Cedarwood Quality Classification using SVM Classifier and Convolutional Neural Network (CNN)

Muhammad Ary Murti[1], Casi Setianingsih[2], Eka Kusumawardhani[3], Renal Farhan[4]
School of Electrical Engineering, Telkom University, Bandung, Indonesia[1, 2, 4]
Department of Electrical Engineering, Universitas Tanjungpura, Pontianak, Indonesia[3]

***Abstract*—Cedarwood is one of the most sought-after materials since it can be used to create a wide variety of household appliances. Other than its unique aroma, the product's quality is the most important selling attribute. Fiber patterns allow for a qualitative categorization of this wood. Traditionally, workers in the wood-processing business have relied solely on their eyesight to sort materials into several categories. As a result, there will be discrepancies in precision and efficiency, which will hurt the reputation of the regional wood sector. The answer to this issue is machine learning. In this study, we compare the performance of two different cedarwood quality classification systems where both systems use different machine learning methods namely Support Vector Machine (SVM) and Convolutional Neural Network (CNN). Each system will be sent images captured with a Logitech Brio 4K equipped with a joystick and ultrasonic sensors, labeled as belonging to one of five cedar classes (A, B, C, D, or E). In the initial method to learn the wood's pattern and texture, the Histogram of Oriented Gradient (HOG) will be used to identify the material. Meanwhile, the classification method uses a Support Vector Machine (SVM) which will be compared to find the best accuracy and time computation. The first system's experiment achieves 90 percent accuracy with a computation time of 1.40 seconds. For the second, we use a Convolutional Neural Network, a deep learning technique, to classify cedarwood (CNN). Extraction of features occurs in the convolution, activation, and pooling layers. Experimental results demonstrated a considerable enhancement, with an accuracy of 97% and a prediction speed of 0.56 seconds.**

***Keywords*—Cedarwood classification; convolutional neural network (CNN); HoG feature; SVM classification**

I. Introduction

When it comes to exterior, interior, and home appliance needs, cedarwood is one of the most popularly purchased light wood processing materials. This is not without good reason, the resin content contained in Cedarwood will emit a very fragrant odor, the goal of which is to expel the terms that will eat the wood composition [1]. Cedarwood also has a very high level of resistance to mold because it belongs to the group of hardwoods which incidentally has dense pores so that moist water does not easily enter and cause mold. That is why the quality of cedar is the main point of concern [2]. Like wood in general, cedarwood also has a variety of qualities depending on the factors of sawing and the age of the tree, so these quality factors can be classified directly by paying attention to the color, texture, and pattern of the fiber [3], [4].

Many businesses that deal with processed wood still perform classification by hand, using just their eyes and gut reactions to determine how many items are alike. Compliance with observations is only indicated as a percentage (50-60%) in that matter [5]. Therefore, it is challenging to avoid the issue of ambiguity and misclassification within each category. The legitimacy of Indonesia's processed wood sector is threatened by subjective standards of accuracy and production time [6]. In light of this, the field of machine learning offers a means of addressing and ultimately resolving these issues. Local wood industries can use machine learning to create uniform wood classes, allowing for reliable automatic classification [7].

The implementation of machine learning to create uniform wood classes, it can use two deep learning approaches into the cedarwood quality categorization system. Support Vector Machine (SVM) and Convolutional Neural Network (CNN) will be utilized to automatically capture and predict when wood enters the categorization area. These approaches will be directly integrated with the small conveyor, Controller, and Logitech Brio 4K webcam. Cedarwood quality data from Classes A through E is employed, and an automated feature extraction process is carried out with a total of 16 layers in CNN and in SVM. The Histogram of Oriented Gradient is then used to determine the grain pattern and texture of the wood during recognition.

II. Related Work

Widespread use of machine learning theory emerged during the era of the 4.0 Industrial Revolution, which also paid close attention to technical advancements. As a result, its implementation in the wood industry is quite probable [8] [9]. Over the years, many types of wood classification systems have been developed, such as wood classification based on PCA, 2DPCA, (2D) PCA and LDA [10], the second method is the Classification of Coconut Wood Quality based on Gray Level Co-Occurrence matric [11], and the third method is the Species Identification of Wooden Material using Convolutional Neural Network [12]. However, in the first method, the highest implementation accuracy was only 86.67% and even then, it had to be tested on a different sample, besides the feature extraction and identification process requires a good dataset structure to obtain a good system performance. In the second method Based on Gray Level Co-Occurrence Matrices, the highest accuracy obtained is only 78.82% so that it still requires a feature selection strategy to improve its performance. Then in the third method which classifies fir, pine, and fir wood with various CNN architectures used, the highest accuracy is 99.4%. However, this classification method can only identify the type of wood in certain parts and cannot make the identification of the whole wood being tested if it is done in real-time, so pre-processing is still needed to perfect the research.

III. Research Methodology

In this research, the CNN and SVM-based cedarwood classification system is broken down into three distinct phases: Acquisition, training, and categorization of images, as well as indication systems. In the first phase, an ultrasonic sensor is used to identify the presence of wood on the conveyor path, and the resulting input signal is used for auto-capture. The Controller relays the signal to the server using serial communication. When the server receives the command, the Logitech brio 4k webcam is activated to take a picture for use in the training and classification process. The second stage is responsible for machine learning and categorizing the collected images through the use of the convolution, activation, and max-pooling layers.

An indicator system made up of five LED lights of varying colors, each of which represents a different category of wood, shows the results of the second stage of learning. The following are the three working states of the cedarwood quality classification system illustrated in Fig. 1.

All process of sending commands between the device controller and the server is done using the serial communication method with a data rate of 8 bits per second. The detection and prediction code that the system has processed must first be converted into binary forms so that the data transmission process can be easily carried out in series.

A. Image Acquisition Process

Image acquisition is one of the subsystems directly related to the process of automatically capturing images when scanning wood. The sensor is placed horizontally next to the direction of the conveyor belt rate to detect wood when it passes by. When wood is detected, the capturing image will be performed by a webcam mounted in the sky of the acquisition box. This section functions as a data gateway between hardware and software on the server.

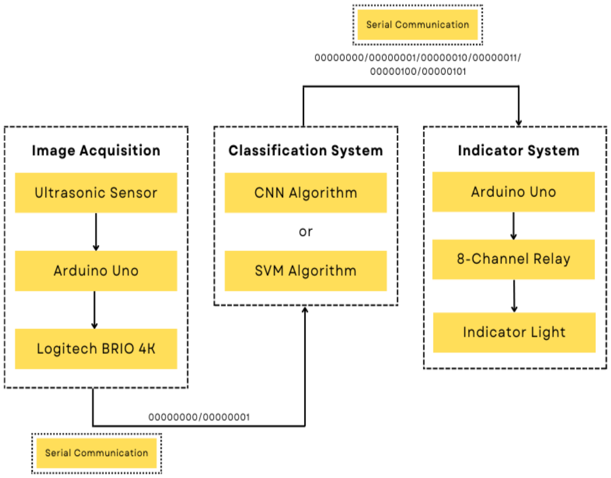


Fig. 1. Block Diagram of Classification System.

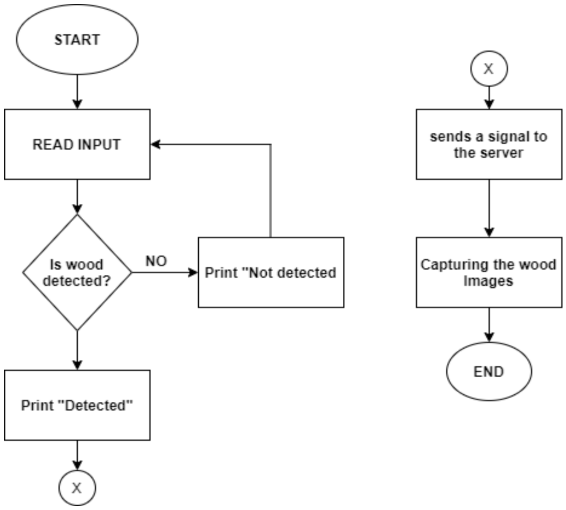
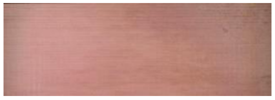
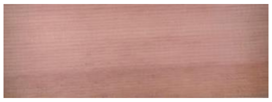
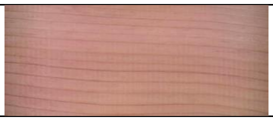
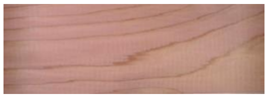
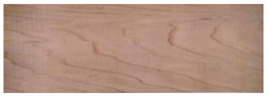


Fig. 2. Flowchart of Images Acquisition Process.

Fig. 2 is a flowchart diagram of the image acquisition process used to give the camera commands to carry out image capture tasks that are integrated between the Controller and the server with serial communication mode. Upon receiving a signal from the HC-SR04 proximity sensor that is incorporated with the Controller, the server will instruct the Logitech brio 4K to automatically capture the cedarwood runs on the conveyor track. The image is captured in a black, matte, 27.6 x 23.6 x 30 cm acquisition box lit from above by LED strips emitting 723 lux. The resulting image has a 640x480 pixel resolution and the following features of the wood used.

Images of cedarwood captured at a zoomed-in percentage of 130%, with a black backdrop, brightness of 26, Contrast of 24, and camera focus of 14 are displayed in Table I.

TABLE I. Characteristics of Cedarwood in Each Class

Class	Fig	Description
A		Magnificently straight fiber, with a distance so short that it is barely perceptible, is the hallmark of this technology.
B		There are a lot of fibers, they're all straight and rather thick, and you can see the separation between them.
C		The space between the fibers is considerable, and there are also some transverse fibers visible.
D		Fiber-to-fiber spacing and transverse pattern are easily visible.
E		Extremely transparent fiber; random fiber patterns are typically not straight.

B. Training and Classification Process

In this section, the image has been baptized and stored on the server, then pre-processing is done to match the classification architecture used. After a match between the image's dimensions and architecture, the image classification process is carried out using the CNN and SVM methods.

1) *Convolutional Neural Network (CNN) Method*

a) Feature Extraction: Fig. 3 explains the data processing up to sending commands to the microcontroller. The left is a classification process of wood images captured and stored in a server. On the right is a training process for the Convolutional Neural Network method to get the best classification model used in the system's classification. The first layer of the convolutional neural network design will be fed with the image acquired during the acquisition phase. But first, it needs to go through some pre-processing to ensure that the image and layer have compatible dimensions [13]. The CNN architecture VGG-16 technique is used in the wood classification system. VGG16 is used because it's considered a type of CNN, one of the most advanced computer vision models. VGG16 is an object detection and classification algorithm that can classify 1000 images of 1000 different categories with 92.7% accuracy. It is one of the most widely used algorithms for image classification and is easy to use with transfer learning. It uses a square picture with dimensions of 224 by 224 by 3 pixels as the suitability parameter. Up until the point of delivering commands to the microcontroller, the data processing is described in detail. The left is a classification process of wood images captured and stored in a server. On the right is a training process for the Convolutional Neural Network method to get the best classification model that will be used in the classification of the system. The acquired image will be used as input for the CNN architecture's first layer. However, prior to doing that, the image must be pre-processed so that its dimensions match those of the layer.

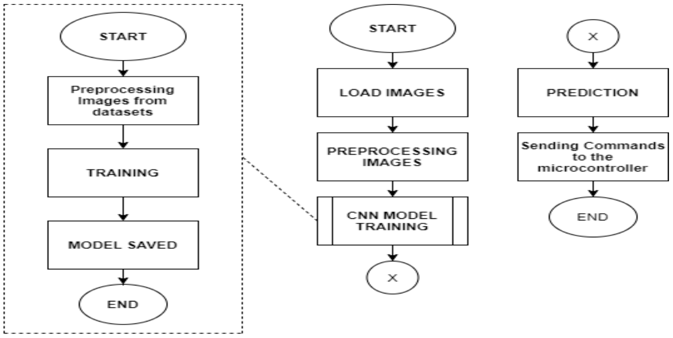


Fig. 3. Flowchart of Training and Classification Process using CNN Method.

In the CNN architecture VGG-16 algorithm [14] employed in the wood classification system, the parameter of appropriateness is a square-shaped image with dimensions of 224 x 224 x 3 pixels.

$$Rz = \frac{P(q,r)+P(q+1,r)+p(q,r+1)+p(q+1,r+1)}{4} \quad (1)$$

Based on Eq. 1, the image will be resized to be smaller, as in Fig. 4. However, so that the original rectangular image can be resized into a new rectangular image, it is necessary to do an additional process by improving the image's aspect ratio or adding padding to the new image to be resized. The step for resizing the image will affect the accuracy of the classification [15]. Also Adding padding can be done in three ways [16]:

- Inserting a zero value at the edge of the dimension.
- A max value at the edge of the dimension.
- Inserting the same value as the image at the edge of the new dimension.

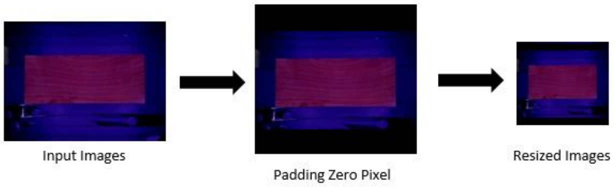


Fig. 4. Resizing Rectangle into Square Images.

A feature extraction technique is carried out to capture features in each wood class after the input image is found to be a match for the first layer of a convolutional neural network. The 2D convolution operation comes at the beginning of the feature extraction procedure. The convolution layer is the initial feature-extracting layer in the CNN architecture, and it does so by convolving the input of a square-resolution picture with a filter, also known as a kernel [17]. In a broad sense, we might say that this 2D convolution procedure uses the sliding window idea to calculate its weight [18].

$$(y,x) = \sum_{p=-m2}^{m2} \sum_{q=-n2}^{n2} h(p+m2+1, q+n2+1) f(y-p, x-q) \quad (2)$$

If we apply the above equation, we get the following representation of the convolution's final image:

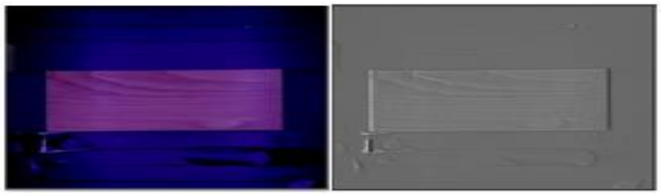


Fig. 5. Visualization of Convolution Layer.

One of the 3 x 3 pixel filters/kernels used in the convolution layer's 2D convolution operations is shown in color in Fig. 5. The study found that 64 features might be used on the first convolution layer when using input dimensions of 224 × 224 × 3. Then, the convolutional layer's output is turned on, which is the second part of the feature extraction procedure. One component of CNN architecture, the activation layer, can prevent vanishing gradients and speed up the convergence process during training [19]. As an activation function, ReLu will swap out negative pixel outputs for zeros without touching the original pixel values [20]. The corresponding mathematical expressions are as follows:

$$ReLu = max\,(y, 0) \quad (3)$$

A new image will be created that is shown in Fig. 6 by plugging the above algorithm into the convolutional layer's output.

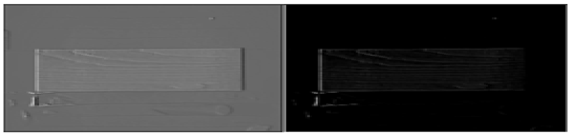


Fig. 6. Visibility of the Activation Layer.

Most of the pixels in the final image produced by the above activation are zero, and hence the image is predominantly black. It will shed lighter on the characteristics and make it easier to calculate data for use in either training or testing. Using a pooling layer as the last stage in feature extraction. The pooling layer is the final stage of feature extraction, and its job is to select the most informative aspects of the activation function at the output layer [21]. It aims to simplify features and accelerate computing time. The challenge is to implement the equation to do down sampling on each pixel of the image:

$$Pooling = max\,(y, 0) \quad (4)$$

Feature clarity can be achieved using the down sampling method, as seen in Fig. 7. Reduce the size of individual pixels to reduce the time needed to classify data in the fully linked layer. The Fully Connected Layer is a Multilayer Perceptron (MLP) that makes use of the softmax activation function.

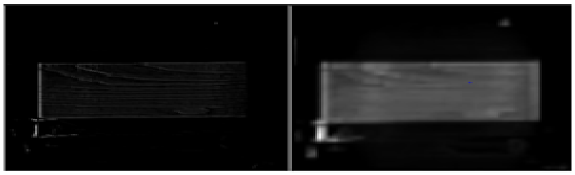


Fig. 7. Visualization of Pooling Layer.

b) Training Process: In this stage, the final probability is computed using the extracted feature. Fully connected layer is an MLP that uses the softmax activation function. It is the responsibility of this layer to take the extracted feature and process it so that the final probability is obtained. This layer also does categorization by selecting the weighted product of the highest value of the flattened results. Using the following simple equation, weight values will be refined until the fewest losses are incurred:

$$W'^{l}_{n,m} = W^{l}_{n,m} - \eta \frac{\partial L}{\partial W^{l}_{n,m}} \quad (5)$$

As such, a loss value of zero represents the best possible categorization likelihood. As used here, "weight" refers to a property of the objects being categorized. Once losses are zero, the following computation will be used to discontinue updating weight:

$$\text{Loss Function} = \sum_{k=1}^{n}(Y - \overline{Y})^2 \quad (6)$$

c) Classification Process: For the record, the multiclass system relies on the Softmax Classifier Function to properly label each class before making a final prediction. One type of function activation is the Softmax classifier. In a multiclass classification problem, the softmax function must be used in the output layer [22]. It's utilized in MLP, and its values range from zero to one, with the overall probability adding up to one [23]. It is undoubtedly essential to evaluate a class prediction on a machine-learning model because, with a softmax classifier, users will quickly know the level of confidence in the system to make predictions [24]. Sometimes the system predicts classes correctly with a confidence level below 50%, while the system predicts wood classes correctly with a confidence level above 50%. Every chance of a multiclass event occurring with a value of one is expressed in the following equation.

$$\text{fj (y)} = \frac{e^{y_i}}{\sum_{k} e^{y_k}} \quad (7)$$

For each jth member in the class output vector, the function's result is displayed using the fj notation. For classification purposes, the training model provides the hypothesis y as the argument to the Softmax function.

2) Support Vector Machine (SVM) Method

a) Pre-Processing: During the SVM classification procedure, the acquired image will be converted to grayscale and scaled to 427 x 240 pixels. Several tests utilizing three image sizes, including the original size of 1280 x 720 pixels, 640 x 360 pixels, 427 x 240 pixels, and 320 x 180 pixels, are required. The results demonstrated that the 427 x 240 pixels have an accuracy of 100 percent and a computation time of 3.08 seconds. After determining the optimal size for image capture, the photographs will be converted to grayscale. The straightforward equation for converting RGB to grayscale is as follows [25], [26]:

$$Gray = (Red * 0.3 + Green * 0.59 + Blue * 0.11) \quad (8)$$

With this equation, the image could be transformed into grayscale (see Fig. 8):

Fig. 8. Grayscale Conversion Image.

b) Feature Extraction: HOG features are the next step. Here, we apply 1-D centering to determine the point of the discrete derivative mask in the x and y directions to compute the gradient value [24] (see Eq. 9 and 10).

$$L_x(r, c) = l(r, c + 1) - l(r, c - 1) \quad (9)$$

$$L_y(r, c) = l(r, c + 1) - l(r, c - 1) \quad (10)$$

The equation for the gradient's magnitude is as follows:

$$\mu = \sqrt{lx^2 + ly^2} \quad (11)$$

Furthermore, gradient orientation is given by:

$$\theta = \tan^{-1} \frac{ly^2}{lx^2} \quad (12)$$

Bin grouping based on spatial orientation is the next step. The goal of this procedure is totally votes and provide a histogram of cellular activity. To determine the image's overall orientation, each pixel will be assigned to the nearest bin between 0 and 180 degrees. As a next step, the HOG descriptor is used to convert the cellular and histogram data to a vector space normalization. Using the L2-Hys norm, the block normalization is carried out as follows as the final step:

$$b = \frac{b}{\sqrt{|b|^2 + \varepsilon^2}} \tag{13}$$

The limit of renormalization is b = 0.2. The Feature used to implement this HOG is depicted in Fig. 9.

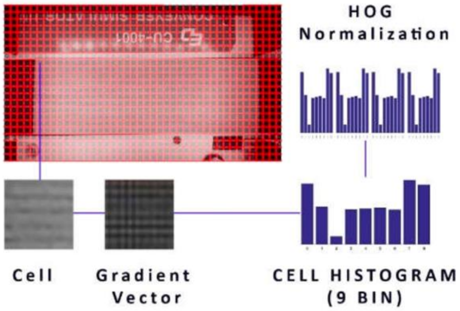


Fig. 9. Feature Process of HOG to Identify the Wood.

For this experiment, we will partition the detection window of 427 × 240 pixels into 966 chunks, 42 in width and 23 in height. There are four cells in each block, and each cell's histogram has 36 values spread out across nine bins. This increases the total size of the vector to 42 blocks across by 15 blocks vertically by 4 cells/block by 9 bins/histogram, or 22,680 values. The *.csv file format can take this final vector as input.

c) Classification Process

As a final stage, we employ a Support Vector Machine. The goal of this process is to assign a category to each photograph. In addition, SVMs enhance the picture of each class by employing hyperplanes or discriminatory boundaries. The margin at the maximal hyperplane point is used to choose the optimal hyperplane. We can express the linear classification hyperplane as [27]:

$$f_{svm}(x) = \sum i \in N a_i y_i K(x_i, x) + b \tag{14}$$

$$class = \begin{cases} 1, & f_{svm}(x) \geq 0 \\ -1, & others \end{cases} \tag{15}$$

C. Indicator System

This section is the last working state of the system that will be created, where the Command that comes from the classification results becomes an input to the microcontroller to function the installed plant, namely the indicator lights with five different colors.

The flowchart above in Fig. 10 shows the last working state in the wood classification system starting from receiving commands sent from the server by the serial communication method to the microcontroller, which will give the state high to each lamp with a relay intermediary. When the prediction results state that the test wood is class A wood, the server will give an Arduino command with the serial communication method to contact the relay so that it becomes NC and makes the red light turn on. Likewise, on the results of other conditions, a similar process will be carried out according to the predicted wood class— LED lights are employed as the indicator and will light up in response to data from the server. Signals will be transmitted via serial communication to an Arduino that will then activate relays and LED lights as expected. Signals that are transmitted as coded signals based on the classification of the wood, with the following stipulations: Light color codes: "A" for red, "B" for yellow, "C" for white, "D" for green, and "E" for blue.

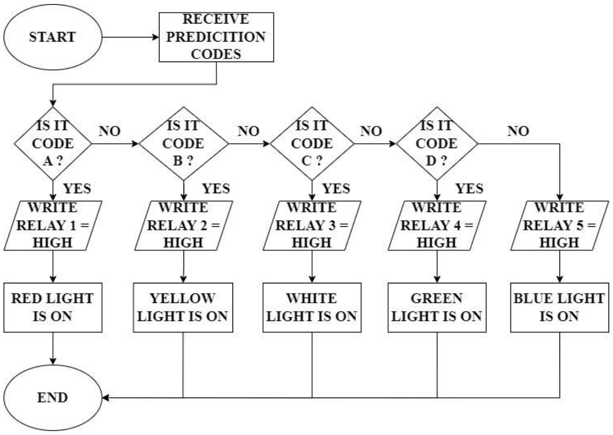


Fig. 10. Flowchart of Indicator System.

IV. Result and Discussion

The algorithm utilized by the cedarwood classification system is a CNN and SVM written in Python with the Open CV library. Up to 50 boards of the cedarwood utilized as testing material were examined. The dimensions of the wood for classes A, B, and D are 18.3x6.2x0.45 cm, while the measurements for classes C and E are 18.3x7.6x0.45 cm. The cedarwood training dataset reached 2,830 instances due to data enhancement using a condition that involves rotating through 180 degrees and flipping under the conveyor condition. This section provides an overview of the system. Fig. 11 is the overall design of the system that implemented. The system consists of nine pieces of hardware use like conveyors, ultrasonic sensors, laptop that use for server, Arduino uno, image acquisition boxes, webcams, led strips, indicator lights, and 8 channel of relays.

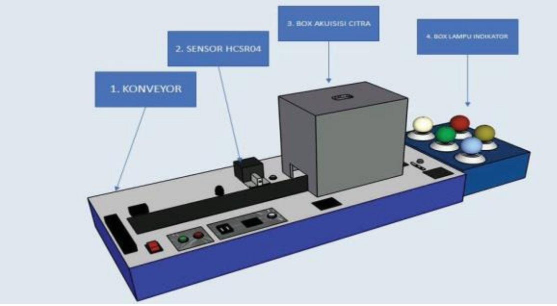


Fig. 11. Integrated System with Plant.

A. Observation of Data Distribution Effect Result

Observation of data distribution is done to find the most optimal model when the training process on the computer occurs. The augmented dataset is divided into four distribution variations which are illustrated in Table II.

TABLE II. DATA DISTRIBUTION VARIATIONS

Data Distribution	Training Data	Test Data	Total Datasets
60 : 40	1710	1140	2850
70 : 30	1995	855	2850
80 : 20	2280	570	2850
90 : 10	2565	285	2850

The goal of the data distribution experiment is to evaluate the system's performance in terms of the distribution of the input data using wood from datasets that were chosen at random depending on each class, the experiment was run 20 times, ten times from the front view and ten times from the back perspective, with the following findings from both CNN and SVM.

The experiment result shown in Fig. 12 shows that the 90:10 dataset experiment is the best accuracy in both SVM and CNN methods. It indicates that as the proportion of training data increases, the system will generalize more object attributes of each class, also data variability is more important than the testing data, and identification will be better when the training dataset is more significant than the testing data. So, the 90:10 dataset will be used as the parameter for the next following experiment of both methods.

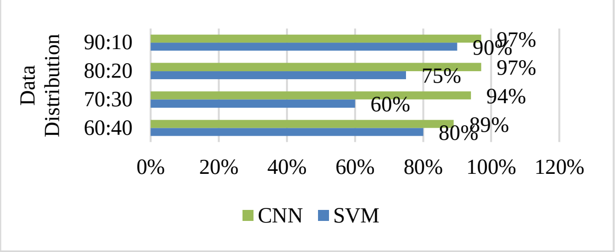


Fig. 12. Data Distribution Experiment Result.

B. *Experimental Result of System using SVM Method*

There are three experiments carried out to know the system’s performance using the SVM method:

- The test angle for the camera position.
- The parameters for the HoG test.
- The parameters for the SVM test.

The first experiment involves changing the angle of the camera position test. The goal of this experiment was to find the best camera angle for this system. On the camera's surface, five angle sites will be established: Five degrees to the right, ten degrees to the right, zero degrees, five degrees to the left, and ten degrees to the left. As a result, the findings are shown in Table III.

Table III demonstrates that a 90 percent success rate can be attained with a camera positioned at a zero-degree angle. Similarly, time computation is important but not always a constrained-range issue. The conveyor belt serves as the background for the training dataset consisting of wood fibers. It will be impacted by the system for it to detect the things. Predicated on the work of Sahki et al. [28], His article concerning HOG-based Fast person detection utilized a two-folder training database from the INRIA pedestrian database. Each folder contains both "post" and "neg" photos. Negative training or test photos, such as a park, ocean, cityscape, etc., are included in the neg post, while the positive training or test photographs centered on a person with their left-right reflections are included in the post. These conditions may facilitate recognition of the object in a different position inside the image. It might be concluded that 0 degrees Celsius will be employed in this experiment.

TABLE III. OVERALL RESULT OF ANGLE CAMERA EXPERIMENT

	Right Side		0º	Left Side	
	+10˚	+5˚		+5˚	+10˚
Accuracy	0	30	90	40	10
Duration (s)	1.61	1.45	1.58	1.61	1.61

The second experiment is a test of HOG parameters to determine the optimal orientation, block norm, pixel/pixel, and block/pixel parameters. Fig. 13 and 14 depict the outcome of the experiment involving the modification of each HOG indicator. There are several methods for computing precision and time. The ideal parameter for this test is 100% with a computation length of 2 seconds, pixel/cell 10 x 10, and cell block 2 x 2. The cells/block indicator shows that each 2x2 is more essential than a 4 x 4. That data points to a 2 x 2 layout being preferable to a 4 x 4 one in this system. This result is shown in Fig. 13.

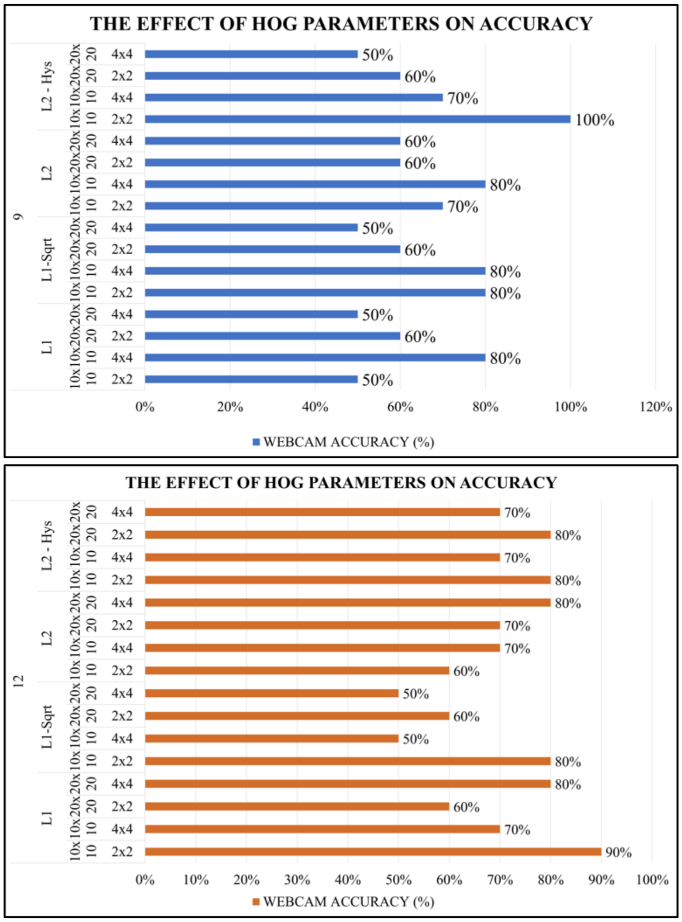


Fig. 13. The Accuracy Rate of HOG Feature (%).

In addition, for the pixel-per-pixel indicator, 10x10 is superior to 20 x 20, indicating that the gradient produces greater detail and precision. In addition, the Block value prevents images from having excessive brightness and contrast. This technique is ideal for controlling cells per 2-by-2 block and pixels every 10-by-10 pixel. Next, the indicator for the block norm has a variable value; it has not yet been affected. Block normalization minimizes brightness and contrast variations in an image depending on neighboring gradient cells. By dividing each vector element by its vector length, it aims to preserve the image's luminance and contrast [29]. The result is shown in Fig. 14.

Last is orientation; the average accuracy of the nine orientation values is 66% for a total of nine values. It indicates that the accuracy is less than 12 orientations. The 12 orientation value has an average performance of 70%. Therefore, if the system employs a greater orientation value, the optimum analysis for picture detail and gradient value will be compromised. It is possible to determine that orientation 9, block norm L2-Hys, pixel/cell 10 x 10, and cell block 2 x 2 are optimal for this system and provide the highest performance rate.

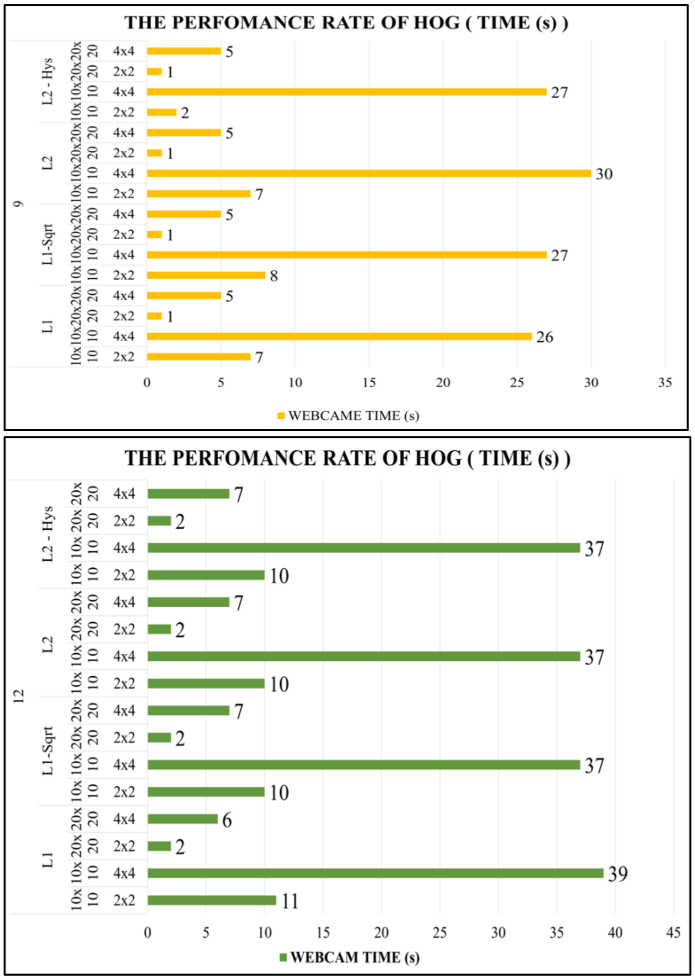
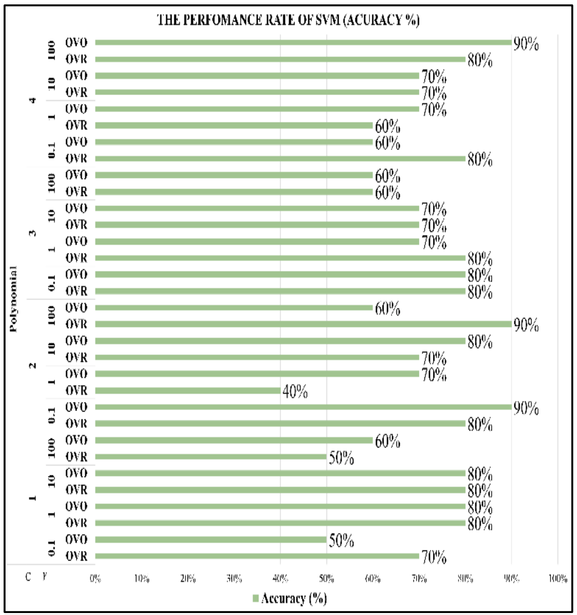
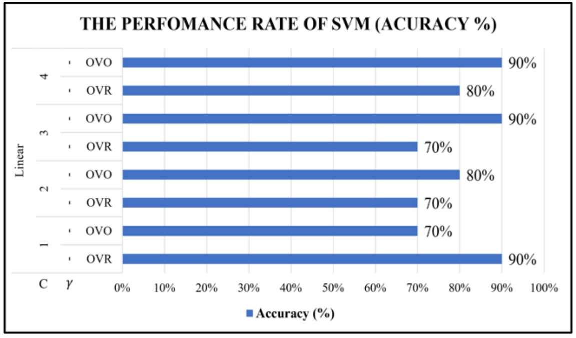


Fig. 14. Time Computation Rate of HOG Feature (s)

In this section, the objective of the third experiment is to determine the multiclass type, gamma, and c value, optimal kernel type parameters. The outcome of this experiment is depicted in Fig. 15 and 16.

Fig. 15 and 16 shows that the linear kernel with a one-versus-rest multiclass produced the best performance rate at 1.40 seconds for 90% accuracy. One-to-one accuracy in the radial basis function kernel resulted in the least accurate solution. The highest accuracy can be achieved in this system by using a linear kernel. Due to the database creation process, the image array will become flatter. This implies that the data from each column-major column of the picture array has been moved to a single row or flattened into a 1-D iterator through the array. This database has been converted to text format, with column one indicating the categorization type and columns two through the end carrying picture codes. The soft margin cost function is controlled by a parameter called C or gamma, which controls the effect of each support vector. This process involves trading error penalty for stability. Text-classification-appropriate kernels include linear kernels. Radial Basis Function (RBF) is a nonlinear sample on a higher-dimensional space. In contrast to the linear kernel, it can handle situations when the relationship between class labels and attributes is nonlinear. Nonlinear kernels, such as a polynomial kernel, offer advantages over linear and RBF kernels in terms of speed and accuracy. Since the best performance rate and time for the C = 1 condition using multiclass OVR and a linear kernel are 90% and 1.40 seconds, respectively, this suggests that a linear kernel could be used in this situation.





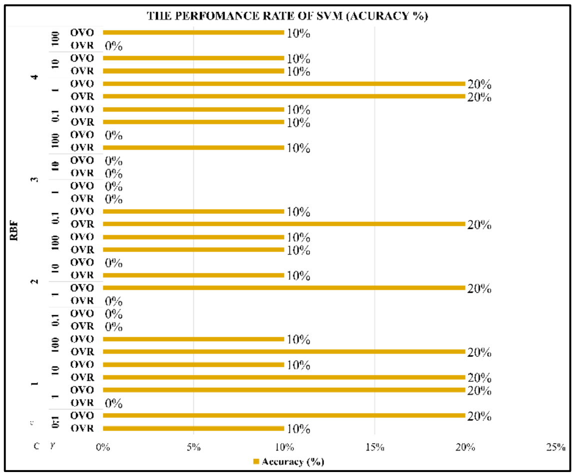


Fig. 15. The Accuracy Rate of SVM (%).

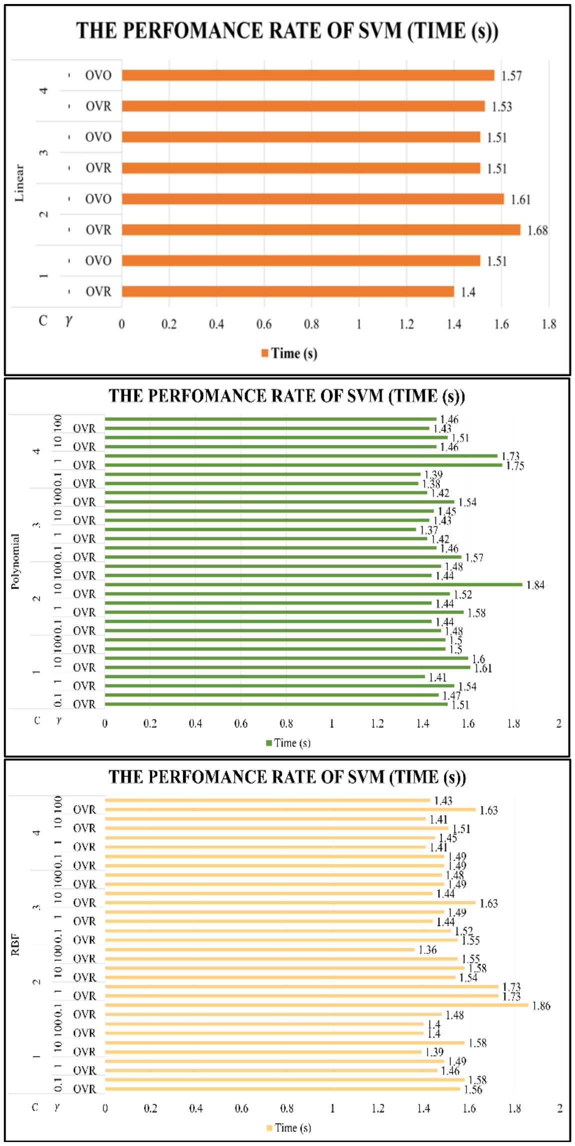


Fig. 16. The Time Computation Rate of SVM (s).

C. Experimental Result of CNN Method

To evaluate the performance of the wood classification system using the CNN method training parameter test, which includes observations on number of epochs, batch size, learning rate, and optimizer the CNN training parameter test trials must be conducted. The following Table IV is a scan of CNN algorithm parameter observations.

TABLE IV. CNN TRAINING PARAMETERS

No	CNN Training Parameters			
	Learning Rate	Batch Size	Epoch	Optimizer
1	0.001	8	1	Adam
2	0.0001	16	5	Rmsprop
3	0.00001	32	10	Adagrad
4	0.000001	64	15	SGD

The experiments will be conducted by modifying the Table III parameters. The subsequent experiment evaluates the learning rate as a CNN training parameter. The goal is to measure how quickly the system converges on the global minimum point, where performance is maximized. When the system reaches the greatest levels of accuracy, precision, and recall with variations in the learning rate of 0.000001, 0.00001, 0.0001, and 0.001, then the conditions have been met. Fig. 17 displays the outcomes of the tests.

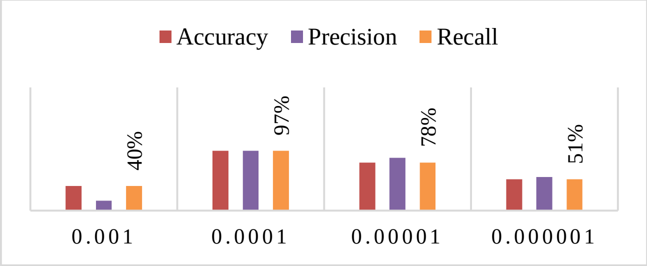


Fig. 17. Observation of Learning Rate Effect.

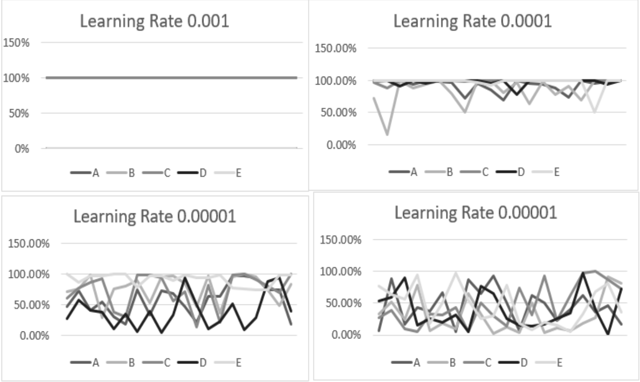


Fig. 18. Optimization of the Predicted Results from each Learning Rate.

Based on Fig. 17 and 18, which depict the system's hopeful attitude toward the expected object, the system's ideal learning rate is 0.0001. A learning rate of 0.001 is deemed excessively high, as it causes the system to diverge. The alternative is that training accuracy does not improve as quickly as with a learning rate of 0.000001 since the system cannot reach the global region of minima due to the rapid rise in weight and numerous errors throughout the rebuilding process. The system rejects the input as being too low. Therefore, the length of the step reaches the minimal global area, yielding inferior accuracy, because the iterative step for updating the weight is quite small.

The second experiment of CNN training batch size analyzes the optimal test-time learning rate. Finding the optimal model for generalizing features and speeding the training process for wooden items is the objective. The observed batch sizes are 8, 24, and 32, as shown in Fig. 19, which depicts the experimental outcomes.

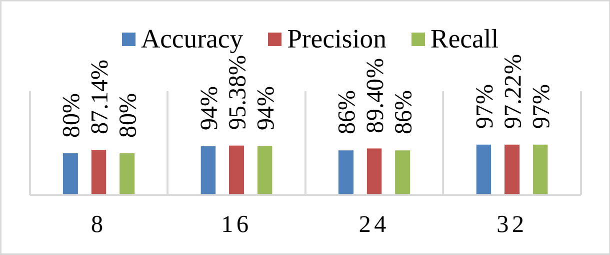


Fig. 19. Observation of Batch Size Effect.

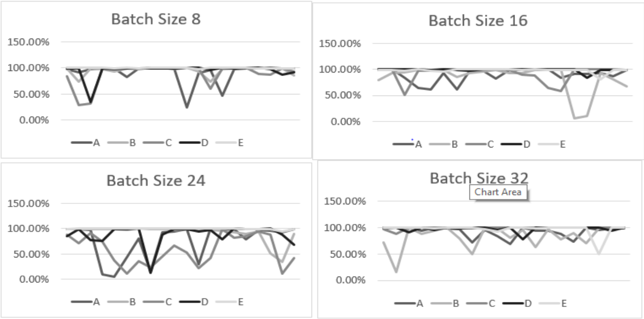


Fig. 20. Optimization of the Predicted Results from each Batch Size.

According to Fig. 19 and 20, the best batch size for a learning rate of 0.0001 is 32. A batch size that is too big can reduce accuracy since changing the weight per batch can result in errors. However, if the batch size is too small, the lengthy training period and system workload will increase. For this reason, it is absolutely necessary to adjust the learning rate parameters in accordance with the batch size in order to obtain precision and the optimum amount of training time [30].

Considering the best parameters from previous experiments, the number of epochs is the subject of the third experiment of CNN training parameters. This test is designed to identify the system with the best performance, one that is neither underfitted nor overfitted. 1, 5, 10, and 15 epochs are utilized, with the performance characteristics depicted in Fig. 21.

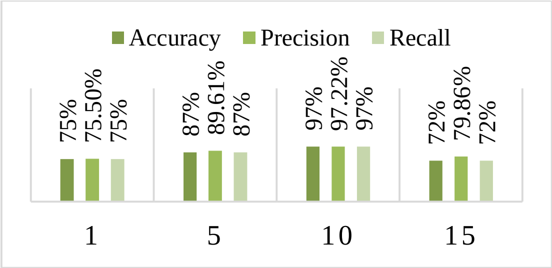


Fig. 21. Observation of Epoch Effect.

Experimentally, training with ten epochs yields the best performance as it can see in Fig. 22. A training cycle for all datasets is one epoch. Too few epochs will induce underfitting in the model we are training, hence the resultant accuracy is typically subpar. In contrast, if there are too many epochs, the training model may become overfit. Consequently, the system is less capable of generalizing the characteristics of the first new timber system site.

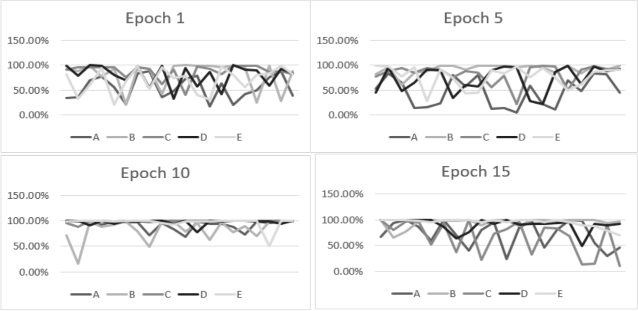


Fig. 22. Optimization of the Predicted Results from each Epoch.

The fourth CNN training parameter is the optimizer, which requires consideration of the preceding three experiment parameters. The optimizer is intimately connected to the system's ability to lower the value of losses and find the most effective actions to reach the global lowest point. The observed optimizer algorithms are SGD, adagrad, rmsprop, and adam, with test results depicted in the Fig. 23.

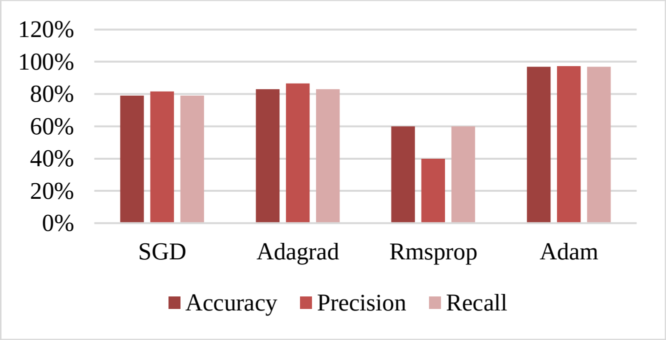


Fig. 23. Observation of Optimizer Effect.

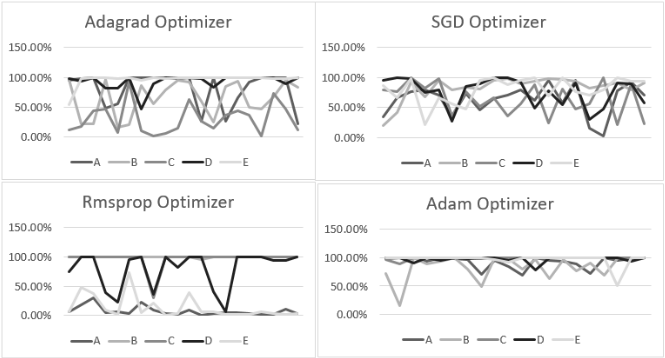


Fig. 24. Optimization of the Predicted Results from each Optimizers.

Based on Fig. 23 and 24, Adam Optimizer provides the most ideal performance. Adam is the most recent optimizer based on the notion of integrating SGD with momentum and rmsprop to tune the learning rate for each system network weight.

V. Conclusion

In conclusion, this research shows that a cedarwood classification system based on the CNN and SVM approach correctly recognized five wood classes using the features listed in Table I. SVM's classification output revealed that this system's performance rate is 90%, with a computation time of 1.40 seconds. The value derived from the experiment using the difference in the data set consists of 90% training data and 10% test data. In addition, the camera's angular position is 0°. Utilizing a linear kernel and one-versus-rest multiclass, the SVM classifier used in this study was useful. This condition easily categorizes and identifies the texture and fiber pattern of cedar. In contrast, the CNN improved the system's performance to 97% accuracy with an average computation time of 0.5472s. The value is calculated by comparing training and testing data distributions of 90:10, CNN training parameters of 0.0001, batch size of 32, epoch of 10, adaptive moment estimate as an optimizer and 723 lux of light intensity. It reveals that CNN is a significantly more effective method for image classification than SVM classification utilizing the HOG Feature.

Acknowledgment

The authors would like to express their gratitude to those who helped make this research possible from the Telkom University, School of Electrical Engineering and the Universitas Tanjungpura, Department of Electrical Engineering.

References

[1] M. I. Taqyudin, B. Irawan, and C. Setianingsih, “Wood Classification Based on Fiber Texture Using Backpropagation Method,” in 2019 International Conference on Sustainable Engineering and Creative Computing (ICSECC), Aug. 2019, pp. 245–250. doi: 10.1109/ICSECC.2019.8907197.

[2] J. Feng, P. Dong, R. Li, C. Li, X. Xie, and Q. Shi, “Effects of wood fiber properties on mold resistance of wood polypropylene composites,” Int Biodeterior Biodegradation, vol. 140, pp. 152–159, May 2019, doi: 10.1016/j.ibiod.2019.04.005.

[3] V. Bucur, “Properties of Wood Species for Percussion Instruments,” in Handbook of Materials for Percussion Musical Instruments, Springer, 2022, pp. 695–785.

[4] S.-W. Hwang and J. Sugiyama, “Computer vision-based wood identification and its expansion and contribution potentials in wood science: A review,” Plant Methods, vol. 17, no. 1, p. 47, Dec. 2021, doi: 10.1186/s13007-021-00746-1.

[5] L. Novakova, “The impact of technology development on the future of the labour market in the Slovak Republic,” Technol Soc, vol. 62, p. 101256, Aug. 2020, doi: 10.1016/j.techsoc.2020.101256.

[6] S. Hartini, U. Ciptomulyono, M. Anityasari, and Sriyanto, “Manufacturing sustainability assessment using a lean manufacturing tool,” International Journal of Lean Six Sigma, vol. 11, no. 5, pp. 943–971, Nov. 2020, doi: 10.1108/IJLSS-12-2017-0150.

[7] T. Han and G. A. Sánchez-Azofeifa, “A Deep Learning Time Series Approach for Leaf and Wood Classification from Terrestrial LiDAR Point Clouds,” Remote Sens (Basel), vol. 14, no. 13, p. 3157, Jul. 2022, doi: 10.3390/rs14133157.

[8] G. Bonaccorso, Machine Learning Algorithms: Popular algorithms for data science and machine learning. Packt Publishing Ltd, 2018.

[9] G. Bonaccorso, Machine learning algorithms. Packt Publishing Ltd, 2017.

[10] M. You and C. Cai, “Wood Classification Based on PCA, 2DPCA, (2D)2PCA and LDA,” in 2009 Second International Symposium on Knowledge Acquisition and Modeling, 2009, pp. 371–374. doi: 10.1109/KAM.2009.321.

[11] R. A. Pramunendar, C. Supriyanto, Dwi Hermawan Novianto, Ignatius Ngesti Yuwono, G. F. Shidik, and P. N. Andono, “A classification method of coconut wood quality based on Gray Level Co-occurrence matrices,” in 2013 International Conference on Robotics, Biomimetics, Intelligent Computational Systems, Nov. 2013, pp. 254–257. doi: 10.1109/ROBIONETICS.2013.6743614.

[12] D. Shustrov, “Species Identification of Wooden Material using Convolutional Neural Networks ,” Lappeenranta University of Technology, 2018.

[13] X. Zhuang, D. Zhou, X. L. Yu, and Z. M. Zhao, “An Image Processing and Laser Ranging Approach for Radio Frequency Identification (RFID) Tag Group Reading Performance Prediction.,” Lasers in Engineering (Old City Publishing), vol. 51, 2021.

[14] C. B. Gonçalves, J. R. Souza, and H. Fernandes, “CNN architecture optimization using bio-inspired algorithms for breast cancer detection in infrared images,” Comput Biol Med, vol. 142, p. 105205, Mar. 2022, doi: 10.1016/j.compbiomed.2021.105205.

[15] W. R. PERDANI, R. MAGDALENA, and N. K. CAECAR PRATIWI, “Deep Learning untuk Klasifikasi Glaukoma dengan menggunakan Arsitektur EfficientNet,” ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, vol. 10, no. 2, p. 322, Apr. 2022, doi: 10.26760/elkomika.v10i2.322.

[16] M. Giménez, J. Palanca, and V. Botti, “Semantic-based padding in convolutional neural networks for improving the performance in natural language processing. A case of study in sentiment analysis,” Neurocomputing, vol. 378, pp. 315–323, Feb. 2020, doi: 10.1016/j.neucom.2019.08.096.

[17] W. Huang, J. Cheng, Y. Yang, and G. Guo, “An improved deep convolutional neural network with multi-scale information for bearing fault diagnosis,” Neurocomputing, vol. 359, pp. 77–92, Sep. 2019, doi: 10.1016/j.neucom.2019.05.052.

[18] F. Millstein, Convolutional neural networks in Python: beginner’s guide to convolutional neural networks in Python. Frank Millstein, 2020.

[19] L. Alzubaidi et al., “Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions,” J Big Data, vol. 8, no. 1, pp. 1–74, 2021.

[20] [20] W. Ouyang, B. Xu, J. Hou, and X. Yuan, “Fabric Defect Detection Using Activation Layer Embedded Convolutional Neural Network,” IEEE Access, vol. 7, pp. 70130–70140, 2019, doi: 10.1109/ACCESS.2019.2913620.

[21] Y. H. Liu, “Feature Extraction and Image Recognition with Convolutional Neural Networks,” J Phys Conf Ser, vol. 1087, p. 062032, Sep. 2018, doi: 10.1088/1742-6596/1087/6/062032.

[22] S. Maharjan, A. Alsadoon, P. W. C. Prasad, T. Al-Dalain, and O. H. Alsadoon, “A novel enhanced softmax loss function for brain tumour

detection using deep learning," J Neurosci Methods, vol. 330, p. 108520, Jan. 2020, doi: 10.1016/j.jneumeth.2019.108520.

[23] S. Widiyanto, R. Fitrianto, and D. T. Wardani, "Implementation of Convolutional Neural Network Method for Classification of Diseases in Tomato Leaves," in 2019 Fourth International Conference on Informatics and Computing (ICIC), Oct. 2019, pp. 1–5. doi: 10.1109/ICIC47613.2019.8985909.

[24] J. Wu, L. Chang, and G. Yu, "Effective Data Decision-Making and Transmission System Based on Mobile Health for Chronic Disease Management in the Elderly," IEEE Syst J, vol. 15, no. 4, pp. 5537–5548, Dec. 2021, doi: 10.1109/JSYST.2020.3024816.

[25] B. Sugiarto et al., "Wood identification based on histogram of oriented gradient (HOG) feature and support vector machine (SVM) classifier," in 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Nov. 2017, pp. 337–341. doi: 10.1109/ICITISEE.2017.8285523.

[26] V. A. Gunawan, L. S. A. Putra, F. Imansyah, and E. Kusumawardhani, "Identification of Coronary Heart Disease through Iris using Gray Level Co-occurrence Matrix and Support Vector Machine Classification," International Journal of Advanced Computer Science and Applications, vol. 13, no. 1, 2022, doi: 10.14569/IJACSA.2022.0130177.

[27] N. Boyko and R. Hlynka, "Application of Machine Algorithms for Classification and Formation of the Optimal Plan.," in COLINS, 2021, pp. 1853–1865.

[28] M. Kachouane, S. Sahki, M. Lakrouf, and N. Ouadah, "HOG based Fast Human Detection," Jan. 2015, doi: 10.1109/ICM.2012.6471380.

[29] A. Rana, P. Singh, G. Valenzise, F. Dufaux, N. Komodakis, and A. Smolic, "Deep Tone Mapping Operator for High Dynamic Range Images," IEEE Transactions on Image Processing, vol. 29, pp. 1285–1298, 2020, doi: 10.1109/TIP.2019.2936649.

[30] S. L. Smith, P.-J. Kindermans, C. Ying, and Q. v. Le, "Don't Decay the Learning Rate, Increase the Batch Size," Nov. 2017.