas. During the research, LoRa-based drone payload data packets were successfully sent to GCS. On average, 97% of the packets contained identification information collected by the sensors of the device, such as flight time, location, calibration, payload drone battery energy, garden profile, temperature, and humidity.

The data transformation results from LoRa-based drone to GCS provided the device trajectory that considered throughput, successful data transmission, identification of locations suitable for drone picking and tracking, as well as battery energy consumption. Moreover, TSP method, combined with SA algorithm used the location trajectory results to calculate the shortest distance from the position of the picker, producing a new location graph. The resulting trajectory showed maximum data throughput and minimum battery energy consumption of the drone payload.

ACKNOWLEDGEMENTS

The authors are grateful to IPB University, partners of PPTK Gambung and PTPN 1 Region 2 RancaBali Gardens, Ciwidey, Bandung Regency as the place for this research.

REFERENCES

[1] A. Tzounis, N. Katsoulas, T. Bartzanas, and C. Kittas, “Internet of things in agriculture, recent advances and future challenges,” *Biosystems Engineering*, vol. 164, pp. 31–48, 2017, doi: 10.1016/j.biosystemseng.2017.09.007.
[2] A. Castrignanò, R. Khosla, D. Moshou, G. Buttafuoco, A. M. Mouazen, and O. Naud, *Agricultural internet of things and decision support for precision smart farming*, Academic Press, 2020, doi: 10.1016/c2018-0-00051-1.
[3] F. Al-Turjman, M. Abujubbeh, A. Malekloo, and L. Mostarda, “UAVs assessment in software-defined IoT networks: an overview,” *Computer Communications*, vol. 150, pp. 519–536, 2020, doi: 10.1016/j.comcom.2019.12.004.
[4] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, “A compilation of UAV applications for precision agriculture,” *Computer Networks*, vol. 172, 2020, doi: 10.1016/j.comnet.2020.107148.
[5] V. M. M. O. Ompusunggu, M. K. D. Hardhienata, and K. Priandana, “Application of ant colony optimization for the selection of multi-uav coalition in agriculture,” in *2020 International Conference on Computer Science and Its Application in Agriculture, ICOSICA 2020*, 2020. doi: 10.1109/ICOSICA49951.2020.9243226.
[6] C. Y. Lee, H. J. Lin, M. Y. Yeh, and J. Ling, “Effective remote sensing from the internet of drones through flying control with lightweight multitask learning,” *Applied Sciences (Switzerland)*, vol. 12, no. 9, 2022, doi: 10.3390/app12094657.
[7] L. P. Putri, D. J. Russell, B. G. O’Sullivan, A. Meliala, and R. Kippen, “A critical review of definitions of rural areas in Indonesia and implications for health workforce policy and research,” *Health Research Policy and Systems*, vol. 20, no. 1, pp. 1–15, 2022,

doi: 10.1186/s12961-022-00847-w.
[8] A. Mulyana, S. Wahjuni, T. Djatna, H. Sukoco, H. Rahmawan, and S. N. Neyman, "Internet of things (IoT) device management in rural areas to support precision agriculture," *IOP Conference Series: Earth and Environmental Science*, vol. 1012, no. 1, 2021, doi: 10.1088/1755-1315/1012/1/012083.
[9] A. T. P. Khun, L. Shan, Y. Lim, and Y. Tan, "MCST scheme for uav systems over LoRa networks," *Drones*, vol. 7, no. 6, pp. 1–16, 2023, doi: 10.3390/drones7060371.
[10] F. A. Almalki, B. O. Soufiene, and S. H. Alsamhi, "A low-cost platform for environmental smart farming monitoring system based on iot and uavs," vol. 13, no. 11, 2021, doi: 10.3390/su13115908.
[11] J. Gallego-Madrid *et al.*, "Enhancing extensive and remote lora deployments through mec-powered drone gateways," *Sensors (Switzerland)*, vol. 20, no. 15, pp. 1–15, 2020, doi: 10.3390/s20154109.
[12] D. Popescu, F. Stoican, G. Stamatescu, L. Ichim, and C. Dragana, "Advanced UAV–WSN system for intelligent monitoring in precision agriculture," *Sensors*, vol. 20, no. 3, 2020, doi: 10.3390/s20030817.
[13] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, "Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search," *Applied Soft Computing Journal*, vol. 11, no. 4, pp. 3680–3689, 2011, doi: 10.1016/j.asoc.2011.01.039.
[14] H. Bayram and R. Şahin, "A new simulated annealing approach for travelling salesman problem," *Mathematical and Computational Applications*, vol. 18, no. 3, pp. 313–322, 2013, doi: 10.3390/mca18030313.
[15] N. Adil and H. Lakhbab, "A new improved simulated annealing for traveling salesman problem," *Mathematical Modeling and Computing*, vol. 10, no. 3, pp. 764–771, 2023, doi: 10.23939/mmc2023.03.764.
[16] M. Thangaraj and R. S. Sangam, "Intelligent uav path planning framework using artificial neural network and artificial potential field," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, no. 2, pp. 1192–1200, 2023, doi: 10.11591/ijeecs.v29.i2.pp1192-1200.
[17] S. Douch, M. R. Abid, K. Zine-Dine, D. Bouzidi, and D. Benhaddou, "Edge computing technology enablers: a systematic lecture study," *IEEE Access*, vol. 10, pp. 69264–69302, 2022, doi: 10.1109/ACCESS.2022.3183634.
[18] C. Toma, M. Popa, B. Iancu, M. Doinea, A. Pascu, and F. Ioan-Dutescu, "Edge machine learning for the automated decision and visual computing of the robots, IoT embedded devices or uav-drones," *Electronics (Switzerland)*, vol. 11, no. 21, 2022, doi: 10.3390/electronics11213507.
[19] A. Ador and R. Jurga, "Deploying machine learning models for edge computing on drones," *Dataiku and UAVIA*, 2021.
[20] L. S. Eng, R. Ismail, W. Hashim, and A. Baharum, "The use of VARI, GLI, and VIgreen formulas in detecting vegetation in aerial images," *International Journal of Technology*, vol. 10, no. 7, pp. 1385–1394, 2019, doi: 10.14716/ijtech.v10i7.3275.
[21] M. Silvagni, A. Tonoli, E. Zenerino, and M. Chiaberge, "Multipurpose uav for search and rescue operations in mountain avalanche events," *Geomatics, Natural Hazards and Risk*, vol. 8, no. 1, pp. 18–33, 2017, doi: 10.1080/19475705.2016.1238852.
[22] M. B. Bejiga, A. Zeggada, A. Nouffidj, and F. Melgani, "A convolutional neural network approach for assisting avalanche search and rescue operations with uav imagery," *Remote Sensing*, vol. 9, no. 2, 2017, doi: 10.3390/rs9020100.
[23] L. Y. Chen, H. S. Huang, C. J. Wu, Y. T. Tsai, and Y. S. Chang, "A LoRa-based air quality monitor on unmanned aerial vehicle for smart city," *2018 International Conference on System Science and Engineering, ICSSE 2018*, pp. 1–5, 2018, doi: 10.1109/ICSSE.2018.8519967.
[24] M. Lyu, Y. Zhao, C. Huang, and H. Huang, "Unmanned aerial vehicles for search and rescue: a survey," *Remote Sensing*, vol. 15, no. 13, pp. 1–35, 2023, doi: 10.3390/rs15133266.
[25] I. Donevski, N. Babu, J. J. Nielsen, P. Popovski, and W. Saad, "Federated learning with a drone orchestrator : path planning for minimized staleness," *IEEE Open Journal Communication Society*, vol. 2, pp. 1000-1014, 2021, doi: 10.1109/OJCOMS.2021.3072003.
[26] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: a review, solutions, and challenges," *Computer Communications*, vol. 149, pp. 270–299, Jan. 2020, doi: 10.1016/j.comcom.2019.10.014.
[27] K. Priandana, M. Hazim, Wulandari, and B. Kusumoputro, "Development of autonomous UAV quadcopters using pixhawk controller and its flight data acquisition," in *2020 International Conference on Computer Science and Its Application in Agriculture, ICOSICA 2020*, 2020. doi: 10.1109/ICOSICA49951.2020.9243289.
[28] N. Y.-Z. Tan, C.-Y. Ting, and C. C. Ho, *Location analytics for churn service type prediction*, Lecture Notes in Electrical Engineering, Singapore: Springer Singapore, 2020, vol. 603. doi: 10.1007/978-981-15-0058-9.
[29] E. P. Nugroho *et al.*, "Metadata modeling of lora based payload information for precision agriculture tea plantation," *Scientific Journal of Informatics*, vol. 10, no. 2, pp. 217–228, 2023, doi: 10.15294/sji.v10i2.43432.
[30] N. S. A. Ibrahim and F. A. Saparudin, "Review on path planning algorithm for unmanned aerial vehicles," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 2, pp. 1017–1026, 2021, doi: 10.11591/ijeecs.v24.i2.pp1017-1026.
[31] Y. Zhou, W. Xu, M. Zhou, and Z. H. Fu, "Bi-trajectory hybrid search to solve bottleneck-minimized colored traveling salesman problems," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 1, pp. 895-905, Jan. 2024, doi: 10.1109/TASE.2023.3236317.

BIOGRAPHIES OF AUTHORS

Eddy Prasetyo Nugroho received a M.T. degree in informatics (special field of software engineering) from Institut Teknologi Bandung, Indonesia in 2005. Currently he is a Ph.D. Student of Department of Computer Science, IPB University, Indonesia. He is lecturer in Computer Science Study Program, Universitas Pendidikan Indonesia, Indonesia since 2008. His research interests include internet of things (IoT), computer vision, machine learning, network security, and software engineering. He can be contacted at email: eddypnugroho@apps.ipb.ac.id or eddypn@upi.edu.

Prof. Dr. Ir. Taufik Djatna, M.Si. is a Professor at the Department of Agroindustrial Technology, Faculty of Agricultural Engineering and Technology, IPB University, Indonesia. He is a distinguished academic affiliated with IPB University. His educational background includes degrees from Institut Pertanian Bogor and Hiroshima University, focusing on agro-industrial system and information engineering. He has made significant strides in research, particularly in the application of blockchain technology and ERP digital solutions for agro-industry supply chains, as evidenced by numerous publications in reputable journals published by international journal publishers. He has also been recognized for his exceptional teaching and research efforts, receiving several awards. Additionally, he holds professional certifications such as AWS Academy Educator and PMI Project Management Ready, reflecting his commitment to continuous learning, and excellence in his field. He can be contacted at email: taufikdjatna@apps.ipb.ac.id.

Imas Sukaesih Sitanggang is a Professor at the Department of Computer Science, Faculty of Mathematics and Natural Sciences, IPB University, Indonesia. She received a Ph.D. degree in computer science from the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, in 2013. She is Postgraduate Coordinator, Computer Science Department, IPB University since 2019. Her main research interests include spatial data mining and smart agriculture. She can be contacted at email: imas.sitanggang@apps.ipb.ac.id.

Irman Hermadi received a Ph.D. degree in computer science from University of New South Wales (UNSW), Australia in 2015. He is lecturer in Computer Science Department, Faculty of Mathematics and Natural Sciences, IPB University, Indonesia since 2012. His research interests include computational intelligence, software engineering, software testing, information system, knowledge management system, information technology, blockchain, robotics, and artificial intelligence networks (BRAIN). He is National E-agriculture Expert at FAO Indonesia since July 2021, Internal Auditor Team Leader for ISO 9001:2008 certification in Academic Quality Management Standard of the Department of Computer Science at IPB University, and IEEE member since Dec 2019. He can be contacted at email: irmanhermadi@apps.ipb.ac.id.