SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

The Oyster Mushroom Harvesting Determination Based on Image Processing and Multi-Layer Perceptron

Nursuci Putri Husain[1)*], Muh. Ichwan Kadir[2)], Muh. Dzulkifli P[3)]
[1,2,3)] Makassar Islamic University, Indonesia
[1)]nursuciputrihusain.dty@uim-makassar.ac.id, [2)]muhichwank.dty@uim-makassar.ac.id

Submitted :Sept 20, 2024 | **Accepted** : Sept 29, 2024 | **Published** : Oct 2, 2024

Abstract: Oyster mushroom cultivation has become increasingly popular in South Sulawesi due to its high economic potential, but many farmers, including those at Sahabat Jamur cultivation in Parepare, struggle to optimize harvests due to a lack of precise methods for determining the ideal harvest time. Traditional methods rely on visual inspection, often leading to inconsistencies in both timing and quality. This highlights the need for technological solutions to help farmers better identify the optimal harvest period. Leveraging advancements in image processing and machine learning, this study proposes an automated system designed to improve the accuracy and consistency of mushroom harvests. The system analyzes images of oyster mushrooms captured using a digital camera, identifying key visual characteristics such as color, texture, and size to determine maturity. The process involves several stages: image dataset collection, preprocessing, segmentation, morphological operations, feature extraction, and classification using a Multi-Layer Perceptron (MLP). A dataset of 150 images, categorized into 'ready for harvest' and 'not ready for harvest,' was used to test the system, and the classification model achieved an accuracy rate of 96.67%. By utilizing this technology, the system aims to enhance efficiency and consistency in the harvesting process, helping farmers make more informed decisions.

Keywords: Classification, Determination of Harvest, Image Processing, Multi Layer Perceptron, Oyster Mushrooms.

INTRODUCTION

Oyster mushrooms are one of the mushroom species with high economic value and are widely cultivated in Indonesia. The demand for oyster mushrooms continues to rise due to their relatively low price and their status as a rich source of vegetable protein (Amir et al. 2023; Ichwan K and Husain 2021; Prasekti and Nugroho 2021). Sahabat Jamur, a business focused on oyster mushroom cultivation, faces challenges in fulfilling market demand due to limited harvest yields. This is because harvesting oyster mushrooms requires precision and care to avoid damaging mushrooms that are ready for harvest.

To determine whether oyster mushrooms are ready to be harvested, several characteristics can be observed. First, oyster mushrooms that are ready for harvest are typically larger and according to the expected size for a particular variety. In addition, the color of ready-to-harvest oyster mushrooms usually ranges from white to brownish. Regular observation of these characteristics is crucial to determining the right time to harvest the mushrooms, so efforts are needed to develop farmers' abilities to recognize these signs (Umniyatie and Pramiadi 2013).

One of the first steps that can be taken is to utilize an intelligent system for determining oyster mushroom harvest based on image processing. In recent years, there has been significant progress in image processing research, especially in the context of oyster mushroom images (Ayuninghemi et al. 2022; Nurfitri, Pradana, and Widaningrum 2022; Pratomo 2021; Sert and Okumus 2014). One of the segmentation algorithms that has been used in previous research is Canny Detection (Ayuninghemi et al. 2022). Canny detection uses a Gaussian filter to smooth the image before edge detection, so it can reduce noise in the image. However, canny detection makes it difficult to handle hidden or overlapping edges in the image.

Furthermore, the classification algorithm that has been used to classify oyster mushroom images is C.45 (Pratomo 2021). The C.45 method has the advantage of being easy to understand the algorithm and implement. However, despite these advantages, the C.45 also has disadvantages that need to be considered. Decision tree methods, including C4.5, are prone to overfitting, especially if the resulting tree is very complex. Overfitting data

* Corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

can cause classification errors (Pratama, Mutaqin, and Manuela 2023). To increase the accuracy of the classification model, this research uses an image processing-based method and the Multi Layer Perceptron (MLP) classification method. MLP is a type of artificial neural network (ANN) architecture which consists of several layers of interconnected neurons. MLP, especially with the use of regularization techniques such as dropout, can help reduce the possibility of overfitting by reducing model complexity and preventing too strong connections between neurons (Husain and Aji 2019; Pardede and Hayadi 2022).

The model used in this research has several stage: image pre-processing stage, image segmentation stage, morphological operations, feature extraction stage and MLP-based image classification stage. By using image processing and MLP techniques, it is hoped that we will be able to classify oyster mushroom harvest images well, so that it can increase farmers' oyster mushroom harvests and being able to contribute the development of the digital economy.

LITERATURE REVIEW

The development of automated systems for oyster mushroom harvesting has seen substantial progress with the application of image processing and machine learning techniques. Lei Shi, Zhanchen Wei, and colleagues introduced OMC-YOLO, a lightweight detection method designed for classification of oyster mushrooms. This model is efficient, making it suitable for practical, field-based applications where time-sensitive decisions are critical (Shi et al. 2024). While it performs well in controlled environments, the main limitation of OMC-YOLO is its sensitivity to background variations and complex lighting conditions, which can reduce accuracy in dynamic agricultural environments, especially in large-scale farms with diverse setups.

Similarly, Saputra and Ibadillah's study on digital image processing for oyster mushroom harvest determination uses simple image segmentation techniques such as thresholding to differentiate between mushrooms ready for harvest and those that are not (Saputra and Ibadillah 2019). This method is easy to implement and works well under controlled conditions, but it lacks the flexibility to handle variations in background and mushroom appearance, especially in real-world farm environments where lighting and conditions are less predictable. The reliance on basic segmentation limits its scalability and effectiveness in broader agricultural applications.

Other research, such as Ayuninghemi et al.'s use of Canny edge detection for grading and Sert and Okumus's modified K-Means clustering algorithm for cap width measurement, contributes to the development of oyster mushroom classification systems (Ayuninghemi et al. 2022; Sert and Okumus 2014). These segmentation techniques successfully extract key features like cap size and edges to aid in the classification process. However, these methods can struggle with noise, variations in morphology, and inconsistent lighting, limiting their robustness. Integrating these image processing methods with machine learning algorithms like the Multi-Layer Perceptron (MLP) can improve classification accuracy by handling non-linear relationships in data (Husain and Aji 2019; Pardede and Hayadi 2022). Despite the advantages, MLP models require significant computational resources and large, well-labeled datasets, which can pose challenges for small-scale farmers.

METHOD

In this research, the proposed research method consists of several steps as in Fig 1. In the sub-chapters below, each stage will be explained.

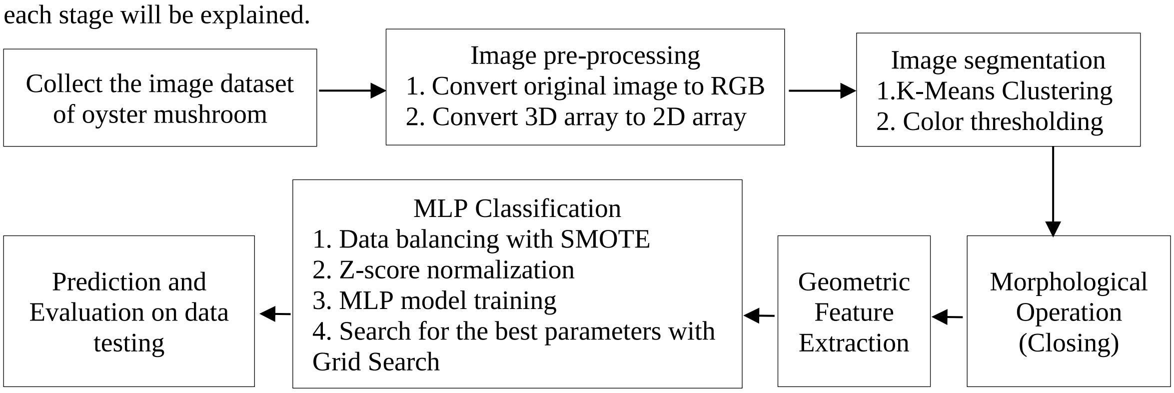

Fig 1. Design of an Oyster Mushroom Harvesting Determination System.

Dataset of Oyster Mushroom

The image collection of oyster mushrooms was conducted at the "Sahabat Jamur" oyster mushroom cultivation site in Parepare City, using prepared equipment including a camera. The primary goal of this collection is to obtain a representative variety of oyster mushroom images, which will later be used in the development of a classification model based on image data. Each image is labeled with the date and the condition of the oyster mushroom. Each image capture session is carried out consistently, such as maintaining the same distance and angle for all photos.

* Corresponding author

 e-ISSN : 2541-2019
p-ISSN : 2541-044X

A total of 150 images were collected, with 75 images of mushrooms ready for harvest and 75 images of those not yet ready. Sample images of the oyster mushrooms can be seen in Fig 2.

Fig 2. Sample images of the oyster mushrooms were directly taken from the Sahabat Jamur cultivation site

Pre-processing Stage

In the proposed pre-processing stage for oyster mushroom images, the original images are converted to RGB format. This step is crucial to ensure that the images are in a consistent color format. Conversion to RGB also facilitates subsequent processing steps, including segmentation and feature extraction, which often require color information in RGB format. After that, the images are transformed from a 3D array into a 2D array, where each row represents an individual pixel with three color values (red, green, and blue). This process flattens the image data, making it easier to process using machine learning algorithms and segmentation techniques, such as K-Means Clustering, which are more effective on data in 2D format.

Image Segmentation Stage

In the image segmentation stage, two methods are employed: K-means clustering and color thresholding. The K-Means algorithm is used to divide the pixels into k different clusters. Each cluster has a centroid, which is the average value of all the pixels in that cluster (Lee, Lee, and Hong 2023). The equation (1) for determining the centroid of a cluster in the *t-th* iteration is as follows:

$$\mu_j^{(t)} = \frac{1}{\left|C_j^{(t)}\right|} \sum_{x_i \in C_j^{(t)}} x_i \qquad (1)$$

Where $\mu_j^{(t)}$ represents the centroid of the *j-th* cluster in the *t-th* iteration, and $C_j^{(t)}$ is the set of pixels that belong to the *j-th* cluster in the *t-th* iteration. Pixels closest to the same centroid are grouped together, resulting in a segmentation that separates objects based on color similarity. Equation (2) is as follows:

$$c_i = \arg min_j \left\|x_i - \mu_j\right\|^2 \qquad (2)$$

Where c_i is the cluster label for the *i-th* pixel, and x_i is the pixel value (consisting of three RGB components). The brightest cluster (based on color intensity) is selected as the area containing the oyster mushroom object. After the initial segmentation using K-Means, a Color Thresholding process is applied to refine the segmentation results. The image is converted from RGB to HSV (Hue Saturation Value) because, in the HSV color space, colors can be more easily separated based on their intensity (Sitanggang, Yuhandri, and Adil Setiawan 2023). Equation (3) is as follows:

$$mask = \begin{cases} 1 & if\ lower\_white \leq HSV \leq upper\_white \\ 0 & if\ not \end{cases} \qquad (3)$$

Where $lower\_white$ and $upper\_white$ are the predefined lower and upper bounds for the color white. Through thresholding, areas with white color intensity are identified, producing a mask that highlights only the white portions of the image. This mask is then combined with the K-Means results to refine the segmentation, ensuring that only areas with the appropriate white color are retained. This process is essential for improving segmentation accuracy by focusing on specific color characteristics, such as the white color often seen in mature oyster mushrooms.

Morphological Operation

In the morphological operation stage, a Closing process is performed. Closing is a combination of dilation and erosion, aimed at removing small gaps within objects and uniting separate areas that should be a single entity(Dhanya and Ramkumar 2023). Using a 5x5 kernel, the closing operation fills these gaps and smooths the edges of the object, resulting in a cleaner mask that is free from noise. The basic formulas for dilation and erosion (4), (5), (6) are as follows:

$$A \oplus B = \{z | (B)_2 \cap A \neq \emptyset\} \qquad (4)$$

Dilation adds pixels to the boundaries of an object in a binary image. In this context, *A* represents the object area, *B* is the structural element (kernel), and ⊕ denotes the dilation operation.

$$A \ominus B = \{z | (B)_2 \subseteq A\} \qquad (5)$$

* Corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Erosion removes pixels from the boundaries of an object in a binary image, where ⊖ represents the erosion operation. Closing is defined as dilation followed by erosion, and the formula is as follows:

$$A \cdot B = (A \oplus B) \ominus B \qquad (6)$$

After the morphological operations, the cleaned mask (*final_mask*) is used to extract the object from the original image by applying the mask to the RGB image. This process is done by creating a black background and displaying only the areas corresponding to the *final_mask*. Technically, only pixels with a value of 1 in the *final_mask* are retained from the original image, while the rest are replaced with the black background. The final result is a segmented image with a black background that clearly and cleanly shows the main object (such as the oyster mushroom). This stage is crucial for ensuring that the object is accurately segmented and ready for the next analysis step, minimizing interference from noise or irrelevant areas. The results of the segmentation and morphological processing of the oyster mushroom image can be seen in Table 1.

Table 1
Sample results of the segmentation and morphological processing

No	Original Image	Segmented Image
1		
2		
3		

Feature extraction stage

The feature extraction stage aims to extract geometric features from images that have gone through a segmentation process and morphological operations. In the initial stage, the process carried out to find contours in the mask that represent the boundaries of the segmented object. Contours are a representation of the boundary lines that surround an object (Kang et al. 2024). If no contours are found, the function returns the default value [0, 0, 0, 0, 0, 0], indicating that no features can be extracted. These features are calculated by the formula:

$$Area = \sum_{\text{contour}} Pixel \qquad (7)$$

$$Perimeter \ = \sum_{\text{contour}} Distance\ between\ points - contour\ points \qquad (8)$$

$$Wide = w, Tall = h \qquad (9)$$

$$Aspect\ Ratio = \frac{Wide}{Tall} = \frac{w}{h} \qquad (10)$$

* Corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

$$White\ Intensity = \frac{\sum white\ pixel\ value}{total\ pixels} \qquad (11)$$

The extracted features provide important information about the geometric characteristics of objects in the image, such as size, shape, and brightness, which are calculated using the aforementioned equations. The area of an object is calculated by adding up the pixels in the contour, giving an idea of how big the object is (7). Perimeter is calculated as the sum of the distances between points along the boundary of an object, which can indicate the complexity or detail of the object's shape (8). Width and Height provide the dimensions of the bounding box surrounding the object (9), while Aspect Ratio, calculated as the ratio between the width and height of the object (10), helps in recognizing the specific shape of the object. White Intensity is calculated as the average white pixel value in the mask, which is relevant for identifying objects based on their whiteness or brightness compared to the background (11).

By extracting and analyzing these features using these equations, the model can be more accurate in classifying oyster mushrooms as suitable for harvest and not suitable for harvest. The results of feature extraction can be seen in Table 2, there are 150 samples of oyster mushroom images, and 6 features were produced.

Table 2
Feature Extraction Results

No	***Area***	***Perimeter***	***Width***	***Height***	***Aspect Ratio***	***White Intensity***
1	27	32.14214	7	14	0.5	0.273
2	96.5	92.38478	36	12	3	0.034
3	1.5	5.414214	2	3	0.6666	0.169
4	0	4	3	1	3	0.056
5	0	2	2	1	2	0.064
6	29	60.14214	28	7	4	0.069
7	32.5	35.07107	5	16	0.3125	0.054
8	6	13.65685	5	5	1	0.026
9	0	0	1	1	1	0.131
10	73	140.6274	68	8	8.5	0.060
…	….	….	….	….	….	….
150	….	….	….	….	….	….

Classification Stage

At this classification stage, the Multi Layer Perceptron (MLP) method is used with several important steps such as data balancing, normalization, MLP model training, searching for the best parameters using Grid Search, and model evaluation. The first step is data balancing using SMOTE (Synthetic Minority Over-sampling Technique), which aims to handle data inequality between the majority and minority classes. SMOTE synthesizes new samples from minority classes using interpolation between data points so that the class distribution becomes more balanced (Chawla et al. 2002). This balancing equation is written as follows:

$$X_{res}, Y_{res} = SMOTE(X_{train}, Y_{train}) \qquad (12)$$

Where X_{res} and Y_{res} are the balanced training data and labels. Next, the data is standardized using the *Z-score* normalization technique, which transforms each feature into a distribution with a mean of 0 and a standard deviation of 1. This normalization ensures that all features are on the same scale, preventing features with larger value ranges from dominating the model. The process is formulated as follows:

$$X_{norm} = \frac{X-\mu}{\sigma} \qquad (13)$$

μ is the mean and σ is the standard deviation of the feature. MLP (Multilayer Perceptron) is an artificial neural network consisting of an input layer, one or more hidden layers, and an output layer. Essentially, MLP uses a non-linear activation function to capture complex relationships within the data. For each neuron *j* in layer *l*, the output is calculated by summing the weighted inputs and adding a bias, followed by applying an activation function:

$$z_j^{(l)} = \sum_{i=1}^{n} w_{ij}^{(l)}\, a_i^{(l-l)} + b_j^{(l)} \qquad (14)$$

Where $z_j^{(l)}$ is the calculated input value of neuron *j* in layer *l*. Then, $w_{ij}^{(l)}$ is the weight connecting neuron *i* in the previous layer $(l-l)$with neuron *j* in layer *l*. The term $a_i^{(l-l)}$ represents the output (activation) of neuron ii in the

* Corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

previous layer, while $b_j^{(l)}$ is the bias of neuron *j* in layer *l*. For the model training, MLP is trained using Grid Search to find the optimal parameters. Grid Search evaluates combinations of parameters such as hidden layer size, activation function, regularization value (α), and learning rate. Grid Search applies 5-fold cross-validation (cv=5) to test each parameter combination. Once the best model is identified, it is evaluated again with 5-fold cross-validation (cv=5), yielding an average score that reflects the stability and performance of the model across different subsets of the training data.

Prediction And Evaluation On Testing Data

The best model is used to make predictions on testing data, and prediction results are evaluated using accuracy metrics, classification reports, and confusion matrices. The equation for finding accuracy is as follows:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{total\ predictions} = \frac{\sum_{i=1}^{n} I(y_i = \dot{y}_i)}{n} \qquad (15)$$

where y_i is the actual label, $\dot{y}_i$ is the predicted label, *I* is an indicator function that takes the value 1 if the prediction is correct and 0 if it is wrong, and n is the number of samples. Furthermore, evaluation metrics such as precision, recall, and F1-score are also used for each class ("Not Ready to Harvest" and "Ready to Harvest"). Precision measures the proportion of positive predictions that are correct, the formula is:

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \qquad (16)$$

Recall measures the model's ability to find all actual positive samples, with the formula:

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \qquad (17)$$

F1-score is the average of precision and recall, formulated as:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (18)$$

Furthermore, a confusion matrix is also used which describes the number of correct and incorrect predictions for each class. This matrix helps in identifying model error patterns more clearly. The confusion matrix has the form:

$$Cunfusion\ matrix = \begin{bmatrix} True\ Negatif & False\ Positif \\ False\ Negatif & True\ Positif \end{bmatrix} \qquad (19)$$

where each value indicates the number of occurrences for each prediction type and their correctness. Using these metrics, the evaluation provides a comprehensive picture of the model's performance in classification, allowing the identification of areas that need to be improved or further calibrated to increase the model's accuracy.

RESULT

In this research, the dataset that we used consisted of 150 samples of oyster mushroom images, consisting of "Ready to Harvest" and "Not Ready to Harvest" classes. To train a classification model using Multi-Layer Perceptron (MLP), a Grid Search approach is used which combines cross-validation techniques to find the best parameters, such as hidden layers, activation function, regularization value (α), and learning rate. Grid Search uses cross-validation with 5 fold (cv=5), which means the dataset is divided into 5 equal subsets, each consisting of 30 samples. In each iteration, 4 subsets are used as training data (120 samples), and 1 subset is used as testing data (30 samples). This process is repeated 5 times so that each sample has the opportunity to be part of the testing data once and part of the training data four times.

Table 3
Example of Classification Results Using the Proposed Model

No	Image	Label	Prediction
1		Ready To Harvest	Ready To Harvest
2		Ready To Harvest	Ready To Harvest

* Corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

			
3		Not Ready To Harvest	Not Ready To Harvest
4		Ready To Harvest	Not Ready To Harvest

After finding the best parameters with Grid Search, the best model is evaluated using cross-validation 5 times (cv=5). In this scheme, the dataset is divided into 5 subsets of 30 samples each. At each evaluation iteration, 4 subsets are used as training data (120 samples), and 1 subset is used as test data (30 samples). The evaluation was repeated 5 times to provide more stable and representative results of the model performance on various subsets of the training data. This cross-validation technique ensures that all samples in the dataset are used alternately as training and test data, which minimizes bias and provides a more accurate estimation of the classification model's performance on data that has never been seen before. Thus, this approach helps to produce a more robust model and good generalization in the classification of oyster mushrooms based on their maturity. Example of classification results using the proposed model can be seen in Table 3.

$$Cunfusion\ Matrix = \begin{bmatrix} 17 & 0 \\ 1 & 12 \end{bmatrix}$$

The classification model performed very well in distinguishing between the “Not Ready to Harvest” and “Ready to Harvest” classes. From the confusion matrix, it can be seen that out of 17 “Not Ready to Harvest” samples, the model predicted all of them correctly (True Negatives), while out of 13 “Ready to Harvest” samples, the model predicted 12 correctly (True Positives) and misclassified 1 “Ready to Harvest” sample as “Not Ready to Harvest” (False Negative).

Table 4
Evaluation Results of Classification Model Test

Metric	**Not Ready to Harvest**	**Ready to Harvest**	***Macro Avg***	***Weighted Avg***
Accuracy	-	-	96.67%	96.67%
Precision	1.0	0.9231	0.9615	0.9667
Recall	1.0	0.9231	0.9615	0.9667
F1-Score	1.0	0.9231	0.9615	0.9667
Support	17	13	30	30

In Table 4, the model's accuracy, which shows the percentage of correct predictions out of the total predictions, is 96.67%, meaning the model correctly predicted 29 out of 30 samples. The precision for the "Not Ready for Harvest" class is 1.0, indicating that every "Not Ready for Harvest" prediction was correct. The precision for the "Ready for Harvest" class is 0.9231, meaning that 92.31% of the "Ready for Harvest" predictions were accurate. The recall for "Not Ready for Harvest" is 1.0, showing that the model detected all "Not Ready for Harvest" samples without error. The recall for "Ready for Harvest" is 0.9231, indicating that the model successfully detected 92.31% of the "Ready for Harvest" samples.

* Corresponding author

DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

The F1-Score, which is the mean of precision and recall, is 1.0 for the "Not Ready for Harvest" class and 0.9231 for the "Ready for Harvest" class, indicating an excellent balance between precision and recall. The support indicates the number of samples for each class, with 17 samples for "Not Ready for Harvest" and 13 samples for "Ready for Harvest." Overall, these results demonstrate that the model performs exceptionally well, with high accuracy, strong precision and recall, and balanced performance across both classes.

DISCUSSIONS

The performance of the proposed classification model is demonstrated through several key evaluation metrics and graphical representations. The Evaluation focus on the learning behavior of the model during training, its ability to generalize to new data, and its consistency across various subsets of data used in cross-validation. By examining the Learning Curve, Receiver Operating Characteristic (ROC) graph, and Cross-Validation Scores Distribution, we can gain a deeper understanding of how the model handles both the training data and the validation process, as well as its overall robustness and accuracy. Each of these graphs provides valuable insights into different aspects of the model's performance, which will help identify its strengths and potential areas for improvement.

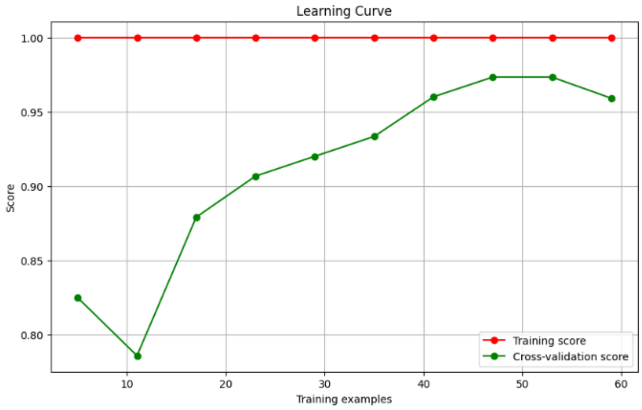

Fig.3 Learning Curve graph of the proposed classification model

The Learning Curve in Fig.3 shows that the model has excellent training performance, as indicated by the red line (Training score) consistently at 1.0, meaning the model perfectly learns the training data. Additionally, the validation score (Cross-validation score), represented by the green line, continues to improve as the amount of training data increases, indicating that the model effectively utilizes more data to improve its performance on the validation set. Although there are slight fluctuations in the validation score, the general trend remains stable at a high level (close to 1.0), demonstrating that the model has strong generalization capabilities and can work well outside the training data.

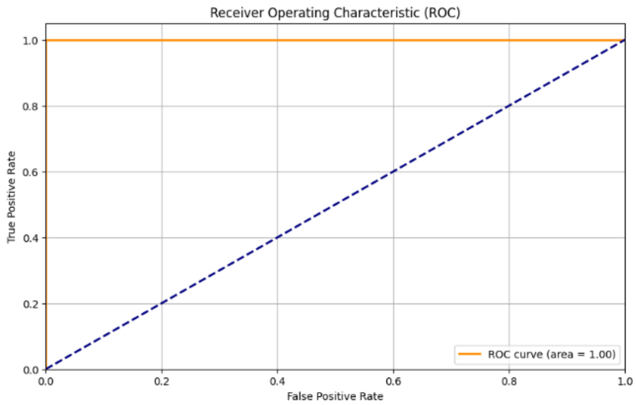

Fig.4 Receiver Operating Characteristic (ROC) graph of the proposed classification model

The Receiver Operating Characteristic (ROC) graph in Fig.4 shows that the model performs very well, with the ROC curve (orange line) reaching the top left corner of the graph, indicating a perfect True Positive Rate (TPR) versus False Positive Rate (FPR). The Area Under the Curve (AUC) of 1.0 indicates that the model has perfect classification ability, being able to separate positive and negative classes with 100% accuracy. This graph confirms that the model is very effective in distinguishing between the two classes, without any errors in prediction.

* Corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 4, October 2024
DOI : https://doi.org/10.33395/sinkron.v8i4.14126
e-ISSN : 2541-2019
p-ISSN : 2541-044X

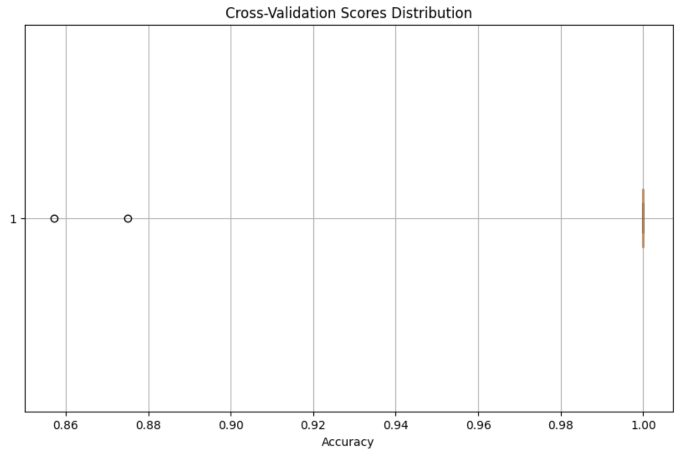

Fig.5 Cross Validation Scores Distribution graph of the proposed classification model

The Cross-Validation Scores Distribution graph in Fig.4 shows the distribution of accuracy scores of the models evaluated using the cross-validation technique. The plot shows that most of the model accuracy scores are at very high values close to 1.0, indicating that the model performs consistently well on the various subsets of data used in cross-validation. However, there is a slight spread of lower scores around 0.86 to 0.88, indicating a small variability in model performance on some subsets of data. Overall, this graph shows that the model tends to provide stable and highly accurate results, with only minor insignificant variations.

Table 5. Comparison of Accuracy and Image Processing Techniques in Oyster Mushroom Harvesting Systems

Method	Image Processing Technique	Classification Algorithm	Accuracy
MLP (proposed method)	Image preprocessing, segmentation, morphological operations, feature extraction	Multi-Layer Perceptron (MLP)	96.67%
C4.5	Feature extraction (Order 1: mean, skewness, variance, kurtosis, entropy)	C4.5 Decision Tree	84%
Canny Edge Detection & Contour analysis	Edge detection and contour analysis	Grade based classification	80%

In Table 5, we compare the three methods. The MLP-based approach achieves the highest accuracy due to its more comprehensive image processing and classification pipeline. The C4.5 method, while simpler, offers a balance between accuracy and computational efficiency, making it suitable for scenarios with limited resources (Pratomo 2021). The Canny edge detection and contour method, despite being effective in detecting mushroom features, has a lower accuracy (80%) as it primarily relies on basic edge detection techniques, which may not fully capture the complexities of the mushroom's visual characteristics needed for precise classification (Ayuninghemi et al. 2022). Each method has its strengths, but the MLP-based system stands out for its ability to generalize and classify mushroom maturity with greater precision.

CONCLUSION

This research successfully developed an oyster mushroom harvest determination system based on image processing and Multi-Layer Perceptron (MLP), demonstrating high accuracy in classifying mushroom maturity with a success rate of 96.67%. The system, tested on 150 images divided into "Ready to Harvest" and "Not Ready to Harvest" classes, utilized Grid Search with 5-fold cross-validation to identify optimal parameters. The evaluation showed strong performance, with only one misclassification and high precision, recall, and F1-scores for both classes. The Learning Curve and ROC graphs further indicated the model's ability to generalize well and accurately separate the two classes. While the study enhances the consistency and accuracy of harvest timing, it does not directly address increased yields, as this would require further testing in real agricultural settings. Future research should explore the system's impact on yields through field trials and its applicability in varying environmental conditions. Overall, this system offers a reliable tool to support efficiency in mushroom farming and contributes to the advancement of digital agriculture in Indonesia.

ACKNOWLEDGMENT

The researcher would like to thank the program of Bantuan Operasional Perguruan Tinggi Negeri (BOPTN) organized by the Directorate of Research, Technology, and Community Service for Fiscal Year 2024 for the

* Corresponding author

research grant awarded to us with contract number: 111/E5/PG.02.00.PL/2024. Thanks also to Mushroom Cultivation “Sahabat Jamur Parepare” for allowing us to take mushroom images.

REFERENCES

Amir, N., Yunus, A., Kadir, M.I., Syamsinar, S., & Hardiani, A. S. (2023). “Strategi Pengembangan Usaha Jamur Tiram Putih di Desa Simbang Kecamatan Simbang Kabupaten Maros (Studi Kasus Kelompok Wanita Tani Simbang Sejahtera).” *Jurnal Galung Tropika*, 12(2). https://doi.org/10.31850/jgt.v12i2.1095

Ayuninghemi, R., Sjamsijah, N., Yuana, D.B.M., Perdanasari, L., Hidayatullah, M., & Amal, I.I. (2022). “Image Segmentation for Oyster Mushroom Grade with Canny Detection for Image Classification.” *International Journal Of Artificial Intelegence Research* 6(1). DOI: 10.29099/ijair.v6i1.2.468

Chawla, N. V., Bowyer, K. W., Lawrence, O. H., & Kegelmeyer, W. P. (2002). “SMOTE: Synthetic Minority Over-Sampling Technique.” *Journal of Artificial Intelligence Research* 16:321–57. doi: 10.1613/jair.953.

Dhanya, P.M., & Ramkumar, P.B. (2023). “Text Analysis Using Morphological Operations on a Neutrosophic Text Hypergraph.” *Neutrosophic Sets and Systems*, Vol. 61. https://digitalrepository.unm.edu/nss_journal/vol61/iss1/19

Husain, N. P., & Aji, N.B. (2019). “Klasifikasi Sinyal EEG Dengan Power Spectra Density Berbasis Metode Welch Dan MLP Backpropagation.” *Jurnal ELTIKOM* 3(1):17–25. doi: 10.31961/eltikom.v3i1.99.

Kadir, M.I., & Husain, N.P. (2021). “Pelatihan Budidaya Jamur Dan Pembuatan Media Tanam Jamur Tiram Di Sulawesi Selatan.” *Patria Artha Journal of Community (PKM)* 1(2). doi: 10.33857/pajoco.v1i2.484.

Kang, D. H., Kim, N. K., Lee, W., & Kang, H, W. (2024). “Geometric Feature Extraction in Nanofiber Membrane Image Based on Convolution Neural Network for Surface Roughness Prediction.” *Heliyon* 10(15):e35358. doi: 10.1016/j.heliyon.2024.e35358.

Khoiru, N., Pradana, D.A., & Widaningrum, I. (2022). “PENERAPAN ALGORITMA PRINCIPAL COMPONENT ANALYSIS (PCA) DAN K-NEAREST NEIGHBORS (KNN) PADA KLASIFIKASI JENIS JAMUR BERACUN DAN TIDAK BERACUN.” *Jurnal Rekayasa Teknologi dan Komputasi*.

Lee, S., Lee, A., & Hong, M. (2023). “Cardiac CT Image Segmentation for Deep Learning–Based Coronary Calcium Detection Using K-Means Clustering and Grabcut Algorithm.” *Computer Systems Science and Engineering* 46(2):2543–54. doi: 10.32604/csse.2023.037055.

Pardede, D., Hayadi, B. H., & Iskandar. (2022). “KAJIAN LITERATUR MULTI LAYER PERCEPTRON: SEBERAPA BAIK PERFORMA ALGORITMA INI.” *Journal of ICT Aplication and System*, Volume 1 Nomor 1, hal 23-34, doi: https://doi.org/10.56313/.v1i1.127

Prasekti, Y. H., & Nugroho, B. B. (2021). “FAKTOR SOSIAL EKONOMI PETANI JAMUR TIRAM DI TULUNGAGUNG.” *Jurnal AGRIBIS* 7(2):1–6. doi: 10.36563/agribis.v7i2.391.

Pratama, D. A., Mutaqin, I. R., & Manuela, K. R. (2023). “Analisis Terjadinya Kanker Paru-Paru Pada Pasien Menggunakan Decision Tree: Penerapan Algoritma C4.5 Dan RapidMiner Untuk Menentukan Risiko Kanker Pada Gejala Pasien.” *Jurnal Teknik Mesin, Industri, Elektro Dan Informatika (JTMEI)* 2(4). https://doi.org/10.55606/jtmei.v2i4.3004

Pratomo, C. H. (2021). “KLASIFIKASI CITRA JAMUR MENGGUNAKAN ALGORITMA C4. 5.” Thesis, Sekolah Tinggi Manajemen Informatika dan Komputer (Akakom Yogyakarta).

Saputra, D. E., Rahmawati, D., & Ibadillah, A. F. (2019). “Pengolahan Citra Digital Dalam Penentuan Panen Jamur Tiram.” *Jurnal Teknik Elektro dan Komputer TRIAC* 6(1). doi: 10.21107/triac.v6i1.4356.

Sert, E., & Okumus, I. T. (2014). “Segmentation of Mushroom and Cap Width Measurement Using Modified K-Means Clustering Algorithm.” *Advances in Electrical and Electronic Engineering* 12(4):354–60. doi: 10.15598/aeee.v12i4.1200.

Shi, L., Wei, Z., You, H., Wang, J., Bai, Z., Yu, H., Ji, R., & Bi, C., (2024). “OMC-YOLO: A Lightweight Grading Detection Method for Oyster Mushrooms.” *Horticulturae* 10(7):742. doi: 10.3390/horticulturae10070742.

Sonang, S. S., Yuhandri, Y., & Setiawan, A. (2023). “Image Transformation With Lung Image Thresholding and Segmentation Method.” *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)* 7(2):278–85. doi: 10.29207/resti.v7i2.4321.

Umniyatie, S., & Pramiadi, D. (2013). “BUDIDAYA JAMUR TIRAM (PLEURETUS.sp) SEBAGAI ALTERNATIF USAHA BAGI MASYARAKAT KORBAN ERUPSI MERAPI DI DUSUN PANDAN, WUKIRSARI, CANGKRINGAN, SLEMAN DIY.” *Inotek*, Vol. 17, No. 2. Doi: 10.21831/ino.v17i2.3357

* Corresponding author