

Classification of Cedar Wood Quality using Convolutional Neural Network Method

1st Renal Farhan

School of electrical engineering
Telkom University
Bandung, Indonesia

2nd Muhammad Ary Murti

School of electrical engineering
Telkom University
Bandung, Indonesia

3rd Casi Setianingsih

School of electrical engineering
Telkom University
Bandung, Indonesia

Abstract— As a fulfillment of household appliance needs, cedarwood is one of the most sought after materials [1]. Apart from its distinctive fragrant, quality is the main point of concern. The quality of this wood can be classified based on fiber patterns. In general, the wood processing industry does the classification process manually by relying on the sense of sight. As a result of the accuracy and time efficiency also varies so that it can reduce the credibility of the local wood industry. machine learning is the solution to solve that problem. Some researches have been done, one of which uses the HOG feature and SVM classification with an accuracy of 90% and a time of 1.40 seconds [2]. However, in the industrial era 4.0 which incidentally paid great attention to technological updates, so in this paper, the writer implements one method of deep learning in the classification system of cedarwood, the Convolutional Neural Network. The dataset used consists of five classes: Class A, Class B, Class C, Class D and Class E. The feature extraction process is carried out at the convolution, activation, and pooling layers. Total layers used are 16 weight layers with input in the form of images are taken automatically using the Logitech Brio 4K integrated with Arduino Uno and ultrasonic sensors. The experimental results showed a significant improvisation with an accuracy of 97% and a prediction speed of 0.56 seconds.

Keywords— Cedarwood, convolutional neural network (CNN), digital image processing.

I. INTRODUCTION

Cedarwood is one of the light wood processed materials that are most in-demand by consumers, both as an exterior material or household appliance needs [1]. Noted potential transactions of light wood products of 5.64 million USD from 320 buyers at the Interzum furniture exhibition, Germany, which was held in the last May 2019 [3]. Apart from its distinctive aroma, quality is the main point of concern. The quality of this wood can be classified based on color, texture, and fiber pattern.

In general, industries engaged in wood processing do the classification process manually by relying on the sense of sight and instinct as a comparison of the many similar objects. It gives the percentage of conformity of observations of only 50 - 60% [4]. As a result, the accuracy and efficiency of production time also vary, tend to be subjective, and can reduce the credibility of the Indonesian processed wood industry. Therefore, the branch of machine learning is a solution so that the local wood industry can classify wood that is able to do processed classification with consistent quality standards automatically.

In the industrial era 4.0, the implementation of machine learning theory is already widespread and it is possible to be implemented in the local wood industry in order to maintain its

quality and credibility. Some related research has also been carried out in Indonesia, one of which is the classification of wood quality using HOG features and SVM classification with an accuracy of 90% and a computing time of 1.40 seconds [2].

Therefore, this research is proposed to develop this research by implementing one of the deep learning methods into the cedar wood quality classification system, the Convolutional Neural Network algorithm which is integrated directly with the mini conveyor, Arduino Uno and Logitech Brio 4K webcam to realize the auto features capture and predict when the wood starts to enter the classification area. The dataset used consists of five classes of cedarwood quality: Class A, Class B, Class C, Class D, and Class E, and the feature extraction process is carried out automatically with a total of 16 layers used.

II. THEORY AND METHODOLOGY OF THE SYSTEM

In this paper, the methodology that is used in the cedar wood classification system based on the Convolutional Neural Network Algorithm is divided into three work states namely image acquisition, training & classification system, and indicator systems. The first state has the task of doing auto-capture with the help of an ultrasonic sensor that will detect the presence of wood on the conveyor path and make it an input signal system. The signal is sent to the server via Arduino with the serial communication method, then the server will forward the command to the Logitech brio 4k webcam to capture it as a feed of the training & classification system. The second state has the task to do machine learning and classifies captured images by automatically extracting features through the convolution, activation, and max-pooling layers. The results of the learning carried out in the second state are displayed on an indicator system consisting of five LED lights with different color indications that symbolize each class of wood.

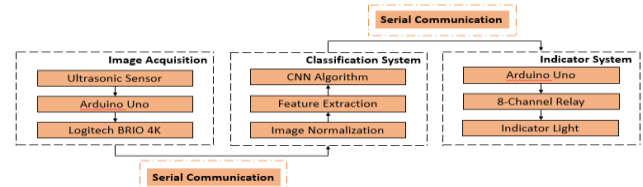







Fig. 1. Block diagram of the classification system.

A. Image Acquisition

Cedarwood that runs on the conveyor track will be automatically captured by Logitech brio 4K on instructions from the server that has received a signal from HC-SR04 which functions as a proximity sensor and integrated with Arduino as a controller to detect the presence of objects. Capturing the

image is carried out in an acquisition box that has dimensions of 27.6 cm x 23.6 cm x 30 cm with matte black color with LED strip lighting on the ceiling with a light intensity of 723 lux. The resulting image has a resolution of 640 x 480 pixels with the following wood characteristics:

TABLE I. CHARACTERISTICS OF WOOD CLASS

| Class | Figure | Description |
|-------|---|--|
| A |  | Fiber is prodigious and straight, the distance of fiber is so close until it is almost invisible |
| B |  | Fiber is many, straight and rather thick and visible distance among the fibers |
| C |  | The fibers are clearly visible, the distance between the fibers is far away, as well as some transverse fibers |
| D |  | The fibers are clearly visible, the distance between the fibers apart and the transverse pattern |
| E |  | Fiber is very clear, random fiber patterns tend not to be straight |

The picture above shows the results of the capture of cedarwood images of each quality with a black background, the brightness of 26, contrast by 24, and camera focus by 14 with a zoom-in percentage of 130%.

B. Training and Classification System

In this step, the acquisition image will be used as a feed for the first layer on the CNN architecture, but before that, it is necessary to pre-process it first to get a dimensional match between the image and the layer. The parameter of suitability in the CNN architecture VGG-16 algorithm used in the wood classification system this time is a square-shaped image with dimensions measuring 224x224x3 pixels. The following is an illustration of a simple calculation of the image resizing operation.

| | | |
|----------|------------|------------|
| P(q,r) | P(q+1,r) | P(q+2,r) |
| P(q,r+1) | P(q+1,r+1) | P(q+2,r+1) |
| P(q,r+2) | P(q+1,r+2) | P(q+2,r+2) |

Fig. 2. Illustration of 3x3 pixel image.

Figure 2 illustrates a two-dimensional 3x3 pixel image with different index variations. In this process, a sample of 4 pixels including P (q, r) was taken; P (q + 1, r); P (q, r + 1) and P (q + 1, r + 1). Assuming the value to be changed is P (q, r), the determination of the value is determined based on the average of all values at 4 pixels associated with the formula as follows:

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

Based on the formula above, the image will be resized to be smaller as in Figure 3.

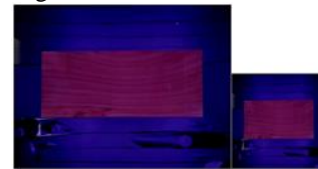


Fig. 3. Resizing image.

After the input image matches the first layer of CNN, a feature extraction process is then performed which aims to collect features in each wood class. The first step in the feature extraction process is the 2D convolution operation.

- Convolutional Layer

Convolution Layer is a layer on the CNN architecture that functions as the first extracting feature by convoluting an input in the form of an image that has a square resolution with a filter or commonly referred to as a kernel. In general, it can be said that this 2D convolution process adopts the sliding window concept to get the value of its weight [5]. The following figure is an illustration of the convolution process.

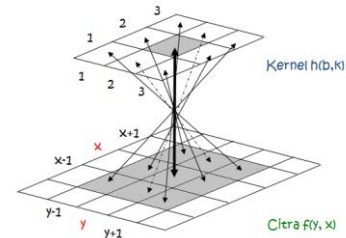


Fig. 4. Convolutional Operation [6].

A 5x5 pixel image that is convoluted with a vertical square filter with a size of 3x3 and produces a new image measuring 3x3 pixels containing one particular feature of the original image with pixel values can be calculated through the following formula:

$$g(y,x) = \sum_{p=-m_2}^{m_2} \sum_{q=-n_2}^{n_2} h(p+m_2+1, q+n_2+1) f(y-p, x-q) \quad (2)$$

Based on the formula above, the new image from the convolution can be visualized as follows:

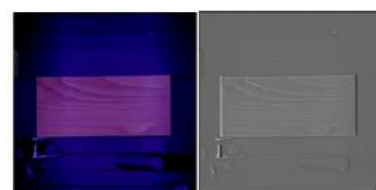


Fig. 5. Output of convolutional layer.

Fig. 5 is a visual representation of a color channel from 2D convolution operations performed on one of the 3x3 pixel filters/kernels on the convolution layer. Based on the research results obtained total feature maps on the first convolution layer with 224x224x3 input dimensions totaling 64 features. Then the second step in the feature extraction process is to activate the output of the convolutional layer.

- Activation Layer

The activation layer is one of the constituent parts of CNN architecture that serves to accelerate the convergence process when training occurs and can eliminate vanishing gradients[7]. By using ReLu as an activation function, the negative pixel output will be automatically replaced with a zero value and the actual pixel value will be left. Mathematically can be written as follows:

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

By implementing the above formula into the output of the convolutional layer, a new image will be formed.



Fig. 6. Visualization of output activation layer.

The output image of the activation above is dominated by black color which indicates the majority pixel is zero. This certainly will further clarify the features and facilitate the computation of training or testing data. The final step in the feature extraction process is by implementing the pooling layer.

- Pooling Layer

The pooling layer is the last layer in the feature extraction process that functions to find the best features of the output activation layer. It is intended to clarify features and speed up computing time. The trick is to do down-sampling of each pixel in the image by implementing the formula:

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

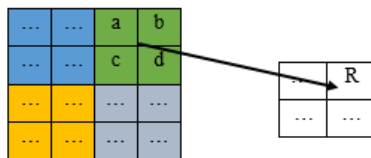


Fig. 7. Pooling layer.

By applying formula 4 to the output image of the activation layer, a new image that has been done downsampling will be obtained as visualized in Figure 8.

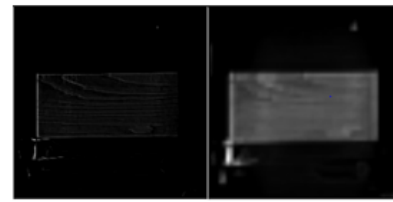


Fig. 8. Output of pooling layer.

Figure 8 is a visualization of the downsampling process, which can clarify features. cause pixel shrinkage and speed up the computing process in the classification process that occurs in the fully connected layer.

- Fully Connected Layer

Fully Connected Layer is an MLP that uses the softmax activation function. This layer is responsible for taking the extracted feature and then processing it to get the final probability. The way this layer can do the classification is to take the highest value of flattened results that have been multiplied by the weight. Weight values will continue to be updated until the lowest losses are obtained by taking into account the following basic formula:

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

The highest value for classification probability is one that refers to the lowest loss value that is zero and the term weight here is a characteristic of each object to be classified. Weight will stop being updated when losses are zero with the following calculation :

$$\text{Loss Function} = (Y - \bar{Y})^2 \quad (6)$$

As a record, in making the final prediction, the multiclass system requires the Softmax Classifier Function to be able to categorize each class.

- Softmax Classifier

Softmax classifier is one of the activation functions used in MLP with a range between 0 and 1 and the sum of all probabilities will be equal to 1. This is certainly very important in evaluating class predictions on a machine learning model. The following formula is the formulation used so that each multi-classes probability has an amount equal to 1.

$$f_j (y) = \frac{e^{y_i}}{\sum_k e^{y_k}} \quad (7)$$

The f_j notation shows the result of the function for each j^{th} element in the class output vector. The argument y is the hypothesis given by the training model so that it can be classified by the Softmax function.

C. Indicator System

The indicator used in the form of LED lights that will light based on signals from the server. Predicted signals will be sent to Arduino which has been integrated with relays and LED lights using serial communication.

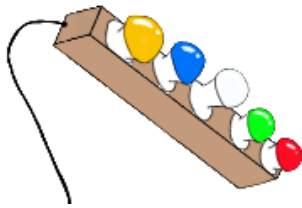


Fig. 9. Five color indicator lights.

Signals that are sent in alphabetical code based on the class of each wood with the provisions: Code "A" for red light, Code "B" for yellow light, code "C" for white light, code "D" for green light and code "E" For the blue light.

III. HASIL PERCOBAAN DAN ANALISIS

The algorithm that is used in the cedar wood classification system uses a convolutional neural network with python as the programming language and the OpenCV library. Total cedar wood used as testing material, as many as 50 boards were tested in two sessions namely from the front view and back view so that the total amount of 100 tests. The dimensions of the wood are 18.3 cm x 6.2 cm x 0.45 cm for classes A, B, and D and 18.3 cm x 7.6 cm x 0.45 cm for classes C and E. The cedar wood dataset used for the training process totaling 2830 resulting from data augmentation with a 180 degree rotate condition and flipping under the conveyor condition. The following is the overall view of the system.

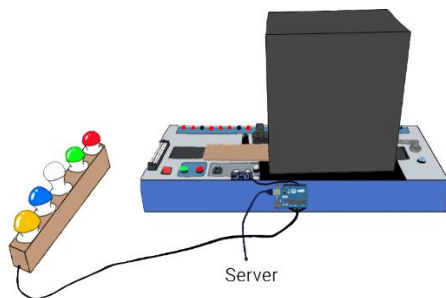


Fig. 10. Integrated system with plant.

To find out the performance of the wood classification system that is made, there are 4 important points that need to be tested namely the test of data distribution variation, CNN training parameter test which includes observations on learning rate, batch size, number of epochs, optimizer, environmental parameter test, and integrated system testing. The first test aims to find out how well the accuracy, precision, and recall of the system when viewed from the distribution of data used. The test was carried out 20 times consisting of 10 times the front view test and 10 times the back view test using wood from randomly selected datasets based on each class with the following results.

TABLE II. CONFUSION MATRIX DISTRIBUSI DATA 90:10

| Actual | Predict | | | | |
|--------|---------|----|----|----|----|
| | Class | A | B | C | D |
| A | 18 | 2 | 0 | 0 | 0 |
| B | 0 | 20 | 0 | 0 | 0 |
| C | 0 | 0 | 19 | 0 | 1 |
| D | 0 | 0 | 0 | 20 | 0 |
| E | 0 | 0 | 0 | 0 | 20 |

TABLE III. CONFUSION MATRIX DISTRIBUSI DATA 80:20

| Actual | Predict | | | | |
|--------|---------|----|----|----|----|
| | Class | A | B | C | D |
| A | 19 | 1 | 0 | 0 | 0 |
| B | 1 | 19 | 0 | 0 | 0 |
| C | 0 | 0 | 19 | 0 | 1 |
| D | 0 | 0 | | 20 | 0 |
| E | 0 | 0 | 0 | 0 | 20 |

TABLE IV. CONFUSION MATRIX DISTRIBUSI DATA 70:30

| Actual | Predict | | | | |
|--------|---------|----|----|----|----|
| | Class | A | B | C | D |
| A | 19 | 1 | 0 | 0 | 0 |
| B | 2 | 18 | 0 | 0 | 0 |
| C | 0 | 0 | 19 | 0 | 1 |
| D | 2 | 0 | 0 | 18 | 0 |
| E | 0 | 0 | 0 | 0 | 20 |

TABLE V. CONFUSION MATRIX DISTRIBUSI DATA 70:30

| Actual | Predict | | | | |
|--------|---------|----|----|----|----|
| | Class | A | B | C | D |
| A | 19 | 1 | 0 | 0 | 0 |
| B | 4 | 16 | 0 | 0 | 0 |
| C | 0 | 0 | 18 | 0 | 2 |
| D | 4 | 0 | 0 | 16 | 0 |
| E | 0 | 0 | 0 | 0 | 20 |

Based on tables II, III, IV and V we can find out the percentage of the level of accuracy, precision, and recall of the system using the following formula:

$$\text{Accuracy} = \frac{\sum_{i=1}^l \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i}}{l} * 100\% \tag{8}$$

$$\text{Precision} = \frac{\sum_{i=1}^l TP_i}{\sum_{i=1}^l (FP_i + TP_i)} * 100\% \tag{9}$$

$$\text{Recall} = \frac{\sum_{i=1}^l TP_i}{\sum_{i=1}^l (TP_i + FN_i)} * 100\% \tag{10}$$

From the formulas 8, 9, 10 obtained the level of accuracy, precision and recall as in table VI.

TABLE VI. OBSERVATION OF DATA DISTRIBUTION'S EFFECT

| Data Distribution | Parameters | | |
|-------------------|------------|-----------|--------|
| | Accuracy | Precision | Recall |
| 60:40 | 89% | 91,07% | 89% |
| 70:30 | 94% | 94,51% | 94% |
| 80:20 | 97% | 97,04% | 97% |
| 90:10 | 97% | 97,22% | 97% |

Table VI provides information that the best percentage is in the 90:10 data distribution (2547 training data and 283 testing data that the system has never seen). This shows that the increasing percentage of training data will cause the system to generalize more object characteristics of each class [2]. So that when testing is done, the system optimistically makes predictions.

The next testing is observing the learning rate as one of the CNN training parameters by using the best results from observing the distribution of data in table 6. This is intended to make observations on the acceleration of the weight correction that makes the system converge at the global point minimum so that the system can achieve optimal performance. These conditions are met when the system can achieve the greatest accuracy, precision, and recall with learning rate variations between 0.000001, 0.00001, 0.0001, and 0.001. Test results are presented in Table 7.

TABLE VII. OBSERVATION EFFECT OF LEARNING RATE

| Learning Rate | Parameters | | |
|---------------|------------|-----------|--------|
| | Accuracy | Precision | Recall |
| 0.001 | 40% | 16% | 40% |
| 0.0001 | 97% | 97,22% | 97% |
| 0.00001 | 78% | 85,68% | 78% |
| 0.000001 | 51% | 54,38% | 51% |

Based on Table VII, the most optimal learning rate for the system is 0.0001. learning rate with a value of 0.001 is considered to be too large which causes the system to diverge or in other words, the system cannot reach the global area of minima due to the drastic increase in weight, many errors during the reconstruction process so that training accuracy does not increase [8]. As with the learning rate 0.000001. the system considers the value to be too small, as a result, the step of updating the weight of each iteration is insignificant which causes the length of step to reach the global minimum area so that the accuracy obtained is not optimal.

The second test of the CNN training parameters, namely batch size by taking into account the best learning rate in the tests shown in table 7. The goal is to find the best model for generalizing features and accelerating the training process for wooden objects. The observed batch sizes are 8, 16, 24, and 32 with the experimental results presented in Table VIII.

TABLE VIII. OBSERVATION EFFECT OF BATCH SIZE

| Batch Size | Parameters | | |
|------------|------------|-----------|--------|
| | Accuracy | Precision | Recall |
| 8 | 80% | 87,14% | 80% |
| 16 | 94% | 95,38% | 94% |
| 24 | 86% | 89,40% | 86% |
| 32 | 97% | 97,22% | 97% |

Table VIII explains that the most optimal batch size with a learning rate of 0.0001 is 32. Basically a batch size that is too large will actually cause accuracy to fall because it can cause errors when updating the weight per batch [9]. But if the batch size is too small, it will make the long training time plus the system work that gets heavier. So based on research conducted by Samuel L. Smith, Pieter-Jan Kindermants, Chris Ying & Quoc V. Le, entitled "Don't Decay the Learning Rate, Increase the Batch Size", it is necessary to adjust the learning rate parameters with the batch size used to obtain accuracy and optimal training time [10].

The third test of CNN training parameters is the number of epochs by taking into account the best parameters from tables

6, 7, 8. This test is intended to find the most optimal performance, which is a system that is not classified as underfitting or overfitting. The number of epochs that are used are 1, 5, 10, and 15 with the performance shown in table IX.

TABLE IX. OBSERVATION EFFECT OF EPOCH

| Epoch | Test Parameters | | |
|-------|-----------------|---------|--------|
| | Akurasi | Presisi | Recall |
| 1 | 75% | 75,50% | 75% |
| 5 | 87% | 89,61% | 87% |
| 10 | 97% | 97,22% | 97% |
| 15 | 72% | 79,86% | 72% |

The most optimal performance in table IX is training with 10 epochs. One epoch is a training cycle for all datasets. Too little epoch will cause the model we are training to experience underfitting [11], so the accuracy obtained tends not to be too good, but conversely, if it is too many epochs it can cause the training model to be overfitting, as a result, the system is less able to generalize the characteristics of the first new timber system sight.

The fourth parameter to the CNN training parameter is the optimizer by paying attention to the 3 test parameters performed previously. Optimizer itself is closely related to how the system can reduce the value of losses and determine efficient steps for achieving the global minimum point. The optimizer algorithm that is observed is SGD, adagrad, rmsprop, and adam with the test results illustrated by table X.

TABLE X. OBSERVATION EFFECT OF OPTIMIZER

| Optimizer | Parameter | | |
|-----------|-----------|-----------|--------|
| | Accuracy | Precision | Recall |
| Adam | 97% | 97,22% | 97% |
| Rmsprop | 60% | 40% | 60% |
| SGD | 79% | 81,63% | 79% |
| Adagrad | 83% | 86,58% | 83% |

The most optimal performance is by using Adam Optimizer. Adam is the latest optimizer with the principle of combining SGD with momentum and coupled with rmsprop so that it can adapt the learning rate for each weight in the system network [12].

After the CNN training parameters have been observed, then the next step we use the best results of previous tests to review the system performance of the system work environment. Because the environment is one of the factors that affect system performance based on visual sensors. This time the test was carried out by adjusting the light intensity in the image acquisition box sourced from the LED strip mounted on the ceiling of the box. This test aims to find out how well the system performs in predicting the quality of wood analyzed in various lighting conditions. Ranging from lighting for routine work such as administrative space to fine work such as textiles [13]. Light intensity was measured using an Android application with 332 lux, 519 lux, 723 lux, and 1024 lux respectively. The results of the observations are presented in Table XII.

TABLE XI. OBSERVATION OF LIGHT INTENSITY EFFECT

| Light Intensity | Test Parameter | | |
|-----------------|----------------|-----------|--------|
| | Accuracy | Precision | Recall |
| 332 Lux | 83% | 85,67% | 83% |
| 519 Lux | 84% | 86,08% | 84% |
| 723 Lux | 97% | 97,22% | 97% |
| 1024 Lux | 91% | 92,41% | 91% |

Table XI provides information that is an intensity of 723 lux is the best environmental condition for the system when viewed in terms of lighting. The intensity that is too low can result in the loss of some fiber pattern features, while overexposure can also result in reflections of the result that the identified features are sometimes incorrect.

The final test in this paper is an integrated system test, involving the indicator system in it. Figure 9 which consists of five light units with different color variations. This test aims to determine the suitability of the color indication with the prediction code for each class sent by the system and can find out the average time prediction of the wood quality classification system. The following is a graph of the results of the experiment.

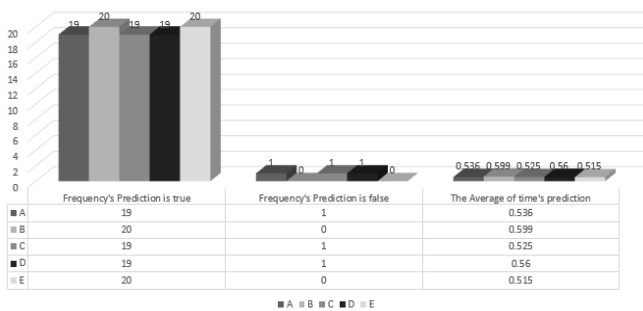


Fig. 11. Integrated system performance

Figure 11 explains each wood class based on the color of the lights of each indicator. If the calculation is done, tables 1, 2, 3, 4, 5 have an average time in a row of 0.5472 seconds with the fastest prediction time being in class E.

IV. CONCLUSION

The results of this study indicate that the cedar wood classification system using the convolutional neural network method successfully identified five wood classes based on the characteristics described in table 1 with the best performance achieving an accuracy of 97% and an average prediction speed of 0.5472 seconds. The value is obtained from the observation of comparison of training and testing data distribution of 90:10, CNN training parameters which include learning rate of 0.0001, batch size of 32, the epoch of 10, adaptive moment estimation as optimizer and light intensity of 723 lux. This shows that

CNN is a very efficient method of classifying images. With the support of automatic feature extraction on several main layers, there is an increase in performance from similar studies using HOG features and SVM classification.

REFERENCES

- [1] A. Ahmadi, "Kayu Cedar – Kayu yang Memiliki Kandungan Resin" 17 April 2018. [Online]. Available: <https://asyraafahmadi.com/en/knowledge/material-knowledge/alami/non-tambang/kayu/mengenal-jenis-kayu-keras-dan-kayu-lunak/>
- [2] Z. Nurthohari, M. A. Murti and C. Setianingsih, "Wood Quality Classification Based on Texture and Fiber Pattern Recognition using HOG Feature and SVM Classifier," 2019 IEEE International Conference on Internet of Things and Intelligence System (IoT&IS), BALI, Indonesia, 2019, pp. 123-128, doi: 10.1109/IoT&IS47347.2019.8980414.
- [3] R. Anggraeni, "Produk Kayu Ringan RI Catat Potensi Transaksi USD 5,6 Juta di Jerman" 30 Mei 2020. [Online]. Available: <https://ekbis.sindonews.com/berita/1408651/34/produk-kayu-ringan-ri-catat-potensi-transaksi-usd56-juta-di-jerman>
- [4] P. W and S. G, "Real-time Surface Grading of Profiled Wooden Boards," Joanneum Reaserch, vol. II, pp. 283-298, 1992.
- [5] ANDRIJASA, M. F. MISTIANINGSIH, Mistianingsih. Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Jumlah Pengangguran di Provinsi Kalimantan Timur Dengan Menggunakan Algoritma Pembelajaran Backpropagation. Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer, 2016, 5.1: 50-54.
- [6] Abdul, Kadir. Adh., Susanto. (2013). Teori dan Aplikasi Pengolahan Citra, Yogyakarta: Andi.
- [7] SYAIFUDDIN, Imam. DETEKSI MIKROSKOPIS SPERMATOZOA SAPI MENGGUNAKAN DEEP LEARNING CONVOLUTION NEURAL NETWORK. In: Prosiding SENTRA (Seminar Teknologi dan Rekayasa). 2019. p. 69-76.
- [8] Hamed, Mohamed & El Desouky, A.. (1996). Effect of Learning Rate on the Recognition of Images. Active and Passive Electronic Components. 19. 10.1155/1996/45086.
- [9] Hoffer, E., Hubara, I., & Soudry, D. (2017). Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In Advances in Neural Information Processing Systems (pp. 1731-1741).
- [10] Smith, Samuel L., et al. "Don't decay the learning rate, increase the batch size." arXiv preprint arXiv:1711.00489 (2017).
- [11] Satria Wibawa, Made. (2017). Pengaruh Fungsi Aktivasi, Optimisasi dan Jumlah Epoch Terhadap Performa Jaringan Saraf Tiruan. 10.13140/RG.2.2.21139.94241.
- [12] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).
- [13] Kepmenkes RI. Persyaratan Kesehatan Lingkungan Kerja Perkantoran dan Industri. Jakarta : Kemenkes RI; 2002