Institut Riset dan Publikasi Indonesia (IRPI)
MALCOM: Indonesian Journal of Machine Learning and Computer Science
Journal Homepage: https://journal.irpi.or.id/index.php/malcom
Vol. 5 Iss. 4 October 2025, pp: 1387-1395
ISSN(P): 2797-2313 | ISSN(E): 2775-8575

Multi-Classification of Pakcoy Plants using Machine Learning Methods with Smart Greenhouse Dataset

Agung Surya Wibowo[1] , Osphanie Mentari Primadianti[2*],
Muhammad Zimamul Adli[3], Kusnayadi[4],

[1]Electrical Engineering, University of Telecommunication, Indonesia
[2,3,4]Electrical Engineering, University Islam of Nusantara, Indonesia

E-Mail: [1]agungsw@telkomuniversity.ac.id, [2]osphanie@uninus.ac.id,
[3]zimamuladli@uninus.ac.id, [4]kusnayadi8431@gmail.com

Received Jul 30th 2025; Revised Sep 20th 2025; Accepted Oct 20th 2025; Available Online Oct 31th 2025
Corresponding Author: Osphanie Mentari Primadianti

Abstract

This research aims to design and implement a monitoring and classification system for Pakcoy (Brassica rapa L.) plant conditions based on the Internet of Things (IoT) and machine learning algorithms in the Smart Greenhouse of Universitas Islam Nusantara. This study represents one of the applications of IoT and machine learning technology advancements to improve efficiency and effectiveness in the agricultural sector. The developed system utilizes CO_2, SHT30, BH1750, and DHT22 sensors to monitor environmental parameters in real-time, including temperature, humidity, light intensity, panel box temperature, and CO_2 concentration. The monitoring data are used as input for classifying plant conditions using five machine learning methods: Support Vector Machine (SVM), Random Forest, Decision Tree, Logistic Regression, and Multilayer Perceptron (MLP). The results show that the Random Forest algorithm achieves the best performance, with an accuracy of 84%, precision of 86%, recall of 87%, and F1-score of 86%. The implementation of this system serves as a concrete step toward enhancing the efficiency, sustainability, and modernization of hydroponic agriculture in Indonesia.

Keyword: Machine Learning, Pakcoy, Multiclassification, Random Forest, Smart Greenhouse

1. INTRODUCTION

The development of digital-based agricultural technology has driven a major transformation in plant cultivation systems, particularly with the emergence of the Smart Greenhouse concept. This technology integrates various sensors and automated Internet of Things (IoT)-based monitoring systems to create an optimal, efficient, and measurable growing environment. One horticultural crop that requires close monitoring of its environmental conditions is pakcoy (Brassica rapa subsp. chinensis). Both the stems and leaves of Pakcoy (*Brassica rapa* subsp. *chinensis*) are edible and can be consumed either raw or cooked [1]. Because these parts are commonly eaten, especially in salads and stir-fried dishes, monitoring for pesticide residues and environmental contaminants is essential to ensure food safety and maintain nutritional quality, which are highly influenced by carbon dioxide (CO_2) levels, temperature, humidity, light intensity, soil nutrient content, and water availability.

Pakcoy plants come in many varieties, including green, white, and red [2-3]. They have wide and sturdy leaf stalks, and the leaves and veins are thicker than mustard greens and they have smooth leaves, no hairs, and no head. Pakcoy plants have a large market, especially in Indonesia [4-5]. Hydroponic methods for pakcoy cultivation have the advantage of not requiring a large area of land [6]. Pakcoy harvests are determined by measuring plant size and the length of the growing season. These measurements require the garden owner to be present to directly observe the pak choi plants being cultivated. In an effort to increase the effectiveness of pak choi cultivation, several methods have been developed to increase the yield [7-10] and machine learning technology [2,11,12].

Osphanie et al. conducted research using strawberry plants with the Support Vector Machine (SVM) method, achieving an accuracy of 95% [26]. In contrast, our study utilized a different plant species, namely Pakcoy, and obtained an accuracy of 84% using a different dataset. Priyambodo et al. conducted a study and achieved an accuracy of 79% [27]. In contrast, our study, using the same plant species, achieved an accuracy of 84%, which is higher than that reported by the previous researchers. Kumar et al. conducted research on cabbage plants using a different method, namely a hybrid approach combining Convolutional Neural Networks (CNN) and Random Forest for multi-classification. Their model achieved precision, recall, and F1-score values of 86.09%, 86.09%, and 86.09%, respectively [28].

In this study, various sensors were used, including a CO_2 sensor, an SHT30 for temperature and humidity, a BH1750 for light intensity, and a DHT22 as a temperature and humidity data amplifier. The dataset was obtained from a greenhouse system based on the IoT. These sensors were configured to collect 401 rows of data, each containing readings of temperature, humidity, CO_2 concentration, light intensity, and other environmental parameters used as input for model training. The data were collected over a period of 3–4 weeks, covering the entire early growth phase of Pakcoy plants, from the stage of two leaves up to twelve leaves, which were then classified into ten growth classes (class 2 to class 12) based on the number of leaves.

To analyze and classify the collected data, several machine learning algorithms were applied, including SVM, Decision Tree, Logistic Regression, Multilayer Perceptron (MLP), and Random Forest. Among these, the Random Forest method was proposed as the primary model due to its robustness in handling multivariate and complex datasets. Random Forest constructs an ensemble of decision trees to generate more accurate and stable predictions while minimizing the risk of overfitting. In this study, the proposed Random Forest model achieved the best performance, with an accuracy of 84%, a precision of 86%, a recall of 87%, and an F1-score of 86%. Meanwhile, the SVM algorithm was employed to explore its ability to separate data using optimal hyperplanes; the Decision Tree method offered interpretability in decision-making; Logistic Regression provided a simple yet effective linear classification approach; and MLP represented a neural network-based technique capable of capturing nonlinear relationships. By integrating these approaches, the system is expected to automatically identify and classify the growth stages of Pakcoy plants based on measured environmental parameters with high precision and reliability.

This research aims to develop a classification model capable of precisely detecting pakcoy growth stages, which can then be used as a basis for automated decision-making in greenhouse management. This approach not only improves cultivation efficiency but also has the potential to sustainably enhance agricultural productivity and quality.

2. MATERIALS AND METHOD

Figure 1 lists the 100 pakcoy samples that were used as datasets for this investigation. Over the course of three to four weeks, 401 datasets were gathered for use as machine learning input shown in Table 1.

Figure 1. 100 samples of pakcoy

Table 1. Dataset

Dataset	Class
40	2
40	3
40	4
40	5
40	6
40	7
40	8
40	9
40	10
40	11
41	12

This procedure is known as data collecting, and it involves a sample of 100 immature pokcoy plants. The beginning growth of pokcoy starts with 2 to 12 leaves, which are the classes in this categorization. Next, the data is preprocessed, and once the data processing is complete, it is divided into training data (80% composition) and testing data (20%). The employ hyperparameter adjustment after training and testing the data to determine the optimal accuracy from the random forest approach. The flow is depicted in Figure 2.

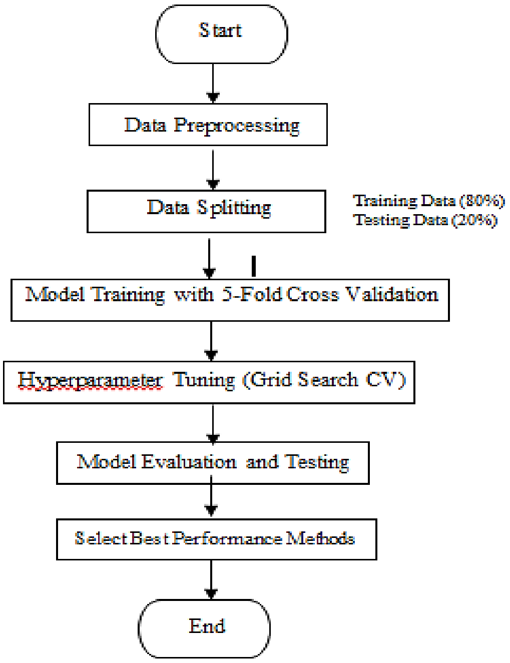

Start
Data Preprocessing
Data Splitting
Training Data (80%)
Testing Data (20%)
Model Training with 5-Fold Cross Validation
Hyperparameter Tuning (Grid Search CV)
Model Evaluation and Testing
Select Best Performance Methods
End

Figure 2. Flow of the Research Method

2.1. Data Collection

During the process of data collecting, we gathered 401 datasets from the web server of the smart greenhouse. These datasets were then categorized into ten classes, beginning with class 2 and going all the way up to class 12. Table 2 depicts the outcome of the investigation.

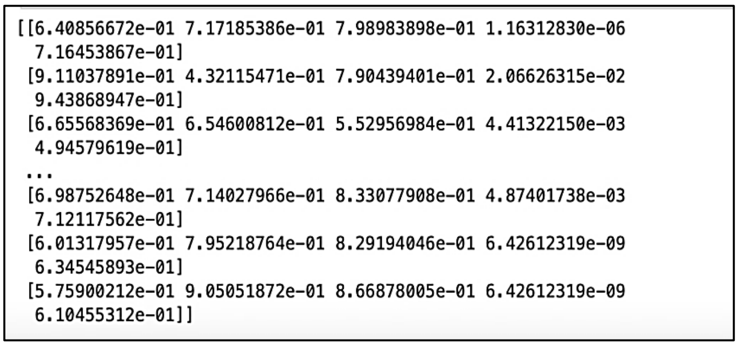
2.2. Data Preprocessing

To facilitate the machine learning process, the dataset has been categorized by class at this point and scaled using the Python Library MinMax Scaler utility as shown in Figure 3. The process of normalization makes sure that the range of all the data in the database is comparable. When the data is unstructured and comprises a wide range of values, this feature is crucial. High-dimensional data benefits from Min Max Scaler normalization [13].

Table 2. Data Collection

No	Temp	Humidity	CO_2	Intensity	Panel Box Temp	Leaf Count
1	29.10	69.00	398.00	755.00	28.40	2
2	1.87	4.53	28.13	56.67	1.91	2
3	25.70	61.00	420.00	775.00	33.90	2
4	24.70	51.00	419.00	895.00	28.90	2
5	27.60	58.00	397.00	700.00	29.70	2
….	…..	…..	…..	…..	…..	…..
397	28.16	66.09	382.62	0	30.40	12
398	27.59	68.73	380.38	0	29.68	12
399	27.06	70.84	390.78	0	29.08	12
400	26.65	71.60	384.54	0	28.57	12
401	26.14	68.95	396.33	0	27.95	12

One kind of normalization that can scale all dataset values to have values between 0 and 1 is called MinMaxScaler. The MinMaxScaler normalization method is shown in Equations (1) and (2) [13].



```
[[6.40856672e-01 7.17185386e-01 7.98983898e-01 1.16312830e-06
  7.16453867e-01]
 [9.11037891e-01 4.32115471e-01 7.90439401e-01 2.06626315e-02
  9.43868947e-01]
 [6.65568369e-01 6.54600812e-01 5.52956984e-01 4.41322150e-03
  4.94579619e-01]
 ...
 [6.98752648e-01 7.14027966e-01 8.33077908e-01 4.87401738e-03
  7.12117562e-01]
 [6.01317957e-01 7.95218764e-01 8.29194046e-01 6.42612319e-09
  6.34545893e-01]
 [5.75900212e-01 9.05051872e-01 8.66878005e-01 6.42612319e-09
  6.10455312e-01]]
```

Figure 3. Data Preprocessing

$$X_{std} = \frac{(X - X_{min})}{(X_{max} - X_{min})} \tag{1}$$

$$X_{scaled} = X_{std} * (X_{max} - X_{\mathrm{min}}) + X_{min} \tag{2}$$

The dataset of normalized values is provided by Equations (1) and (2). Before being utilized for training and testing, the values are fitted and transformed for the entire dataset.

2.3. Machine Learning

As a subset of deep learning, Artificial Intelligence (AI) includes machine learning as a key component [14]. Because they are very simple to use and all learning algorithm rules are automatically generated by the system with the aid of the Python SciKit Learn package, machine learning algorithms are perhaps the most popular among programmers and engineers. A large range of supervised and unsupervised machine learning algorithms are exposed by Scikit-learn through a standardized, task-oriented interface, making it simple to compare approaches for a particular application [15].

2.3.1 Random Forest

Random Forest is an ensemble learning algorithm that constructs multiple decision trees and combines their outputs to improve predictive accuracy and robustness. The algorithm works by sampling random subsets of the training data and features for each tree, thereby reducing overfitting and variance. Each tree in the forest acts as an independent classifier or regressor, and the final prediction is obtained by aggregating the outputs of all trees, through majority voting for classification or averaging for regression. Mathematically, the Random Forest prediction function f(x) for a given input x can be expressed as [29]:

$$\mathrm{F(x)} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{Ti\ (x)} \tag{3}$$

Where N represents the total number of trees in the forest, and Ti(x) denotes the prediction made by the i^{th} decision tree for the input x.

2.3.2 Decision Tree

The Decision Tree algorithm is a supervised learning method used for both classification and regression tasks. It works by recursively partitioning the dataset into subsets based on the attribute that yields the maximum information gain at each step. The main objective of a decision tree is to create a model that predicts the target variable by learning simple decision rules inferred from data features.

At each internal node, the algorithm selects the feature that best splits the data using a statistical measure such as Information Gain (IG), Gini Index, or Gain Ratio. Information Gain is derived from the concept of Entropy (H), which quantifies the impurity or uncertainty of a dataset. The entropy for a dataset S containing c classes is calculated as [30]:

$$\mathrm{H\ (s)} = -\sum_{\mathrm{i=1}}^{\mathrm{c}} \mathrm{pi} \log 2(pi) \tag{4}$$

where pi is the proportion of samples belonging to class i.

2.3.3 Support Vector Machine (SVM)

The SVM is a powerful supervised learning algorithm used for both classification and regression problems. It operates by finding the optimal hyperplane that best separates data points of different classes in a high-dimensional feature space. The optimal hyperplane is the one that maximizes the margin, defined as the distance between the hyperplane and the nearest data points from each class, known as support vectors.

For a given training dataset (xi,yi), where xi $\in R^n$ represents the feature vector and yi $\in\{-1,+1\}$ denotes the class label, the SVM aims to find the decision boundary defined by the following equation [31]:

$$W \cdot x + b=0 \tag{5}$$

2.3.4 Logistic Regression

Logistic Regression is a supervised learning algorithm commonly used for binary and multiclass classification problems. Unlike linear regression, which predicts continuous values, logistic regression predicts the probability of a sample belonging to a particular class. It does this by applying the sigmoid (logistic) function, which maps the output of a linear combination of input features into a range between 0 and 1.

For a given input feature vector $x = [x_1,x_2,\dots,x_n]$, the logistic regression model computes the probability of class membership as follows [32]:

$$P\ (y = 1 \mid x) = \sigma\ (\ w \cdot x+b) = \frac{1}{1+ e^{-(w.x+b)}} \tag{6}$$

2.3.5 Multilayer Perceptron (MLP)

The MLP is a type of artificial neural network (ANN) composed of multiple layers of nodes, including an input layer, one or more hidden layers, and an output layer. Each node (neuron) in a layer is connected to all nodes in the subsequent layer through weighted connections. The MLP learns to approximate nonlinear mappings between input features and output labels through backpropagation and gradient-based optimization. The forward propagation in an MLP can be represented as [33]:

$$\mathrm{Hj} = f \textstyle\sum_{\mathrm{i}=1}^{\mathrm{n}}(\mathrm{wij\ xi} + \mathrm{bj}) \tag{7}$$

2.4. Model Training and Testing

During the model training phase, the model gains knowledge from the processed data. The method by which the model generates predictions using test data is known as model testing. Since Random Forest automatically processes the dataset to produce the best model, these two procedures are crucial.

We employed cross-validation 5 in this procedure. One of the most popular data resampling techniques for model selection and evaluation is cross-validation. Cross-validation can be used to quantify the generalization error of prediction models, avoid overfitting, compare learning algorithms, and adjust the hyperparameters of statistical and machine learning models. The most popular forms of cross-validation, including k-fold, nested, and leave-one-out cross-validation, are introduced in this article, along with their relationship to other data resampling techniques [17].

2.5. Hyperparameter Tuning

The process of selecting the optimal hyperparameter values for a machine learning model to achieve peak performance is known as hyperparameter tuning. For hyperparameter tweaking, we employed grid search and cross-validation to achieve optimal performance. Grid search for hyperparameter tuning is a methodical process that builds a grid from the hyperparameters under consideration, using every conceivable combination to adjust the selected model [18].

2.6. Model Evaluation

The model evaluation step assesses how well the model learns from a particular dataset. Prior to using the model for commercial purposes, this step is crucial. This study evaluates two models: the prediction stage and the confusion matrix stage.

2.6.1 Confusion Matrix

The confusion matrix serves as a metric for assessing how well the algorithm performs machine learning categorization. Positive and negative classes make up the confusion matrix [19-20]. Figure 4 shows that True Positive (TP) refers to the situation in which test data is accurately expected to be positive (+) data and classified as such.

False Positive (FP): This Type I error occurs when a test instance is incorrectly predicted as positive (+) while it actually belongs to the negative (-) class. True Negative (TN): This occurs when test data are correctly predicted as negative (-) and labeled accordingly. False Negative (FN): This Type II error occurs when a test instance is incorrectly predicted as negative (-) while it actually belongs to the positive (+) class

[21–22]. This explanation enables the assessment of a number of evaluation equations, such as recall, accuracy, and precision, as follows [23,25]:

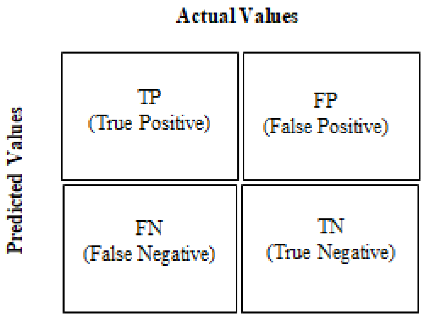


Figure 4. Confusion Matrix

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \tag{3}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{4}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{5}$$

$$\text{F1 Score} = 2 * \frac{precision*recall}{precision+recall} \tag{6}$$

It is possible to formulate the accuracy value of the categorization results in percentage (%) as follows [24]:

$$\text{Accuracy} = \frac{Total\ Prediction\ were\ correct}{Total\ of\ many\ predictions}\ \text{x } 100\% \tag{7}$$

3. RESULTS AND DISCUSSION

In this study, the dataset was divided into two subsets: 80% for training and 20% for testing. The training dataset was used to build and optimize the machine-learning models, enabling them to learn the relationships between environmental parameters such as temperature, humidity, CO_2 concentration, and light intensity and the growth stages of Pakcoy plants. The remaining 20% of the data was reserved exclusively for testing to evaluate the generalization capability of the models on unseen data. This 80/20 division provides a balanced trade-off between training sufficiency and unbiased model evaluation, ensuring that performance metrics such as accuracy, precision, recall, and F1-score reflect the true predictive capability of the algorithms.

Table 3 is the hyperparameter configurations for each machine learning algorithm used in this study. The parameters were optimized through 5-fold cross-validation to achieve the best model performance, including the number of estimators and maximum depth for Random Forest, kernel type and regularization parameter for SVM, splitting criterion for Decision Tree, learning rate for Logistic Regression, and number of hidden layers and activation functions for MLP.

Table 3. Hyperparameter

Model	Result Hyperparameter
SVM	estimator: 100, Kernel: RBF, estimator gamma: scale
RF	random_state: 42, min_samples_leaf: 1, min_samples_split: 5, estimators: 100, bootstrap: True, max_depth: None
DT	criterion: entropy, min_samples_leaf: 1, min_samples_split:2
LR	C: 10, estimator__penalty: l2
MLP	activation: relu, alpha: 0.0001, hidden_layer_sizes: 50, learning_rate: constant, solver: adam

Several options are provided for the random forest approach throughout the training process: bootstrap between True and False, random state 42, max depth between 10 and 30, min samples split between 2 and 10, and min samples leaf between 1 and 4.

Table 4. Results of experiment

Metode	Precision	Recall	F1-score	Accuracy
SVM	0.65	0.58	0.57	0.53
Random Forest	0.86	0.87	0.86	0.84
Decision Tree	0.29	0.31	0.22	0.3
Logistic Regression	0.34	0.26	0.2	0.26
MLP	0.36	0.36	0.34	0.38
Priyambodo et al [27]	-	-	-	0.79

Based on the results as shown in Table 4 and Figure 5**,** the Random Forest algorithm achieved the best overall performance, with a precision of 0.86, recall of 0.87, F1-score of 0.86, and an accuracy of 0.84 (84%). These results demonstrate that the Random Forest model is highly effective in classifying Pakcoy plant conditions, showing strong consistency and robustness. The high recall value indicates that the model successfully identifies most of the correct plant condition classes, while the high precision value suggests a low number of false positives. In contrast, the SVM achieved a precision of 0.65, recall of 0.58, F1-score of 0.57, and accuracy of 0.53, showing moderate performance compared to Random Forest. The Decision Tree, Logistic Regression, and MLP methods exhibited relatively low performance, with accuracy values below 0.40, indicating that they were less capable of handling the complex relationships among environmental parameters.

When compared to the previous study by Priyambodo et al. [27], which reported an accuracy of 79% using the SVM method for classifying Pakcoy plant maturity, the proposed Random Forest model in this study achieved a higher accuracy of 84%. This improvement of approximately 5% highlights the superior capability of the Random Forest approach in modeling complex, nonlinear relationships within the greenhouse environmental data collected through IoT-based systems.

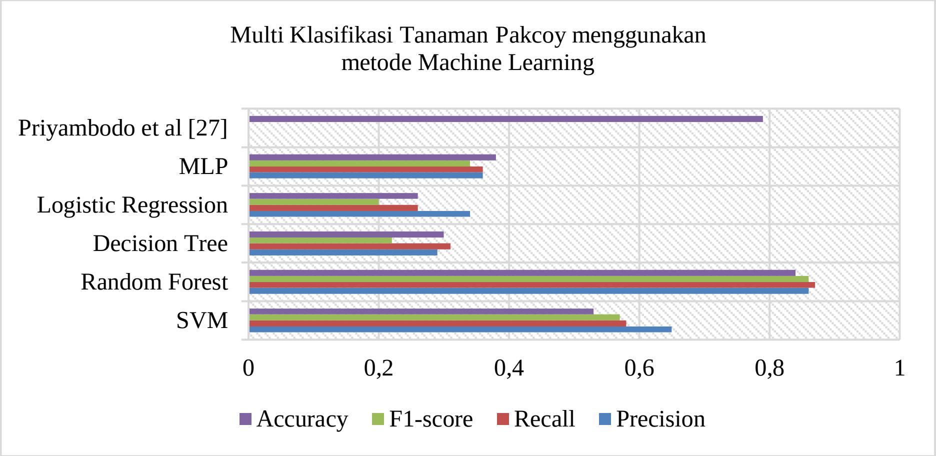


Figure 5. Graph of Results

Figure 5 illustrates the performance comparison of several machine learning algorithms in the multi-classification of Pakcoy plants using four evaluation metrics: accuracy, F1-score, recall, and precision. Among the tested models, the Random Forest algorithm demonstrates the best overall performance, achieving the highest values across all metrics (around 0.85–0.9), indicating its strong capability in accurately classifying Pakcoy growth stages. The SVM method ranks second with moderate precision and recall values (around 0.6), while the MLP model shows fair results with metric values around 0.35–0.4. Logistic Regression and Decision Tree perform lower, suggesting limited classification accuracy for this dataset. Compared to the reference study by Priyambodo et al. [27], Random Forest provides more consistent and reliable results across all evaluation measures, making it the most effective method for Pakcoy plant classification.

4. CONCLUSION

The Random Forest algorithm achieved the highest overall performance, with a precision of 0.86, a recall of 0.87, an F1-score of 0.86, and an accuracy of 0.84 (84%). With high and balanced precision, recall, F1-score, and accuracy values, Random Forest proved to be the most effective approach. Although the SVM method also performed well, its results were still inferior to Random Forest. The relatively poor performance of the Decision Tree and Logistic Regression models may be attributed to their limited ability to manage the complexity and nonlinearity of the greenhouse dataset. Consequently, this study recommends the Random Forest algorithm as the most suitable for classifying Pakcoy plant conditions using IoT-based environmental data. In conclusion, Random Forest demonstrated superior robustness and predictive capability compared to other models. For future work, the system's performance can be further enhanced by applying advanced ensemble or deep learning methods, such as Gradient Boosting, XGBoost, or hybrid neural network approaches.

REFERENCES

[1] Lapcharoensuk, R., Fhaykamta, C., Anurak, W., Chadwut, W., & Sitorus, A. (2023). Nondestructive detection of pesticide residue (Chlorpyrifos) on bok choi (Brassica rapa subsp. Chinensis) using a portable NIR spectrometer coupled with a machine learning approach. Foods, 12(5), 955.

[2] Priyambodo, L., Fuadi, H. L., Nazhifah, N., Huzaimi, I., Prawira, A. B., Saputri, T. E., ... & Goran, P. K. (2022). Klasifikasi Kematangan Tanaman Hidroponik Pakcoy Menggunakan Metode SVM. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 6(1), 153-160.

[3] Junia, L. S. (2017). Uji pertumbuhan dan hasil tanaman pakcoy (Brassica rapa L.) dengan pemberian pupuk organik cair pada system hidroponik. Agrifor, 16(1), 65-74.

[4] Mardilla, M., & Pratiwi, A. (2021). Budidaya tanaman pakcoy (Brassica rapa subsp. chinensis) dengan teknik vertikultur pada lahan sempit di Kelurahan Penaraga Kecamatan Raba Kota Bima. Jurnal Pengabdian Magister Pendidikan IPA, 4(1).

[5] Purwasih, R. (2019). Pemanfaatan lahan pekarangan untuk budi daya sayuran secara hidroponik di Kecamatan Sungailiat, Kabupaten Bangka, Provinsi Kepulauan Bangka Belitung. Agrokreatif: Jurnal Ilmiah Pengabdian Kepada Masyarakat, 5(3), 195-201.

[6] Rahmadhani, L. E., Widuri, L. I., & Dewanti, P. (2020). Kualitas mutu sayur kasepak (kangkung, selada, dan pakcoy) dengan sistem budidaya akuaponik dan hidroponik. Jurnal Agroteknologi, 14(01), 33-43.

[7] Suranata, I. W. A., & Prathama, I. G. H. (2021). Arsitektur Moisture Meter dengan Capacitive Sensing dan Serverless IoT untuk Hidroponik Fertigasi. J. RESTI (Rekayasa Sist. dan Teknol. Informasi), 1(10), 1-3.

[8] Uchiyama, R., Yamaguchi, S., & Takahashi, Y. (2019, October). Solar power generation for compact hydroponic plant cultivation system. In 2019 19th International Conference on Control, Automation and Systems (ICCAS) (pp. 827-832). IEEE.

[9] Afandi, M. A., Hikmah, I., & Agustinah, C. (2021). Microcontroller-based Artificial Lighting to Help Growth the Seedling Pakcoy. Jurnal Nasional Teknik Elektro.

[10] Aliac, C. J. G., & Maravillas, E. (2018, November). IOT hydroponics management system. In 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (pp. 1-5). IEEE.

[11] Muzakir, A. (2021). Perangkat Lunak Mobile Untuk Mendeteksi Daun Pada Tanaman Menggunakan Algoritma K-Nearest Neighbor (K-NN). Journal of Information Technology Ampera, 2(2), 117-126.

[12] Sarifah, L., Sa'adah, L., & Setiana, I. (2025). Comparison of Extreme Learning Machine (ELM) and Multi-Support Vector Machine (Multi-SVM) Methods in Herbal Plants Identification. Jurnal Matematika, Statistika dan Komputasi, 21(2), 354-367.

[13] Deepa, B., & Ramesh, K. (2022). Epileptic seizure detection using deep learning through min max scaler normalization. International journal of health sciences, (I), 10981-10996.

[14] Zhou, Z. H. (2021). Machine learning. Springer nature.

[15] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

[16] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[17] Berrar, D. (2019). Cross-validation.

[18] Ogunsanya, M., Isichei, J., & Desai, S. (2023). Grid search hyperparameter tuning in additive manufacturing processes. Manufacturing Letters, 35, 1031-1042.

[19] Fuadi, A. Z., Haq, I. N., & Leksono, E. (2021). Support Vector Machine to Predict Electricity Consumption in the Energy Management Laboratory. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 5(3), 466-473.

[20] Lamasigi, Z. Y., Hasan, M., & Lasena, Y. (2020). Local Binary Pattern untuk Pengenalan Jenis Daun Tanaman Obat menggunakan K-Nearest Neighbor. ILKOM Jurnal Ilmiah, 12(3), 208-218.

[21] Ting, K. M. (2016). Confusion matrix. In Encyclopedia of machine learning and data mining (pp. 1-1). Springer, Boston, MA.

[22] Luque, A., Carrasco, A., Martín, A., & de Las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognition, 91, 216-231.

[23] Nugroho, A., Gumelar, A. B., Sooai, A. G., Sarvasti, D., & Tahalele, P. L. (2020). Perbandingan Performansi Algoritma Pengklasifikasian Terpandu Untuk Kasus Penyakit Kardiovaskular. J. RESTI

(Rekayasa Sist. dan Teknol. Informasi), 4(5), 998-1006.

[24] Lamasigi, Z. Y., Hasan, M., & Lasena, Y. (2020). Local Binary Pattern untuk Pengenalan Jenis Daun Tanaman Obat menggunakan K-Nearest Neighbor. ILKOM Jurnal Ilmiah, 12(3), 208-218.

[25] Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC genomics, 21(1), 6.

[26] O. Mentari., Wibowo, A. S., Adli, M. Z., & Toha, A. M. (2025). Multi Classification of Strawberry Leaves Using Support Vector Machine (SVM) Method on Smart Greenhouse Plants Based on Internet Of Things (IoT). Lontar Komputer: Jurnal Ilmiah Teknologi Informasi, 16(2), 07-07.

[27] Priyambodo, L., Fuadi, H. L., Nazhifah, N., Huzaimi, I., Prawira, A. B., Saputri, T. E., ... & Goran, P. K. (2022). Klasifikasi Kematangan Tanaman Hidroponik Pakcoy Menggunakan Metode SVM. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 6(1), 153-160.

[28] Kumar, A., Kumar, A., Kumar, R., & Kukreja, V. (2024, March). Cabbage Diseases Identification: A Hybrid CNN and Random Forest Approach for Multi-Classification. In 2024 International Conference on Automation and Computation (AUTOCOM) (pp. 24-28). IEEE.

[29] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

[30] Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., & Brown, S. D. (2004). An introduction to decision tree modeling. Journal of Chemometrics: A Journal of the Chemometrics Society, 18(6), 275-285.

[31] Wang, H., & Hu, D. (2005, October). Comparison of SVM and LS-SVM for regression. In 2005 International conference on neural networks and brain (Vol. 1, pp. 279-283). IEEE.

[32] LaValley, M. P. (2008). Logistic regression. Circulation, 117(18), 2395-2399.

[33] Popescu, M. C., Balas, V. E., Perescu-Popescu, L., & Mastorakis, N. (2009). Multilayer perceptron and neural networks. WSEAS Transactions on Circuits and Systems, 8(7), 579-588.