

Lettuce life stage classification from texture attributes using machine learning estimators and feature selection processes



Sandy Cruz Lauguico ^{a,1,*}, Ronnie II Sabino Concepcion ^{a,2}, Jonnel Dorado Alejandrino ^{a,3}, Rogelio Ruzcko Tobias ^{a,4}, Elmer Pamisa Dadios ^{b,5}

^a Electronics and Communications Engineering Department, De La Salle University, Manila, Philippines

^b Manufacturing Engineering and Management Department, De La Salle University, Manila, Philippines

¹ sandy_lauguico@dlsu.edu.ph; ² ronnie_concepcionii@dlsu.edu.ph; ³ jonnel_alejandrino@dlsu.edu.ph; ⁴ rogelio_tobias@dlsu.edu.ph;

⁵ elmer.dadios@dlsu.edu.ph

* corresponding author

ARTICLE INFO

Article history

Received February 12, 2020

Revised March 18, 2020

Accepted March 23, 2020

Available online July 12, 2020

Keywords

Aquaponics

Feature selection

Growth stage classification

Machine learning

Machine vision

ABSTRACT

Classification of lettuce life or growth stages is an effective tool for measuring the performance of an aquaponics system. It determines the balance in water nutrients, adequate temperature and lighting, other environmental factors, and the system's productivity to sustain cultivars. This paper proposes a classification of lettuce life stages planted in an aquaponics system. The classification was done using the texture features of the leaves derived from machine vision algorithms. The attributes underwent three different feature selection processes, namely: Univariate Selection (US), Recursive Feature Elimination (RFE), and Feature Importance (FI) to determine the four most significant features from the original eight attributes. The features selected were used for training four estimators from Decision Trees Classifier (DTC), Gaussian Naïve Bayes (GNB), Stochastic Gradient Descent (SGD), and Linear Discriminant Analysis (LDA). The models trained using DTC and SGD were then optimized as they have hyperparameters for tuning. A comparative analysis among Machine Learning (ML) algorithms was conducted to identify the best-performing model with the given application. The best features were derived from US and FI as they have the same top four features using the DTC estimator optimized with the hyperparameters tuned to max depth having 5, criterion equated to 'Gini', and splitter was set to 'Best'. The accuracy obtained from cross-validation evaluation resulted in 87.92%. Considering consistency with hold-out validation, LDA outperforms optimized DTC even with lower accuracy of 86.67%. This accuracy of LDA outperformed DTC due to its sufficient fit for generalizing the testing data on classifying lettuce growth stage.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



1. Introduction

Smart aquaponics management is significant in providing efficient resource consumption. For this to realize, cultivars' evaluation is a necessary step to identify the system's performance. Some of the subsystems in the aquaponics setup involve fish feeding systems, irrigation control, nutrient mixture automation, adaptive temperature maintenance, and lighting systems. Determining the conditions of cultivars can be utilized as feedback on how these subsystems are utilized and how they performed to further optimize the technology in adapting to the needs of a high-quality and high-quantity aquaponics' products.

Aquaponics is the integration between a hydroponic unit and an aquaculture unit [1]. Numerous problems are existing in maintaining a productive aquaponics system. One highlight is energy consumption, as this system heavily relies on an artificial environment powered by electronic devices that are energy operated. Le *et al.* [2] focused on providing numerical investigation for a Recirculating Aquaponic System (RAS). The heating method for this system consumes a high amount of energy for an electric heater that maintains the water temperature to achieve optimum temperature for cultivar growth. Automating and monitoring the environment of a smart aquaponics is another challenge to be considered [3]. It includes numerous products not only limited to plants but also aquatic cultivars such as fish, which is the automation system must consider the living necessities of both the products from the hydroponics and aquaculture units.

With the problems in hand, monitoring the performance and efficiency of the subsystems (i.e., energy system and automation) is indispensable. A way to do this is by looking at the cultivars to establish a relationship between the subsystem and its performance. In this study, the cultivar focuses on the hydroponic crop that is the lettuce. Evaluating the lettuce involved the study of its growth and life stages with relation to different physical features. A study conducted used the weight of the lettuce and its number of leaves to assess the quality of light for lettuce growth in a hydroponics setup [4]. Another study considered the stem's diameter, and the plants' height and weight [5] to regulate the data obtained for the Internet of Things (IoT) applications. One of the most significant and most studied attributes of a crop is the texture of its leaves [6] as several studies have proven its capability to determine the life status (i.e., diseases) of different plant species [7]-[10]. Texture, among other features: shape and color, perform better by 5.2% and 6%, respectively, with an identification rate of 92% for image processing recognition [11]. Thus this was utilized for building the dataset as extracted through image processing.

Machine Learning (ML) models integrated with machine vision algorithms have been used for agricultural applications, as this would ensure an increase in crop production through automation [12]. Several studies prove this claim of the integration's effectivity. Mondal *et al.* [13] proposed the use of image processing techniques to extract the morphological features of okra leaves and use the dataset to train Naïve Bayes estimator for classifying the Yellow Vein Mosaic Virus (YVMV) disease resulting to a model with 96.78% accuracy. Another recent research provided a model using Artificial Neural Network (ANN) to assess pasture sward nutrient content on crop leaves related to environmental parameters yielding 94% accuracy [14].

In this paper, the effectivity of machine learning and machine vision integration set the proposal to have an objective of classifying the three main life stages of lettuce through the best-performing machine learning model trained with the training data from the texture features. It was specifically done through texture feature extraction utilizing machine vision algorithms followed by three feature selection processes. We used the selected attributes to train four machine learning estimators that two of them were optimized for improving model performance.

2. Method

The study applied machine vision algorithms for data gathering and extraction of features from the leaves of lettuce at different stages. Data processing techniques were used for feature selection and standardizing their values. Four machine learning algorithms were compared to produce the most appropriate model for the application. Two of the algorithms underwent optimization to improve accuracy for classifying lettuce life stage. Fig. 1 shows the system architecture.

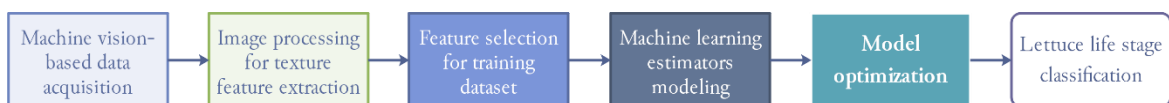


Fig. 1. System architecture.

2.1. Data Gathering and Extraction

Data were gathered for ten weeks through a smartphone camera, capturing images of lettuce planted in August of 2019 from a smart aquaponics farm in Morong, Rizal, Philippines. Thirty different lettuce crops were captured each week, yielding to a total of 300 sample instances. The images underwent image processing algorithms. The process involved machine vision techniques such as background removal and conversion of RGB images to Gray. To extract the features, Haralick Texture Feature analysis was utilized. This algorithm is a statistical entity used to emphasize certain texture properties of an image. It computes for the rotation-invariant value with a recent method of using image rotation of an isolated microstructural object [15]. In Fig. 2 is shown one of the layers of the hydroponics unit of the aquaponics system, while Fig. 3 shown one of the images acquired that went through Haralick texture feature extraction. The Haralick function calculated the Gray Level Co-occurrence Matrix (GLCM) of the images processed. GLCM indices are utilized for quantifying heterogeneous surface patterns and roughness in digital images, further highlighting specific properties of texture [16].

From the function as mentioned above, eight unique features were derived, namely: contrast, correlation, energy, homogeneity, entropy, variance, first information measured correlation, and second information measured correlation.



Fig. 2. Hydroponics unit setup.

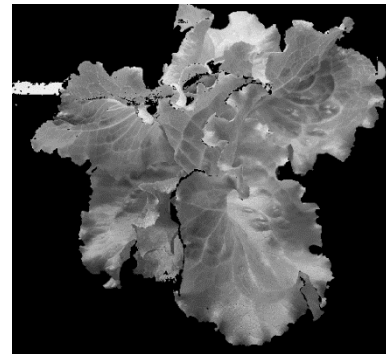


Fig. 3. Image processed through Haralick.

2.2. Feature Selection

Feature selection utilized for texture features is important in classification problems as this can improve classification accuracy while selecting the most relevant attributes of a dataset and decreasing computation time [17]. There are three feature selection methods done in identifying the four most significant attributes from the original eight features extracted from the data. These are Univariate Selection (US), Recursive Feature Elimination (RFE), and Feature Importance (FI). The US is a technique commonly used for developing a multigene predictor. The algorithm determines genes that have P-values lower than a cutoff point compared to genes and survival under univariate Cox models. Utilizing this technique would pick features that are singly related to survival RFE [18]. On the other hand, It is an algorithm that uses estimators to identify the most significant features. Initially, the dataset is trained with a specific estimator, then the most important features are determined through implementing a weight-based backward feature elimination or selection method. This is done by removing one feature for each iteration that the estimator runs [19]. In this study, the estimator for RFE used was Logistic Regression (LR) as it relies on probability for assigning examinations for which data belong in a discrete class [20]. FI was also used as this is capable of ranking the machine learning's feature vector elements for the significance of the model's accuracy [21].

Fig. 4 to Fig. 6 showed the results of the three feature selection techniques used to pick features with the highest k scores. The figures only show 7 features out of 8 because First and Second Information Measure Correlation has a significant of -0.974. The negative value corresponds to an inversely proportional relationship as the first information measure correlation has negative integers. The negative integers cannot be processed using the feature selection process. However, it does significantly affect the model as it is related to the second information measure correlation, which could already represent two attributes into one. It can be understood from the figures that though the order of significant attributes

from both US and FI are different, the four most significant features with the highest k scores are the same, namely: Variance_H4, Entropy, Energy, and the Information Measure Correlation. From RFE, the top four attributes are the ones with the lowest k scores: Information Measure Correlation, Variance_H4, Entropy, and Homogeneity. Thus, two sets of datasets are used for the training of the estimators to produce intelligent models. Only four attributes are considered in the study as using all the features resulted in a significant difference in the cross, and hold-out validation performances were 87.92% and 76.67% respective accuracy. Thus, the model is not properly fit. Four attributes are selected as the other three attributes in the US have 0 k scores, setting a standard for RFE and FI.

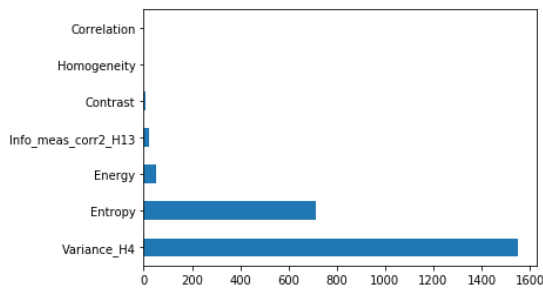


Fig. 4. Univariate selection k score result.

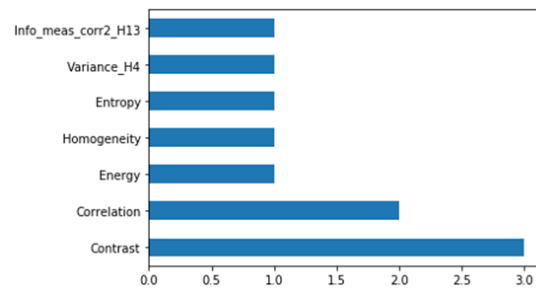


Fig. 5. Recursive feature elimination k score result.

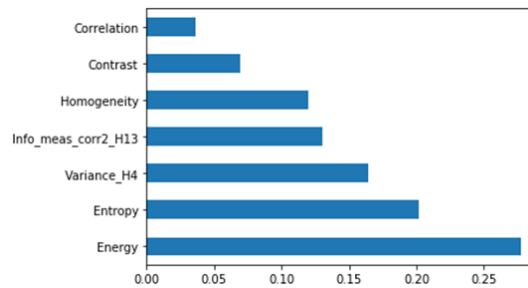


Fig. 6. Feature importance k score result.

2.3 Machine Learning Models

Four ML estimators were used to produce different models for lettuce growth stage classification using the texture attributes with consideration of the four most important features selected beforehand. These four models are Decision Trees Classifier (DTC), Gaussian Naïve Bayes (GNB), Stochastic Gradient Descent (SGD), and Linear Discriminant Analysis (LDA).

Decision trees pattern itself to a structure of a tree on which it has a root node containing all the samples of the feature with the most significant contribution for the classification and expanding to either binary or non-binary classifications resulting in the internal nodes. Each internal node represents each feature from the given dataset. The internal nodes can be composed of a single layer or multiple layers depending on the variables needed to be considered in producing a final decision. The process of classifying relies on a top-down recursive system. These internal nodes branch out to leaf nodes according to the results of the attribute test, which are the final classifiers [22]. DTC structure is shown in Fig. 7. Equation (1) defines the pruning process for minimizing the pruning parameter. Training of the whole classification learner came after pruning and succeeded by performance measurement [23].

$$P = \frac{err(prune(T,t),S) - err(T,S)}{|leaves(T)| - |leaves(prune(T,t))|} \quad (1)$$

Utilizing the default DTC hyperparameters, the features selected from the US, RFE, and FI were used to train the estimator. Since the US and FI have the same attributes selected, only two models were produced from DTC. Fig. 7 represents one of the two decision trees modeled. Shown below is the tree trained model with the four features from RFE. It started with a single attribute, which is entropy as a

root node, implying to be the most significant contributor for the model. The root node branches into a binary tree, in which the true value corresponds to another feature is the measured information correlation, and the false value corresponds to entropy. The DTC consists of 9 layers of internal nodes until reaching the final leaf nodes.



Fig. 7. Decision tree model using default hyperparameters.

Gaussian Naïve Bayes (GNB) is one of the simplest and most effective machine learning algorithms for classification [24]. It was implemented with the problem as it can individualize the attributes that may or may not be dependent and correlated to each other [25] instead of what DCT does. So, It can be determined the independent contribution of each feature. The method assesses each category depends on its prior knowledge by estimating the probability of a new observation from a predefined category [26].

Chen *et al.* [27] similarly applied the method that proposed two approaches for exploring the preferred prior GNB settings considering the individual impacts of the predictors. Shown in equation 2 is the concept of Naïve Bayes [28].

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad (2)$$

where $P(c|x)$ is the posterior probability of class (c , target) given predictor (x , attributes), $P(c)$ is the conditional probability of the class, $P(x|c)$ is the likelihood that is the probability of predictor given class, and $P(x)$ is the probability of the observed pulse or predictor.

With the concept, the trained model depends on the likelihood or probability of one attribute to produce a classification whether the lettuce is under vegetative, head development, or harvest without considering its relation to the other three available attributes, which infers to the algorithms' capability for mutual independency. The other three attributes also underwent an independent GNB process to determine their individual contributions. A classification report is shown in Table 1 and Table 2 to compare the performance of GNB using US or FI and RFE features.

It can be inferred from the tables using two sets of different attributes that the stage of Head Development was the most precisely classified among all lettuce life stages, with an average of 98%. The Vegetative stage comes in second with an average precision of 92%, and the Harvest stage is the most difficult for the model to classify with an average precision of 71.5%.

Table 1. Classification Report for GNB Models Using Features from US or FI.

Growth Stage	Performance Metrics		
	Precision	Recall	F1 Score
Vegetative	92%	100%	96%
Head Development	96%	73%	83%
Harvest	71%	94%	81%

Table 2. Classification Report for GNB Models Using Features from RFE.

Growth Stage	Performance Metrics		
	Precision	Recall	F1 Score
Vegetative	92%	100%	96%
Head Development	100%	73%	85%
Harvest	72%	100%	84%

Stochastic Gradient Descent (SGD) is an iterative ML algorithm that determines the minimum point of an objective function by starting at random points in a given function equated to zero until the slope arrives at the lowest point [29]. The algorithm can address the issue of high computational cost by having faster convergence [30]. To minimize an objective function that has the form of the sum shown in (3), which w is minimized to estimate the parameter $Q(w)$, and the parameter Q_i represents the observation for each iteration when training the dataset.

$$Q(w) = \frac{1}{n} \sum_{i=1}^n Q_i(w) \quad (3)$$

It can be drawn from the confusion matrices in Fig. 8 and Fig. 9 that using US and FI features produced a better SGD model than from RFE features. It was able to classify the Harvest stage 15 times out of 18 while using RFE was not able to classify any sample with the Harvest stage. For the Head Development stage, US and FI features were able to train a model with 83.33% accuracy while utilizing RFE yielded to 96.67% accuracy. Lastly, the former features for classifying the Vegetative stage obtained 83.33% while the latter features were 100%. The US and FI attribute trained a model with an average of 83.33% for each classification, while RFE was only 65.56%, despite having a better performance with classifying Head Development and Vegetative stages, the model was not successful in correctly classifying any sample from the Harvest stage.

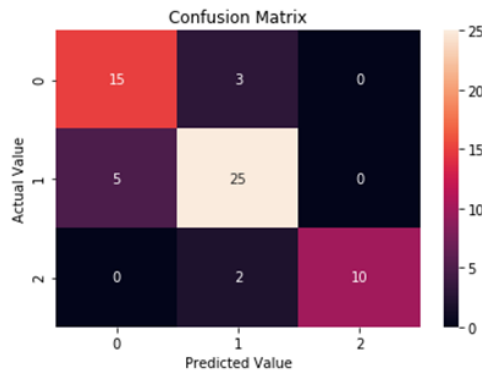


Fig. 8. Default SGD using US or FI confusion matrix.

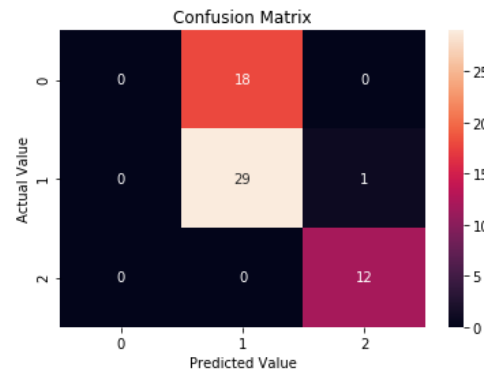


Fig. 9. Default SGD using RFE confusion matrix.

Linear Discriminant Analysis (LDA) is often used as a feature reduction algorithm, but it can also be used as an effective estimator for classifying problems. It is considered optimal Naïve Bayes' classifier for discrete classification if the assumptions regarding normality and homoscedasticity hold. LDA follows two general assumptions. First, the conditional probabilities $p(x|C_i)$ and $p(x|C_2)$ have multivariate normal distributions, and the second, the two classes have equal covariance matrices, also known as homoscedasticity [31]. Fig. 10 shows a comparative plot for the testing data and the predicted data as the output of the model trained from US or FI features. The model using the features from RFE produced an output of predicted values compared to the same testing data is shown in Fig. 11. Both models yielded the same results with an accuracy of 86.67%.

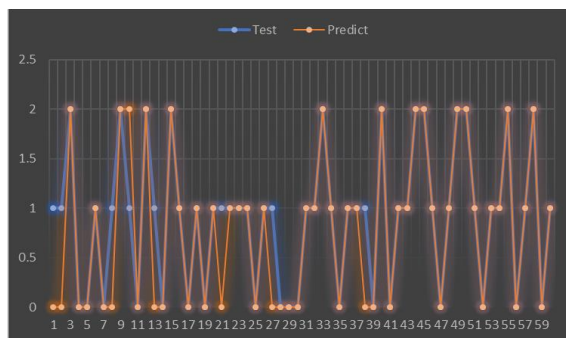


Fig. 10. LDA test vs. predicted values using US or FI.

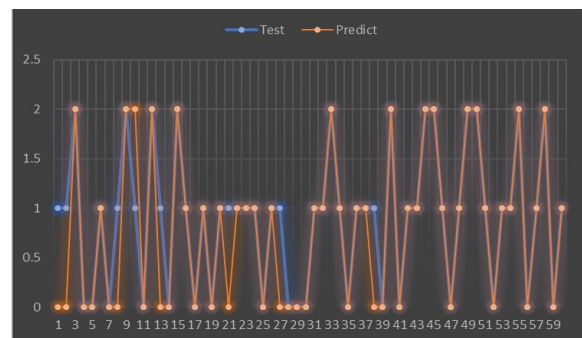


Fig. 11. LDA test vs. predicted values using RFE.

2.4 Optimization

Optimization is a method for determining the best combination of hyperparameters for modeling a certain machine learning estimator that relies on the specific dataset used for training [32]. This process does not guarantee improved accuracy, mainly if the default parameters can already produce the model's highest accuracy. However, optimization can help determine which combination can still further improve the model. Out of the four estimators, only two of them, DTC and SGD, were optimized as the other two do not have hyperparameters for tuning.

Fig. 12 shows one of the optimized DTC models that the root node started with one attribute from RFE, which is the variance extracted from the texture feature. It implies that, for the optimized model, a variance is the most significant contributor among the features for the optimized model trained, unlike the model trained from the default parameters on which entropy was the most significant contributor. The root node branches out into a binary tree, deciding if it falls under information measure correlation or entropy, branching out to several binary layers. It overall consists of four internal nodes before deriving to an accurate decision, which is a better model in comparison to the model with default parameters as

it had nine internal nodes implying higher computational cost. The leaf nodes concluded on differentiating the three life stages according to the series of decisions based on the training of attributes.

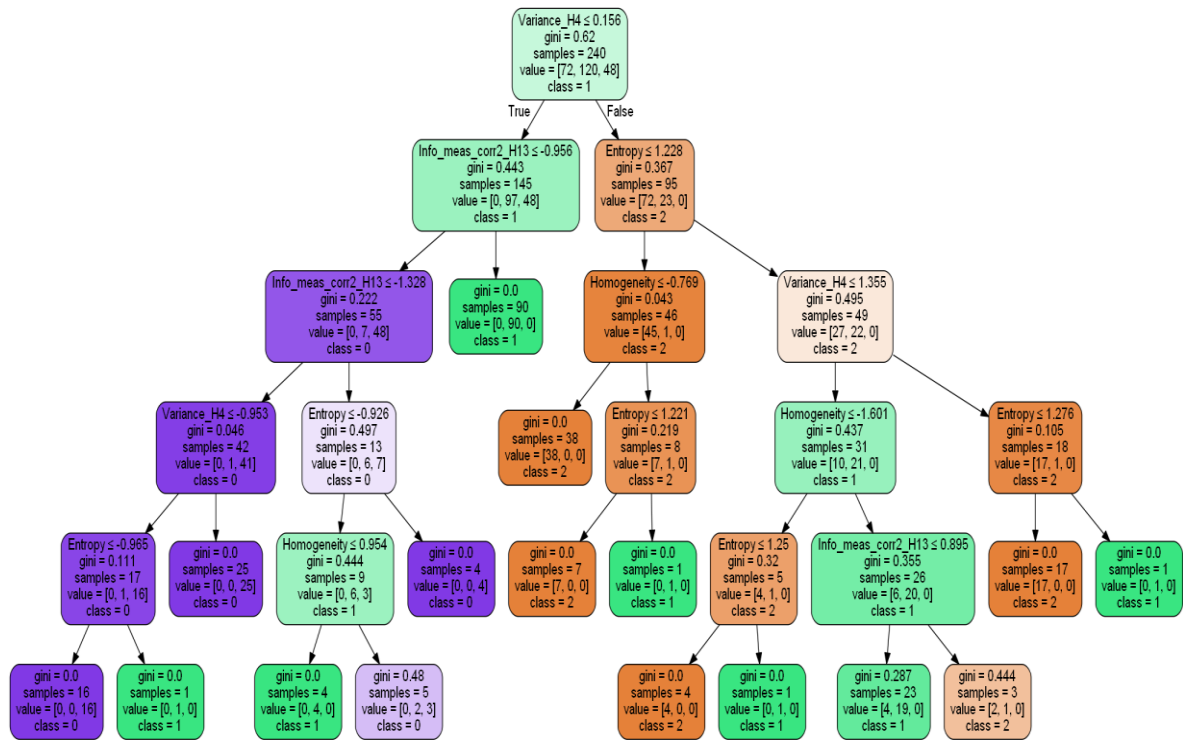


Fig. 12. Decision tree model using optimized hyperparameters.

DTC optimization was done through GridSearchCV. Originally it had default parameters of criterion = ‘gini’, splitter = ‘best’, and max depth = ‘none’. Tuning the hyperparameters resulted in a combination of the criterion being set to ‘gini,’ max depth equal to 5, and the splitter set to ‘best’ with the best accuracy score of 90.67%.

Optimization of SGD was also done through GridSearchCV. Initially, its default parameters are alpha = 0.0001, loss = ‘hinge’, and penalty = ‘l2’. Tuning defines hyperparameters set to alpha equal to 0.0001, loss set to ‘log,’ the penalty set to ‘elasticnet,’ and the l1 ratio equal to 0.3. This combination yielded to the best accuracy score of 86.67%. Fig. 13 and Fig. 14 are shown the confusion matrices. The average accuracy for each lettuce life stage for the attributes from US or FI is 82.22%, implying that the non-optimized model was better by 1.11%. However, the optimized model trained with featured derived from RFE significantly increased from 65.56% to 92.22%.

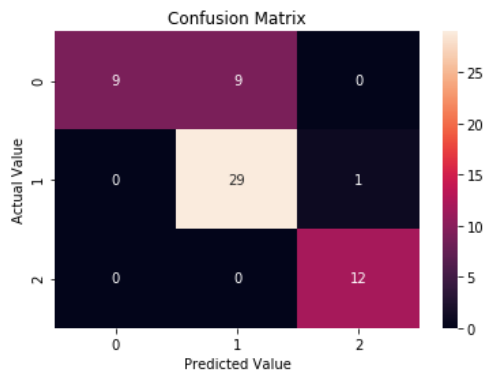


Fig. 13. Optimized SGD using US or FI confusion matrix.

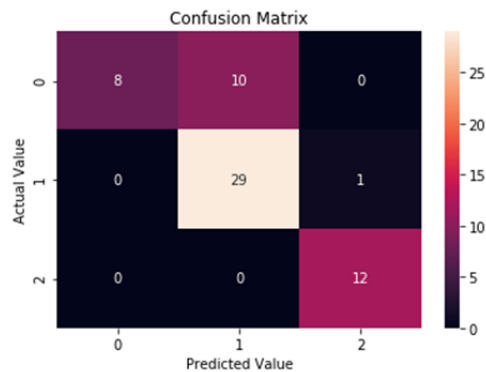


Fig. 14. Optimized SGD using RFE confusion matrix.

3. Results and Discussion

The results for each estimator trained to provide a model were summarized by determining their performance based on cross-validation and hold-out validation. Cross-validation is a method for dividing the dataset into ten splits, and each of the split had experienced being both a training and testing dataset. Hold-out validation simply selects 20% of the data to be the testing dataset while the remaining is for training throughout the validation. Under these validations are metrics as follows: Accuracy denotes the degree of correctness of the predicted values in relation to the actual values. At the same time, the F1 score shows a balance of precision and recall. Precision deals with how often the predicted values are correct when the predicted values are positive. Specificity often describes the predicted is negative when actual is negative, while False Positive Rate (FPR) is how incorrect the specificity is. These are very important metrics as they can show if the model is underfitting or underfit. For DTC and LDA, the optimized models were considered in the discussion of results, while GNB and LDA retain their original model.

3.1. Cross-validation and hold-out validation

Tables 3 and Table 4 are the cross and hold-out validation of models trained with features from US and FI, respectively. Table 3 showed the mean and variance for the accuracy and F1 score. It can be interpreted that DTC is the best performing among the models having the highest accuracy mean and lowest accuracy variance. For its F1 score, it has the highest mean and second to the least variance. In Table 4, it can be interpreted that LDA is the best-performing model having all the best results for every metric.

Table 3. Cross-Validation Result from US or FI.

Estimator	Accuracy Mean	Accuracy Variance	F1 Score Mean	F1 Score Variance
DTC	87.92%	6.31%	87.52%	7.09%
GNB	86.67%	7.41%	86.70%	7.41%
SGD	84.17%	5.20%	82.02%	10.02%
LDA	86.67%	6.40%	86.72%	6.29%

Table 4. Hold-Out Validation Result from US or FI.

Estimator	Accuracy	Specificity	False Positive Rate	Precision	F1 Score
DTC	83.33%	100.00%	0.00%	85.83%	83.33%
GNB	85.00%	94.44%	5.56%	87.54%	85.00%
SGD	83.33%	50%	50%	56.62%	81.85%
LDA	86.67%	100.00%	0.00%	90.06%	86.62%

The cross and hold-out validation of models trained with features from RFE are shown in Table 5 and Table 6. Looking at Table 5, DTC is still the best-performing model in terms of accuracy, while GNB is the best model if the F1 score is considered as it has the highest mean. LDA and GNB are the best models from hold-out validation in Table 6, having the same results for all metrics.

Table 5. Cross-Validation Result from RFE.

Estimator	Accuracy Mean	Accuracy Variance	F1 Score Mean	F1 Score Variance
DTC	87.50%	5.89%	86.50%	6.85%
GNB	87.08%	7.32%	87.16%	7.25%
SGD	84.17%	6.40%	84.02%	5.65%
LDA	86.67%	7.64%	86.67%	7.64%

Table 6. Hold-Out Validation Result from RFE.

Estimator	Accuracy	Specificity	False Positive Rate	Precision	F1 Score
DTC	80.00%	88.89%	11.11%	81.36%	80.09%
GNB	86.67%	100.00%	0.00%	90.06%	86.62%
SGD	81.67%	44.44%	55.56%	85.64%	79.69%
LDA	86.67%	100.00%	0.00%	90.06%	86.62%

3.2. Algorithm Comparative Analysis

Fig. 15 and Fig. 16 summarized the comparison of all the models in terms of the cross and hold-out validation. It can be inferred that the models trained with features from US and FI, that cross-validation result is usually significantly higher than the hold-out validation results with exception to LDA wherein the three metrics have almost the same values. This denotes that DTC, GNB, and SGD models are overfitting, while only the LDA model has the right fit. For the models trained using RFE attributes, GNB and LDA have almost constant metrics between cross and hold-out validation, concluding that the two models have the right fit while the other two models overfit. On the other hand, though DTC using RFE feature-selected attributes has the highest cross-validation accuracy similar to US and FI, its hold-out validation performance is deficient, making the model overfit. SGD using RFE features is also overfit.

Optimized DTC has the highest accuracy in terms of cross-validation for features from all feature selection methods used in training. At the same time, the LDA model is consistent for all the training from the selected features, and it almost has the consistent results for both cross and hold-out validation. To conclude, implying that it was the fittest model for the given dataset with an accuracy of 86.67%.

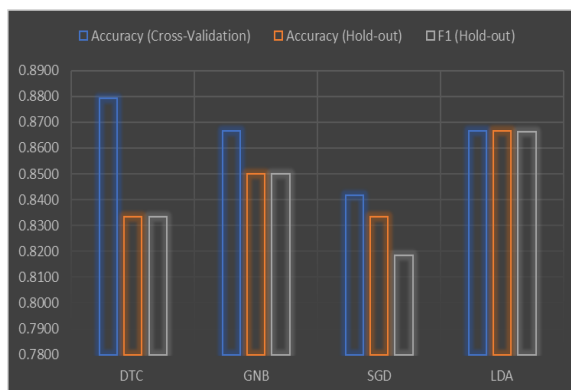


Fig. 15. Model performance comparative analysis for US or FI.

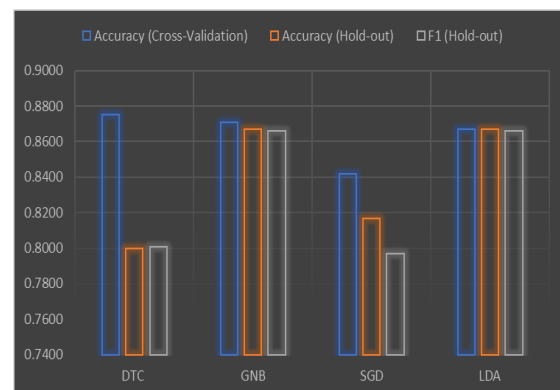


Fig. 16. Model performance comparative analysis for RFE.

4. Conclusion

Classifying lettuce life stage is an important tool for determining its growth rate and the effectiveness of the performance of the entire aquaponics system with its controlled environment. A vision-based algorithm was implemented for extracting eight texture attributes from the lettuce images gathered. The features extracted were selected through three different feature selection techniques, Univariate Selection, Recursive Feature Elimination, and Feature Importance. They yield to two unique sets of four features used for training four estimators. The estimators trained were Decision Classification Tree, Naïve Bayes, Stochastic Gradient Descent, and Linear Discriminant Analysis. Two of them were optimized, and the performances were compared among the models. Though optimized DTC was the best performing during cross-validation with an accuracy of 87.92%, LDA is the most consistent model when looking at both the cross and hold-out validation with a classification accuracy of 86.67%. It is recommended for future works to use other feature selection or reduction methods such as Principal Component Analysis and use the selected attributes to train other machine learning estimators such as

K Nearest Neighbors or Support Vector Machine. Other optimization algorithms can be done as well, like the use of Random Search CV.

Acknowledgment

The researchers would like to thank the Engineering Research and Development Technology (ERDT) of the Department of Science and Technology (DOST) of the Philippines and the De La Salle University (DLSU) Intelligent Systems Laboratory (ISL) for the support they provided.

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding statement. This work has not achieved any fund.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper.

References

- [1] S. A. A. Abusin and B. W. Mandikiana, "Towards sustainable food production systems in Qatar: Assessment of the viability of aquaponics," *Glob. Food Sec.*, no. December 2019, p. 100349, 2020, doi: [10.1016/j.gfs.2020.100349](https://doi.org/10.1016/j.gfs.2020.100349).
- [2] A. T. Le, Y. Wang, L. Wang, V. C. Ta, and D. Li, "Numerical investigation on a low energy-consumption heating method for recirculating aquaponic systems," *Comput. Electron. Agric.*, vol. 169, no. December 2018, p. 105210, 2020, doi: [10.1016/j.compag.2019.105210](https://doi.org/10.1016/j.compag.2019.105210).
- [3] D. Karimanzira and T. Rauschenbach, "Enhancing aquaponics management with IoT-based Predictive Analytics for efficient information utilization," *Inf. Process. Agric.*, vol. 6, no. 3, pp. 375–385, 2019, doi: [10.1016/j.inpa.2018.12.003](https://doi.org/10.1016/j.inpa.2018.12.003).
- [4] R. Chen, H. Liu, S. Song, G. Sun, and R. Chen, "Effects of light quality on growth and quality of lettuces in hydroponic," *2015 12th China Int. Forum Solid State Light. SSLCHINA 2015*, pp. 154–156, 2015, doi: [10.1109/SSLCHINA.2015.7360712](https://doi.org/10.1109/SSLCHINA.2015.7360712).
- [5] S. Gertphol, P. Chulaka, and T. Changmai, "Predictive models for lettuce quality from internet of things-based hydroponic farm," *2018 22nd Int. Comput. Sci. Eng. Conf. ICSEC 2018*, pp. 1–5, 2019, doi: [10.1109/ICSEC.2018.8712676](https://doi.org/10.1109/ICSEC.2018.8712676).
- [6] Z. Tang, Y. Su, M. J. Er, F. Qi, L. Zhang, and J. Zhou, "A local binary pattern based texture descriptors for classification of tea leaves," *Neurocomputing*, vol. 168, pp. 1011–1023, 2015, doi: [10.1016/j.neucom.2015.05.024](https://doi.org/10.1016/j.neucom.2015.05.024).
- [7] M. T. Sánchez, J. A. Entrenas, I. Torres, M. Vega, and D. Pérez-Marín, "Monitoring texture and other quality parameters in spinach plants using NIR spectroscopy," *Comput. Electron. Agric.*, vol. 155, no. June, pp. 446–452, 2018, doi: [10.1016/j.compag.2018.11.004](https://doi.org/10.1016/j.compag.2018.11.004).
- [8] L. Fu *et al.*, "Banana detection based on color and texture features in the natural environment," *Comput. Electron. Agric.*, vol. 167, no. July, p. 105057, 2019, doi: [10.1016/j.compag.2019.105057](https://doi.org/10.1016/j.compag.2019.105057).
- [9] L. Wang, B. Cheng, Z. Li, T. Liu, and J. Li, "Intelligent tobacco flue-curing method based on leaf texture feature analysis," *Optik (Stuttg.)*, vol. 150, pp. 117–130, 2017, doi: [10.1016/j.ijleo.2017.09.088](https://doi.org/10.1016/j.ijleo.2017.09.088).
- [10] T. Pahikkala *et al.*, "Classification of plant species from images of overlapping leaves," *Comput. Electron. Agric.*, vol. 118, pp. 186–192, 2015, doi: [10.1016/j.compag.2015.09.003](https://doi.org/10.1016/j.compag.2015.09.003).
- [11] J. G. Thanikkal, A. Kumar Dubey, and M. T. Thomas, "Whether color, shape and texture of leaves are the key features for image processing based plant recognition? An analysis!," *2017 Recent Dev. Control. Autom. Power Eng. RDCAPE 2017*, vol. 3, pp. 404–409, 2018, doi: [10.1109/RDCAPE.2017.8358305](https://doi.org/10.1109/RDCAPE.2017.8358305).
- [12] T. U. Rehman, M. S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, "Current and future applications of statistical machine learning algorithms for agricultural machine vision systems," *Comput. Electron. Agric.*, vol. 156, no. December 2018, pp. 585–605, 2019, doi: [10.1016/j.compag.2018.12.006](https://doi.org/10.1016/j.compag.2018.12.006).
- [13] D. Mondal, D. K. Kole, and K. Roy, "Gradation of yellow mosaic virus disease of okra and bitter melon based on entropy based binning and Naive Bayes classifier after identification of leaves," *Comput. Electron. Agric.*, vol. 142, no. October, pp. 485–493, 2017, doi: [10.1016/j.compag.2017.11.024](https://doi.org/10.1016/j.compag.2017.11.024).

- [14] M. Safa, K. E. Martin, B. KC, R. Khadka, and T. M. R. Maxwell, "Modelling nitrogen content of pasture herbage using thermal images and artificial neural networks," *Therm. Sci. Eng. Prog.*, vol. 11, no. December 2018, pp. 283–288, 2019, doi: [10.1016/j.tsep.2019.04.005](https://doi.org/10.1016/j.tsep.2019.04.005).
- [15] J. Webel, J. Gola, D. Britz, and F. Mücklich, "A new analysis approach based on Haralick texture features for the characterization of microstructure on the example of low-alloy steels," *Mater. Charact.*, vol. 144, no. May, pp. 584–596, 2018, doi: [10.1016/j.matchar.2018.08.009](https://doi.org/10.1016/j.matchar.2018.08.009).
- [16] Y. Park and J. M. Guldmann, "Measuring continuous landscape patterns with Gray-Level Co-Occurrence Matrix (GLCM) indices: An alternative to patch metrics?," *Ecol. Indic.*, vol. 109, no. October 2019, p. 105802, 2020, doi: [10.1016/j.ecolind.2019.105802](https://doi.org/10.1016/j.ecolind.2019.105802).
- [17] M. K. Uçar, "Classification Performance-Based Feature Selection Algorithm for Machine Learning: P-Score," *Irbm*, vol. 1, pp. 1–11, 2020, doi: [10.1016/j.irbm.2020.01.006](https://doi.org/10.1016/j.irbm.2020.01.006).
- [18] T. Emura, S. Matsui, and H. Y. Chen, "compound.Cox: univariate feature selection and compound covariate for predicting survival," *Comput. Methods Programs Biomed.*, vol. 168, pp. 21–37, 2019, doi: [10.1016/j.cmpb.2018.10.020](https://doi.org/10.1016/j.cmpb.2018.10.020).
- [19] S. Chatterjee, D. Dey, and S. Munshi, "Integration of morphological preprocessing and fractal based feature extraction with recursive feature elimination for skin lesion types classification," *Comput. Methods Programs Biomed.*, vol. 178, pp. 201–218, 2019, doi: [10.1016/j.cmpb.2019.06.018](https://doi.org/10.1016/j.cmpb.2019.06.018).
- [20] S. Belciug, "Logistic regression paradigm for training a single-hidden layer feedforward neural network. Application to gene expression datasets for cancer research," *J. Biomed. Inform.*, vol. 102, p. 103373, 2020, doi: [10.1016/j.jbi.2019.103373](https://doi.org/10.1016/j.jbi.2019.103373).
- [21] J. Heaton, S. McElwee, J. Fraley, and J. Cannady, "Early stabilizing feature importance for TensorFlow deep neural networks," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2017-May, pp. 4618–4624, 2017, doi: [10.1109/IJCNN.2017.7966442](https://doi.org/10.1109/IJCNN.2017.7966442).
- [22] R. F. de Mello, C. Manapragada, and A. Bifet, "Measuring the Shattering coefficient of Decision Tree models," *Expert Syst. Appl.*, vol. 137, pp. 443–452, 2019, doi: [10.1016/j.eswa.2019.07.012](https://doi.org/10.1016/j.eswa.2019.07.012).
- [23] R. S. Concepcion *et al.*, "Trophic state assessment using hybrid classification tree- artificial neural network," vol. 6, no. 1, pp. 1–13, 2019, doi: [10.26555/ijain.v6i1.408](https://doi.org/10.26555/ijain.v6i1.408).
- [24] S. Chen, G. I. Webb, L. Liu, and X. Ma, "A novel selective naïve Bayes algorithm," *Knowledge-Based Syst.*, vol. 192, p. 105361, 2019, doi: [10.1016/j.knosys.2019.105361](https://doi.org/10.1016/j.knosys.2019.105361).
- [25] Y. C. Zhang and L. Sakhanenko, "The naïve Bayes classifier for functional data," *Stat. Probab. Lett.*, vol. 152, pp. 137–146, 2019, doi: [10.1016/j.spl.2019.04.017](https://doi.org/10.1016/j.spl.2019.04.017).
- [26] P. Tsangaratos and I. Iliá, "Comparison of a logistic regression and Naïve Bayes classifier in landslide susceptibility assessments: The influence of models complexity and training dataset size," *Catena*, vol. 145, pp. 164–179, 2016, doi: [10.1016/j.catena.2016.06.004](https://doi.org/10.1016/j.catena.2016.06.004).
- [27] C. Chen, G. Zhang, J. Yang, J. C. Milton, and A. D. Alcántara, "An explanatory analysis of driver injury severity in rear-end crashes using a decision table/Naïve Bayes (DTNB) hybrid classifier," *Accid. Anal. Prev.*, vol. 90, pp. 95–107, 2016, doi: [10.1016/j.aap.2016.02.002](https://doi.org/10.1016/j.aap.2016.02.002).
- [28] D. Petschke and T. E. M. Staab, "A supervised machine learning approach using naïve Gaussian Bayes classification for shape-sensitive detector pulse discrimination in positron annihilation lifetime spectroscopy (PALS)," *Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip.*, vol. 947, no. July, p. 162742, 2019, doi: [10.1016/j.nima.2019.162742](https://doi.org/10.1016/j.nima.2019.162742).
- [29] A. Ma and D. Needell, "Stochastic gradient descent for linear systems with missing data," *Numer. Math.*, vol. 12, no. 1, pp. 1–20, 2019, doi: [10.4208/nmtma.OA-2018-0066](https://doi.org/10.4208/nmtma.OA-2018-0066).
- [30] A. Sharma, "Guided Stochastic Gradient Descent Algorithm for inconsistent datasets," *Appl. Soft Comput. J.*, vol. 73, pp. 1068–1080, 2018, doi: [10.1016/j.asoc.2018.09.038](https://doi.org/10.1016/j.asoc.2018.09.038).
- [31] K. S. Gyamfi, J. Brusey, A. Hunt, and E. Gaura, "Linear classifier design under heteroscedasticity in Linear Discriminant Analysis," *Expert Syst. Appl.*, vol. 79, pp. 44–52, 2017, doi: [10.1016/j.eswa.2017.02.039](https://doi.org/10.1016/j.eswa.2017.02.039).
- [32] W. T. de Sousa Junior, J. A. B. Montevechi, R. de C. Miranda, M. L. M. de Oliveira, and A. T. Campos, "Shop floor simulation optimization using machine learning to improve parallel metaheuristics," *Expert Syst. Appl.*, vol. 150, p. 113272, Jul. 2020, doi: [10.1016/j.eswa.2020.113272](https://doi.org/10.1016/j.eswa.2020.113272).