

Investigation of oil palm fruit bunch ripeness classification using machine learning classifiers

*Muhammad Arif Zulkhoiri*¹, *Hasimah Ali*^{1*}, *Ahmad Firdaus Ahmad Zaidi*¹, *Siti Nurul Aqmariah Mohd Kanafiah*¹, *Yessi Jusman*², *Mohamed Elshaikh*¹ and *Tuan Muhammad Taufiq Aiman Tuan Noor*¹

¹ Department of Mechatronic, Faculty of Electrical & Technology Engineering, Universiti Malaysia Perlis (UniMAP), Malaysia

² Department of Electrical, Faculty of Electrical, Universiti Muhammadiyah Yogyakarta, Indonesia

Abstract. The palm oil industry, particularly in Southeast Asia, relies heavily on accurate ripeness classification of oil palm fruit bunches to ensure high-quality oil production. Despite advances in palm oil classification, distinguishing between different ripeness levels remains challenging due to subjective human judgment and labor-intensive traditional methods. This study proposes an intelligent classifier using color-based features to classify oil palm fruit bunches into three categories: ripe, half-ripe, and unripe. This framework involved capturing images of oil palm fruit bunches at Felda Chuping 2 using commercial camera, followed by image pre-processing such as resizing and cropping. Color-based features by means HSV-, RGB- and YCbCr-based features were extracted and used as significant features. The mean and standard deviation of colour-based features were then subjected to k-Nearest Neighbour (kNN) and Support Vector Machine (SVM) classifier utilizing two different strategies of hold-out and 10-fold cross-validation. Based on the results obtain, the YCbCr based features using kNN classifier achieved 97.40% (hold-out) and YCbCr based features using SVM classifier gives the highest recognition which is 100% (10-fold). The results shows that the use of colour space features able in distinguishing the ripeness levels of oil palm fruit bunches, thus considered as promising approach to be implemented in real-time application.

1 Introduction

The palm oil industry plays a significant role in the global agricultural sector, particularly in Southeast Asia where countries like Indonesia and Malaysia are major producers [1]. Palm oil is a versatile vegetable oil used in various industries, including food, oleochemicals, and biofuels. Ensuring the quality of palm oil presents challenges due to factors such as ripeness classification and traditional manual grading techniques. Oil palm fruit bunches are typically graded through visual inspection by trained workers, who assess them based on color, texture, and the number of loose fruits on the ground. However, this approach is labor-intensive and time-consuming. Moreover, workers may misjudge the ripeness due to factors such as the

* Corresponding author: hasimahali@unimap.edu.my

height of the tree, miscounting the loose fruits, limited visibility of the bunches, and varying lighting conditions. The demand for palm oil continues to rise due to its cost-effectiveness and versatility in different applications [2]. In recent years, there has been a growing emphasis on sustainable palm oil production and the adoption of Industry Revolution 4.0 (IR 4.0) technologies to enhance productivity and reduce costs. These technologies include automation, artificial intelligence, and computer vision systems [3]. The quality of palm oil is influenced by various factors, including the ripeness of the harvested fruit [2]. Achieving the appropriate level of ripeness during harvesting is crucial in determining the content of free fatty acids (FFA) in the oil, which affects its quality. Manual grading by human graders, the traditional method used to classify ripeness, often yields inconsistent results. Human errors and subjective judgments can lead to variations in grading, impacting the overall quality of the palm oil [4].

This project proposes the use of intelligent classifiers based on colour features to classify three classes of oil palm fruit bunch ripeness: ripe, half-ripe, and unripe. The objective of this research is to investigate the oil palm fruit bunch ripeness classification using colour-based features by utilizing intelligent classifiers, to study the characteristics of colour-based features of the oil palm fruit bunch ripeness based on HSV, YCbCr, and RGB, and to evaluate the performance of the oil palm fruit bunch classification using K-NN and SVM classifiers. Data collection for the three categories will be conducted at an oil palm plantation at Felda Chuping 2, Perlis, Malaysia. Preprocessing of fruit bunch oil palm images involves cropping the image to get the region of interest and resizing the image to the same size. Colour-based features, such as HSV, RGB, and YCbCr, will be extracted to categorize the three classes of ripeness. The performance evaluation will use machine learning classifiers K-NN and SVM.

2 Materials and methods

The methodology involves several steps from image acquisition to classification using machine learning models. Figure 1 shows the framework of the proposed method. It consists of image acquisition, image pre-processing, feature extraction and classification.

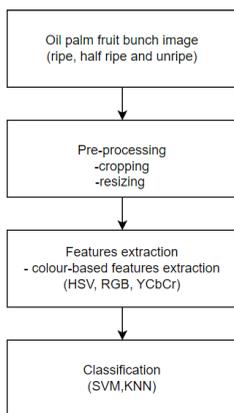


Fig. 1. The framework of the proposed method

2.1 Image Acquisition

The fruit bunch oil palm images were acquired using a commercialized camera; Infinix GT 10 Pro with a 108 MP at Felda Chuping 2, Perlis. A total of 257 were collected, categorized

into ripe (100 images), half-ripe (101), and unripe (57). Figure 2 shows the example of oil palm fruit bunch images.



Fig. 2. The example oil palm fruit bunch images; (a) unripe, (b) ripe and (c) half ripe fruit images

2.2 Image Pre-processing

The main goals of image pre-processing is to enhance the quality of the image and to prepare them for the subsequent analysis such as features extraction and classification. In this research, each of the acquired images was subjected to the cropping procedure in order to isolate only the fruit bunch region of interest (ROI) [6]. Then, the ROI images were resized for several reasons. Firstly, resizing improves processing speed by reducing the dimensions of the images, which decreases the amount of data that needs to be handled. This not only speeds up computational tasks but also makes the process more efficient. Standardizing the size of images is considered crucial for machine learning models, which often require inputs to be uniform. This consistency helps in achieving better results. Resizing images also reduces computational costs, thus making it easier to manage large datasets without straining system resources. In this work, the original fruit bunch oil palm image of size 9000×12000 pixels is resized into 300×400 pixels.

2.3 Feature Extraction: Color-based Features

In this stage, the resized image was then subjected to the feature extraction techniques. Colour-based features have been utilized and extracted to use as features. These includes RGB-based features [8], HSV-based features and YCbCr-based features.

HSV is useful in image processing tasks like object detection and segmentation because it separates colour information (hue) from intensity information (value) [7]. This makes it easier to work with specific colours irrespective of lighting conditions. HSV very useful for colour extraction. In this work, the original images of oil palm are in the RGB form. Thus, to get the HVS value, firstly the image is converted from RGB to HSV. Then, the extracted HSV features were used as features to classify the ripe, half ripe and unripe. The value of HSV colour space is calculated based on the mean and standard deviation of each channel: *hue (H)*, *saturation (S)*, and *value (V)*. The equation of the mean and standard deviation (Std) are as follows:

Mean formula for HSV:

$$Mean_H = \frac{1}{N} \sum_{i=1}^N H_i \tag{1}$$

$$Mean_S = \frac{1}{N} \sum_{i=1}^N S_i \tag{2}$$

$$Mean_V = \frac{1}{N} \sum_{i=1}^N V_i \tag{3}$$

Standard deviation formula for HSV:

$$Std_H = \sqrt{\frac{1}{N} \sum_{i=1}^N (H_i - Mean_H)^2} \tag{4}$$

$$Std_S = \sqrt{\frac{1}{N} \sum_{i=1}^N (H_i - Mean_S)^2} \quad (5)$$

$$Std_V = \sqrt{\frac{1}{N} \sum_{i=1}^N (H_i - Mean_V)^2} \quad (6)$$

On the other hand, *YCbCr* is a colour space used extensively in image compression and image formats [9]. It separates image luminance (brightness) information from chrominance (colour) information, which is advantageous for compression and processing. By separating the luminance (*Y*) from the chrominance *Cb* and *Cr* components, *YCbCr* helps in isolating the colour information from lighting conditions. This separation can make the classification of ripeness more resilient to changes in lighting. *YCbCr* makes it easier to segment colour regions in an image. For instance, ripe and unripe palm fruits have different colours that can be more effectively distinguished using the *Cb* and *Cr* components. The original image firstly is converted from *RGB* to *YCbCr*. The extracted features from the *YCbCr* colour space are the mean and standard deviation of each channel: *Y* (luminance), *Cb* (blue chrominance), and *Cr* (red chrominance). The equation of both mean and standard deviation of *Y*, *Cb* and *Cr* are as followed:

Mean formula for *YCbCr*:

$$Mean_Y = \frac{1}{N} \sum_{i=1}^N Y_i \quad (7)$$

$$Mean_{Cb} = \frac{1}{N} \sum_{i=1}^N Cb_i \quad (8)$$

$$Mean_{Cr} = \frac{1}{N} \sum_{i=1}^N Cr_i \quad (9)$$

Standard deviation formula for *YCbCr*:

$$Std_H = \sqrt{\frac{1}{N} \sum_{i=1}^N (H_i - Mean_H)^2} \quad (10)$$

$$Std_H = \sqrt{\frac{1}{N} \sum_{i=1}^N (H_i - Mean_H)^2} \quad (11)$$

$$Std_H = \sqrt{\frac{1}{N} \sum_{i=1}^N (H_i - Mean_H)^2} \quad (12)$$

2.4 Classification

In this experiment, we employed *k*-Nearest Neighbours (*k*-NN) and Support Vector Machines (SVM) as machine learning classifiers to classify the three-class ripeness of fruit bunch oil palm. The *k*-NN is a straightforward and versatile machine learning algorithm generally used for classification and regression tasks [10].

The *k*-NN classifier stores all available data points and makes predictions based on their similarity to new or unseen data. This approach makes *k*-NN is non-parametric method because it does not assume anything about the underlying data distribution. In the training phase, *k*-NN simply memorizes the entire dataset, storing the input features and their corresponding labels. When tasked with making a prediction for a new data point, *k*-NN calculates the distance between this point and every point in the training set. The most used distance metric is the Euclidean distance, although other metrics like Manhattan distance can also be applied. In this study, the Euclidean distance has been employed as distance metric.

Meanwhile, SVMs are highly effective supervised learning models which used extensively for classification and regression tasks in machine learning [11]. They are

particularly valuable when data not linearly separate into classes using linear boundaries in its original form. Instead of attempting to force a linear fit, SVMs employ a technique called the kernel trick, which transforms data into a higher-dimensional space. This transformation allows SVM to discover a hyperplane that maximizes the margin of the distance between the hyperplane and the closest data points of each class. By maximizing this margin, SVM improve their ability to generalize well to new, unseen data while reducing the risk of overfitting. A key feature of SVMs is their reliance on support vectors data points closest to the decision boundary. These support vectors are instrumental in defining the hyperplane and are the only points used during training, making SVMs efficient with memory, particularly beneficial for handling large datasets. SVMs support various kernel functions such as linear, polynomial, and radial basis function (RBF), enabling them to handle complex, non-linear relationships in data effectively. This adaptability makes SVMs suitable for a wide range of applications including text and image classification, bioinformatics, and financial analysis.

3 Results and discussion

3.1 Evaluation of fruit bunch oil palm image pre-processing

Figure 3 shows the example of the result for unripe class cropped fruit bunch oil palm image. The image was cropped from original size into the region of interest (ROI), so that irrelevant background has been removed. Only ROI was further used for resizing image before the feature extraction process. Figure 4 shows the result of resized image. The original images is a size of 9000×12000 pixels is cropped into the size of 300×400 pixels., thus it not reduce the computational cost but also have uniformity in terms of sizing.

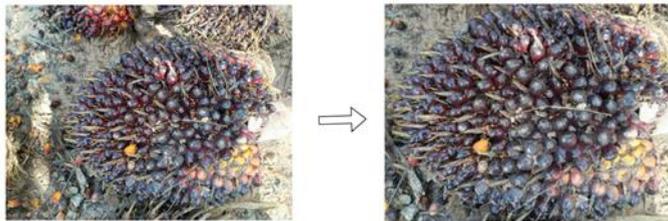


Fig. 3. Result of cropped oil palm fruit bunch images for unripe class



Fig. 4. Result of cropped and resized of oil palm fruit bunch image for ripe class

3.2 Experimental Results

In this study, a total of 257 images of oil palm fruit bunch have been employed. Three level of ripeness class of the fruit bunch images have been investigated namely, ripe, half ripe and unripe class. In this work, colour features have been extracted using HSV, YCbCr and RGB. To evaluate the effectiveness of the proposed method, *holdout* and *k*-fold cross validation

strategies have been used. For hold-out strategy, it divided into two set of data, training and testing data. The data is split into two parts, 70% used for training and 30% for testing. The model is trained on the training set and the trained model is then evaluated on the testing set. Meanwhile, the k -fold cross validation strategy applies the concept at which the dataset is divided into value of the k equally subset or fold. The model then trains and evaluated k times, each time using a difference fold as testing set and the remaining fold as the training set. Then the extracted features will further be subjected to k-NN and SVM classifier to classify three classes, which are ripe, half-ripe and unripe.

3.2.1 Recognition using HSV-based features

In this work extracted HSV-based features is subjected to k -NN and SVM classifier using two strategies, namely *holdout* and k -fold cross validation strategy.

1) Performance of HSV-based features using k -NN classifier: Hold-out cross validation

Table 1 shows the result of confusion matrix of HSV features using k-NN classifier with hold out strategy (70% training, 30% testing). Based on the Table 1, it can be seen that the average recognition rate of HSV features using k -NN classifier achieved about 93.4%. Among the class, unripe gives lowest recognition rate which is 68.8%. Five (5) out of 16 of images are misclassified as ripe. This may be due to similarity of features that represent to the ripe features. Meanwhile, ripe and half ripe contribute perfect recognition rate which is 100%.

Table 1. Confusion matrix of HSV + k-NN classifier (*hold-out*)

| | Ripe | Half ripe | Unripe |
|-----------|-------------|------------------|---------------|
| Ripe | 30 | 0 | 0 |
| Half ripe | 0 | 30 | 0 |
| Unripe | 5 | 0 | 11 |

2) Performance of HSV-based features using k-NN classifier: 10-fold cross validation

Figure 5 shows the recognition rates for 10-fold cross validation of HSV features using kNN classifier. Based on the Figure 5, the average recognition rate of the HSV using 10-fold is 96.12%.

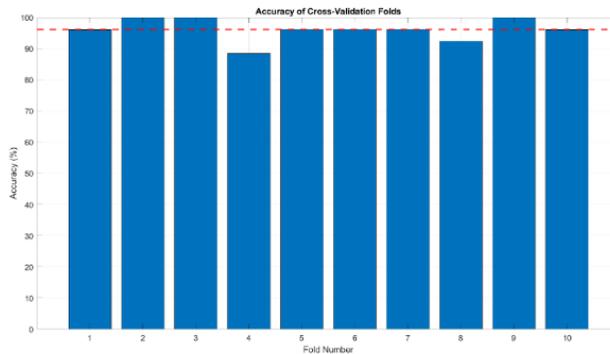


Fig. 5. HSV-based features + kNN classifier using 10-fold cross validation

3) Performance of HSV-based features using SVM classifier: Hold-out cross validation

Table 2 shows the result of confusion matrix of HSV features using SVM classifier with hold out strategy (70% training, 30% testing). Based on the Table 2, it can be seen that the average recognition rate of HSV features using SVM classifier achieved about 96.10%. Among the class, half ripe gives lowest recognition rate which is 93.54%. 2 out of 31 of images are

misclassified as ripe. This may be due to similarity of features that represent to the ripe features. Meanwhile, ripe contribute perfect recognition rate which is 100%.

Table 2. Confusion matrix of HSV + SVM classifier (*hold-out*)

| | Ripe | Half ripe | Unripe |
|-----------|------|-----------|--------|
| Ripe | 30 | 0 | 0 |
| Half ripe | 2 | 29 | 0 |
| Unripe | 1 | 0 | 15 |

4) Performance of HSV-based features using SVM classifier: 10-fold cross validation

Figure 6 shows the recognition rates for 10-fold cross validation of HSV features using KNN classifier. Based on the Figure 6, the average recognition rate of the HSV using 10-fold is 99.23%.

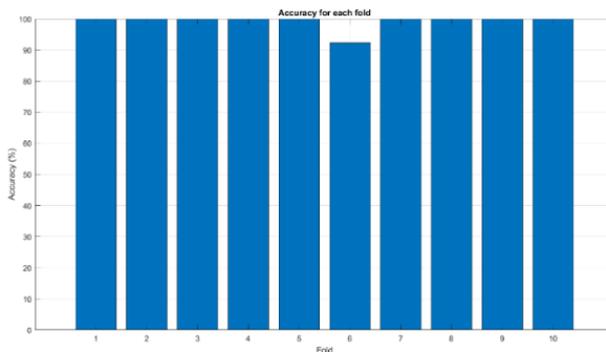


Fig. 6. HSV-based features + SVM classifier using 10-fold cross validation

3.2.2 Recognition using RGB-based features

In this section, the RGB features is conducted to evaluate the performance using kNN and SVM classifier using aforementioned strategies, which are holdout and 10-fold cross validation.

1) Performance of RGB-based features using k-NN classifier: Hold-out cross validation

Table 3 shows the result of confusion matrix of RGB features using KNN classifier with hold out strategy (70% training, 30% testing). Based on the Table 3, it can be seen that the average recognition rate of RGB features using KNN classifier achieved about 92.21%. Among the class, unripe gives lowest recognition rate which is 62.5%. 4 out of 16 of images are misclassified as ripe and 2 out of 16 misclassified as half ripe. This may be due to similarity of features that represent to the ripe features. Meanwhile, ripe contribute perfect recognition rate which is 100%.

Table 3. Confusion matrix of RGB + kNN classifier (*hold-out*)

| | Ripe | Half ripe | Unripe |
|-----------|------|-----------|--------|
| Ripe | 30 | 0 | 0 |
| Half ripe | 0 | 30 | 0 |
| Unripe | 4 | 2 | 10 |

2) Performance of RGB -based features using k-NN classifier: 10-fold cross validation

Figure 7 shows the recognition rates for 10-fold cross validation of RGB features using KNN classifier. Based on the Figure 7, the average recognition rate of the RGB using 10-fold is 92.20%.

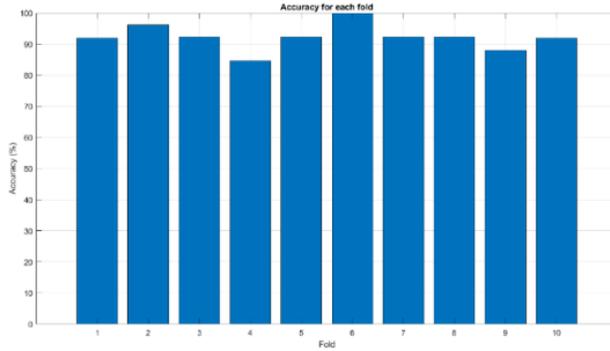


Fig. 7. RGB-based features + kNN classifier using 10-fold cross validation

3) Performance of RGB -based features using SVM classifier: Hold-out cross validation

Table 4 shows the result of confusion matrix of RGB features using SVM classifier with hold out strategy (70% training, 30% testing). Based on the Table 4, it can be seen that the average recognition rate of RGB features using SVM classifier achieved about 100%.

Table 4. Confusion matrix of RGB + kNN classifier (10-fold)

| | Ripe | Half ripe | Unripe |
|-----------|------|-----------|--------|
| Ripe | 30 | 0 | 0 |
| Half ripe | 0 | 31 | 0 |
| Unripe | 0 | 0 | 16 |

4) Performance of RGB -based features using SVM classifier: 10-fold cross validation

Figure 8 shows the recognition rates for 10-fold cross validation of RGB features using SVM classifier. Based on the Figure 8, the average recognition rate of the RGB features using 10-fold achieved 100%.

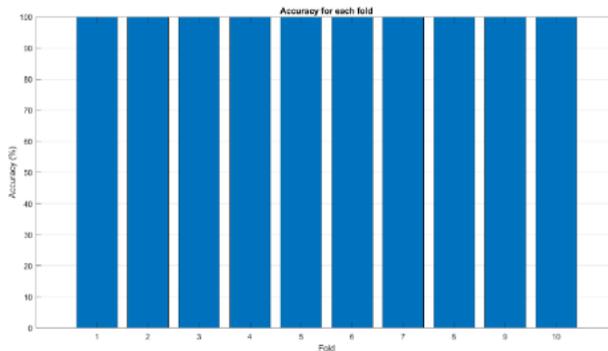


Fig. 8. RGB-based features + SVM classifier using 10-fold cross validation

3.2.3 Recognition using YCbCr-based features

Lastly, the YCbCr features is subjected to investigate the performance using kNN and SVM classifiers utilizing similar strategies: holdout and 10-fold cross validation.

5) Performance of YCbCr -based features using k-NN classifier: Hold-out cross validation

Table 5 shows the result of confusion matrix of YCbCr features using KNN classifier with hold out strategy (70% training, 30% testing). Based on the Table 5, it can be seen that the average recognition rate of YCbCr features using SVM classifier achieved about 97.40%. Among the class, unripe gives lowest recognition rate which is 87.5%. 2 out of 16 of images are misclassified as ripe. This may be due to similarity of features that represent to the ripe features. Meanwhile, ripe and half ripe contribute perfect recognition rate which is 100%.

Table 5. Confusion matrix of YCbCr + kNN classifier (*hold-out*)

| | Ripe | Half ripe | Unripe |
|-----------|------|-----------|--------|
| Ripe | 31 | 0 | 0 |
| Half ripe | 0 | 30 | 0 |
| Unripe | 2 | 0 | 14 |

6) Performance of YCbCr -based features using k-NN classifier: 10-fold cross validation

Figure 9 shows the recognition rates for 10-fold cross validation of YCbCr features using KNN classifier. Based on the Figure 9, the average recognition rate of the YCbCr using 10-fold is 96.12%.

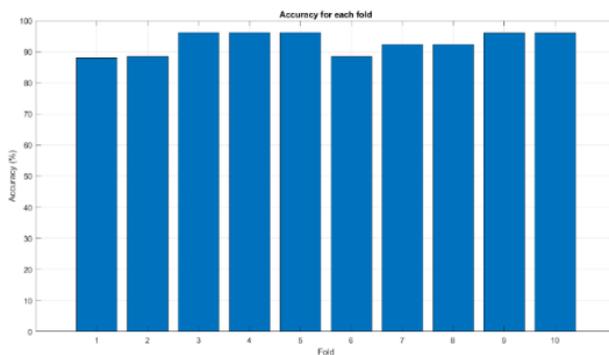


Fig. 9. YCbCr -based features + kNN classifier using 10-fold cross validation

7) Performance of YCbCr -based features using SVM classifier: Hold-out cross validation

Table 6 shows the result of confusion matrix of YCbCr features using SVM classifier with hold out strategy (70% training, 30% testing). Based on the Table 4.6, it can be seen that the average recognition rate of YCbCr features using SVM classifier achieved about 100%.

Table 5. Confusion matrix of YCbCr + SVM classifier (*hold-out*)

| | Ripe | Half ripe | Unripe |
|-----------|------|-----------|--------|
| Ripe | 30 | 0 | 0 |
| Half ripe | 0 | 31 | 0 |
| Unripe | 0 | 0 | 16 |

8) Performance of YCbCr -based features using SVM classifier: 10-fold cross validation

Figure 10 shows the recognition rates for 10-fold cross validation of YCbCr features using SVM classifier. Based on the Figure 10, the average recognition rate of the YCbCr using 10-fold is 100%.

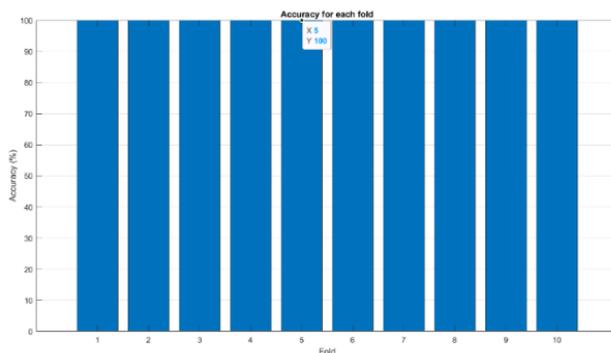


Fig. 10. YCbCr -based features + SVM classifier using 10-fold cross validation

The study found that using colour-based features greatly improves the accuracy of classifying palm oil fruit ripeness. HSV-based features performed particularly well, with the kNN classifier achieving an average recognition rate of 93.4% using hold-out cross-validation and consistent results with k-fold cross-validation. Meanwhile, the SVM classifier has demonstrated promising performance with HSV-based features, reaching a recognition rate of 93.4% with hold out and has achieved about 99% with k-fold cross-validation. These findings suggest that the HSV colour space is highly effective for distinguishing the ripeness levels of oil palm fruit bunches. Likewise, YCbCr-based features yielded excellent results, with the kNN classifier achieving an average recognition rate of 96.12% using k-fold and 97.40% with hold out validation. The SVM classifier achieved a perfect recognition rate of 100% with YCbCr-based features. While RGB-based features also provided good classification accuracy, with the KNN classifier achieving 92.2%, the SVM classifier using both hold-out and k-fold cross-validation achieved perfect accuracy.

4 Conclusion and recommendation

This paper has presented the classification of oil palm fruit bunch ripeness using machine learning classifier; kNN and SVM classifiers. Based on the results obtain, the YCbCr based features using kNN classifier achieved 97.40% (hold-out) and YCbCr based features using SVM classifier gives the highest recognition which is 100% (10-fold). The results shows that the used of colour space features able in distinguishing the ripeness levels of oil palm fruit bunches, thus considered as promising approach to be implemented in real-time application. However, further study should be conducted in large scale data for enhancing the robustness of the proposed system. In practical applications, this system could be incorporated into existing automated sorting machines in palm oil plantations. Cameras placed in key positions would capture images of fruit bunches in real time, allowing the proposed model to analyze and classify their ripeness levels. However, challenges like fluctuating lighting, image noise, and hardware constraints must be resolved to ensure the system reliability in outdoor settings. Furthermore, utilizing specialized hardware, such as edge AI devices, could enhance the system performance for real-time classification. Further improvements to this system could include the use of more advanced deep learning models, such as Convolutional Neural

Networks (CNNs). Incorporating texture-based features, in addition to color-based features, may also enhance classification accuracy. Moreover, expanding the dataset and exploring techniques for handling large-scale data would increase the system robustness and scalability. The integration of this classification model into drones or mobile devices for autonomous plantation monitoring represents a promising direction for future research.

Acknowledgment. The authors would like to acknowledge the support from University Muhammadiyah Yogyakarta (UMY), Indonesia for the Research Activities Fund.

References

1. K. J. Goh, C. K. Wong, and P. H. C. Ng, Oil Palm in Encyclopaedia of Applied Plant Sciences, **3**, (2016).
2. H. Herman, T. W. Cenggoro, A. Susanto, and B. Pardamean, Deep learning for oil palm fruit ripeness classification with densenet, in Proceedings of 2021 International Conference on Information Management and Technology, ICIMTech 2021, (2021).
3. S. A. Ghazali, H. Selamat, Z. Omar, and R. Yusof, Image Analysis Techniques for Ripeness Detection of Palm Oil Fresh Fruit Bunches, ELEKTRIKA- Journal of Electrical Engineering, **18**, no. 3, (2019).
4. Suharjito, M. Asrol, D. N. Utama, F. A. Junior, and Marimin, Real-Time Oil Palm Fruit Grading System Using Smartphone and Modified YOLOv4, IEEE Access, **11**, (2023)
5. S. N. A. B. M. Robi, M. A. B. M. Izhar, M. Bin Sahrim, and N. B. Ahmad, Image Detection and Classification of Oil Palm Fruit Bunches, in 4th International Conference on Smart Sensors and Application: Digitalization for Societal Well-Being, ICSSA 2022, (2022).
6. A. Septiarini, A. Sunyoto, H. Hamdani, A. A. Kasim, F. Utamingrum, and H. R. Hatta, Machine vision for the maturity classification of oil palm fresh fruit bunches based on color and texture features, Sci Hortic, **286**, (2021).
7. I. Kurniastuti, E. N. I. Yuliati, F. Yudianto, and T. D. Wulan, Determination of Hue Saturation Value (HSV) color feature in kidney histology image, in Journal of Physics: Conference Series, (2022).
8. S. Tu, Y. Xue, C. Zheng, Y. Qi, H. Wan, and L. Mao, Detection of passion fruits and maturity classification using Red-Green-Blue Depth images, Biosyst Eng, **175**, (2018).
9. S. Gopinathan and S. Gayathri, A Study on Image Enhancement Techniques using YCbCr Color Space Methods, International Journal of Advanced Engineering Research and Science, **3**, no. 8, (2016).
10. S. Uddin, I. Haque, H. Lu, Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. Sci Rep **12**, 6256 (2022).
11. R. Gandhi, Support Vector Machine — Introduction to Machine Learning Algorithms, Towards Data Science. Jan. 17, (2024)