Comparison of K-Nearest Neighbors and Convolutional Neural Network Algorithms in Potato Leaf Disease Classification

Trisya Nurmayanti[1], Dina Hartini[2], Tatang Rohana[3], Santi Arum Puspita Lestari[4], Deden Wahiddin[5]
[12345]Universitas Buana Perjuangan Karawang
JL.HS.Ronggo Waluyo,Telukjambe,Karawang,West Java 41361
E-mail: if21.trisyanurmayanti@mhs.ubpkarawang.ac.id, if21.dinahartini@mhs.ubpkarawang.ac.id[2], tatang.rohana@ubpkarawang.ac.id[3], santi.arum@ubpkarawang.ac.id[4], deden.wahiddin@ubpkarawang.ac.id[5]

ABSTRACT

Potato production in Central Java was recorded to have decreased by 10.77% by the Central Statistics Agency (BPS), from 278,717 tons in 2022 to 248,700 tons in 2023. This decline is due to the fact that potatoes are susceptible to diseases such as late blight and dry spot (early blight) which can significantly reduce yields. This study aims to evaluate the performance of Convolutional Neural Network (CNN) with VGG16 architecture and K-Nearest Neighbors (KNN) to find the best method for potato late blight classification. The dataset used consists of 1500 potato leaf images divided into training, validation, and testing. This research uses pre-processing including resizing, rescaling, and data augmentation. The results show that CNN with the VGG16 model is superior in classifying potato leaf diseases compared to KNN with the MobileNetV2 model. CNN produced an accuracy of 96% while KNN with the MobileNetV2 model obtained an accuracy of 93%. These results can be used as a powerful tool in supporting potato leaf disease identification. This model makes a significant contribution to the development of disease identification techniques through digital image processing.

Keywords: Potato Leaf Disease, Convolutional Neural Network, VGG16, K-Nearest Neighbors, mobileNetV2.

INTRODUCTION

Potato cultivation in Indonesia is not free from plant disease problems that threaten productivity. The highland areas of Dieng (Central Java) and Kerinci (Jambi), which are the main centers of potato production in Indonesia, are not immune to this problem.[1]. Based on data from the Central Statistics Agency (BPS), potato production in Central Java experienced a significant decline of 10.77%, from 278,717 tons in 2022 to 248,700 tons in 2023. [2]. Farmers often find it difficult to identify diseases in potato crops due to the wide variety of diseases. Potato leaves are the main source of information in identifying diseases, as most disease symptoms can be seen on the leaves. Common diseases on potato leaves are late blight and dry spot (early blight).[3],[4].

Late blight in potatoes is caused by the pathogen Phytophthora infestans (Mont.) de bary, which is often carried by potato seedlings imported from abroad. [5]. Early symptoms of late blight were found in potato growing centers on the island of Java, resulting in poor yields. [6]. Yield loss due to this disease can reach 60-80%. Especially if the environment is favorable for pathogen development, such as in areas with ambient temperatures of 18-21°C and air humidity of more than 80%. [5]. This disease usually affects 5-6 week old potato plants and can spread to other parts such as stems, stalks, and tubers. [6]. Symptoms of the disease include patches of necrosis on the edges and tips of the leaves. It is important for farmers to prune infected leaves early to maintain optimal yields. Meanwhile, dry spot disease in potatoes is caused by the fungus alternaria solani and can cause up to 50% crop failure. Symptoms of the disease include gray to brown patches that grow in a circle around a clean center, resembling the shape of a bull's eye. [4].

Saputro et al's research, classification of longan plant varieties using the Convolutional Neural Network (CNN) model with Visual Geometry Group16 (VGG16) architecture produces the best accuracy of 79% and validation of 82% in 71 seconds.[7]. Research conducted by Mochammad Santosa et al compared AlexNet and ResNet34 architectures for potato leaf disease classification with 16 epochs and batch size 14. AlexNet achieved 98% accuracy in 82 minutes, while ResNet34 achieved 99% accuracy in the same time. [8]. In addition, another study used MobileNet architecture with CNN method in 4 different scenarios. The study used RMSprop optimizer, learning rate 0.0001, 50 epochs, and batch size 32 and achieved 97.90% accuracy and 0.0390 loss.[3]. Research conducted by Risma Yati et al. (2023) used CNN with 3layer convolution sized 16, 32, and 64, and Adam's optimizer to classify five types of mangoes. The research achieved the best accuracy with hyperparameters in the form of input size 100x100, 100 epochs, learning rate 0.001, and batch size 15. The results obtained were 99.56% accuracy, 100% precision, 100% recall, and 100% f1-score. [9].

Research by Utari et al conducted a classification of production results in a company using K-Nearest Neighbor (KNN). The study used 130 training data and one test data, which obtained an accuracy value of 100% with parameter K = 5.[10]. Next research by Ramadhan *et al*, [11], performed cervical cancer classification using KNN and C4.5. The results of the KNN algorithm are superior with an accuracy rate of 84% compared to C4.5. Next research by Tranose et al, using KNN for wood species classification. As a result, KNN achieved 91.6%

accuracy with parameter K = 5, while the lowest accuracy was 61.1% with parameter K = 1. The overall average accuracy is 75.54%[12]. Next research by Siti Nurjanah et al, using KNN for the classification of air pollution in the city of Jakarta, obtained an accuracy value of 95.78% with parameter K = 7.[13].

Based on the results of this study, this research aims to evaluate the performance of CNN and KNN algorithms in potato leaf disease classification. This research will measure and compare the accuracy and efficiency of both models. The results are expected to provide recommendations for a more effective model for identifying potato leaf diseases with digital image processing. So that it can help farmers in recognizing and dealing with the disease better.

METHODS

This research uses CNN and KNN algorithms with several stages which can be seen in Figure 1.

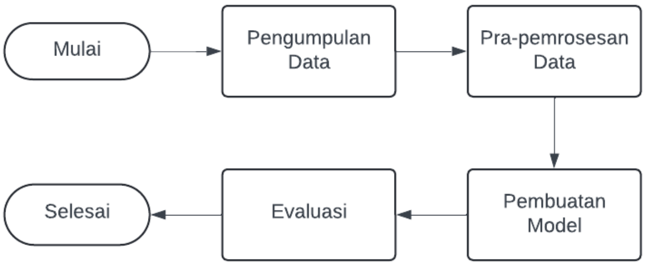

Figure 1. Research flow

Data Collection

This study uses a potato plant leaf dataset that includes three classes, namely early blight, late blight, and healthy. The dataset contains 1,500 images of potato leaves obtained from the Kaggle site. can be seen in Table 1.

Table 1. Example of Potato Leaf Dataset

No	Leaf Type	Image
1	Late Blight	
2	Early Blight	

3	Healthy	

This dataset consists of training, validation and testing data with a balanced proportion of data in each class. For training data, each class has 300 images. As for validation data, each class has 100 images. The testing data also consists of 100 images in each class. Figure 2 shows the visualization of the dataset distribution.

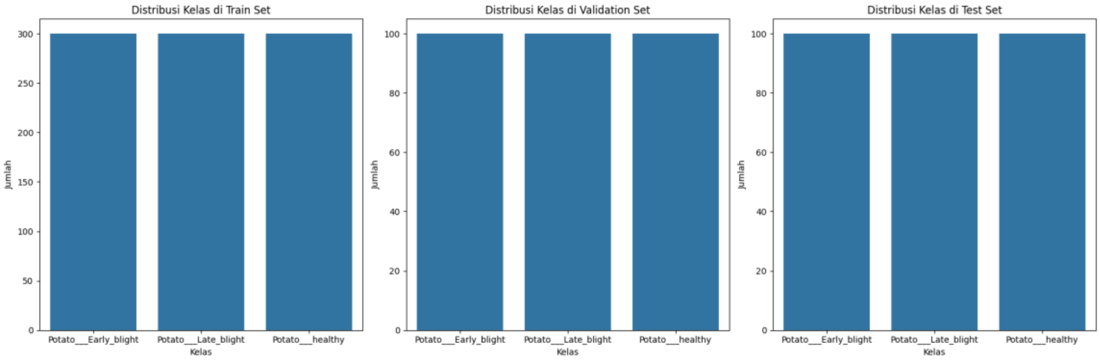

Figure 2. Dataset Distribution

Pre-processing

The pre-processing of this research involved three steps, which can be seen in Figure 3.3.

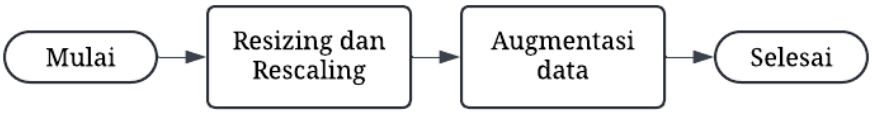

Figure 3. Pre-processing

The pre-processing stage is carried out to prepare the potato leaf image to enhance the image as a preprocessing step before analysis. The pre-processing steps carried out in this study are resizing, rescaling, and data augmentation. Resizing or resizing the image which in this study becomes 128x128 pixels. Rescaling to change the image pixel value from the original range [0, 255] to [0, 1] by dividing the pixel value by 255. Next, augmentation is performed which serves to enrich the data and allow the model to be trained from various different data points of view. [14]. Data augmentation in this study uses random flip horizontally and vertically. In addition, it performs random rotation up to 20% of the total rotation.

Model Making

Convulotional Neural Network (CNN)

CNN, also known as ConvNet, is a development of Multilayer Perceptron (MLP) designed to process two-dimensional data. CNNs process data through several layers, starting with the convolution layer, the main component that performs convolution operations by applying a function to the output of another function repeatedly. Next, the pooling layer serves to maintain the size of the data during convolution and performs downsampling, making the data more manageable and broken down into smaller parts. Finally, the fully connected layer, which is commonly used in MLP applications, aims to transform the data so that it can be classified. [10].

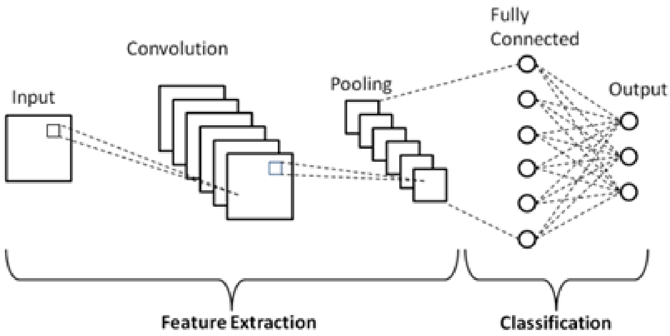

Figure 4. *Convulotional Neural Network*

Visual Geometry Group16 (VGG16)

VGG was developed by Simonyan and Zisserman. The VGG architecture consists of 16 uniform or regular convolution layers, using 3x3 convolution with multiple filters. The advantages of the VGG architecture are its ability to extract features from images in depth, as well as its homogeneous and simple topology. [15].

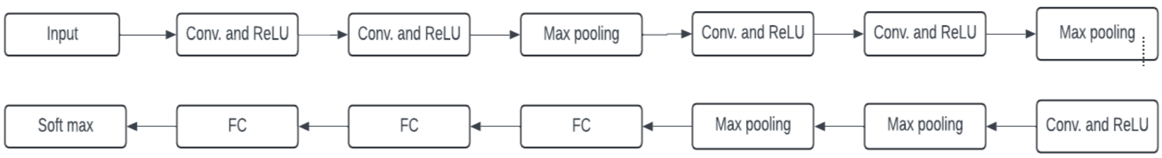

Figure 5. VGG16 Architecture

Figure 5 shows that each convolution array in VGG has a 3x3 kernel dimension. The difference lies in the number of filters in each convolution array. A filter count of 64 is used in the first two convolution arrays, while a filter count of 128 is used in the third and fourth arrays. For the subsequent convolution arrays, the number of filters used is 256 (fourth, fifth, and sixth arrays) and 512 (seventh to twelfth arrays). After the second, fourth, seventh, tenth, and thirteenth convolution arrays, 2x2 max pooling is used. The output of the last pooling will be connected to the fully connected layer, and finally connected to the classification layer to determine the class or label of the dataset.[7].

***K-Nearest Neighbors* (KNN)**

KNN is a supervised learning classification algorithm that uses the proximity distance between data to determine its class. The working principle of KNN is to find the closest distance between the evaluated data and its K closest neighbors in the training data. [16]. Calculating the distance in KNN, the Euclidean distance formula is used in the 1st equation [17].

$$D(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1}$$

Description:

D : distance

n : data dimension

i : data variable

x_i : i-th training data

y_i : i-th test data

Model Evaluation

Model evaluation is an important stage to understand how well the model can generalize to data that has never been seen before. Model evaluation is done using confusion matrix to measure how well the model can predict the class of potato leaf images in the testing data. If positive data is predicted correctly, it is called True Positive (TP). If positive data is predicted incorrectly, it is called False Negative (FN). If negative data is predicted correctly, it is called True Negative (TN). If negative data is predicted incorrectly, it is called False Positive (FP). Accuracy is the proportion of correct predictions (both positive and negative) of all data, can be calculated using Equation 2. Precision is the ratio of correct positive predictions to all predicted positive results, which can be calculated using Equation 3. Recall is the ratio of correct positive predictions to all actual positive data, calculated using Equation 4. F1-score is the harmonic mean of precision and recall, which can be calculated using Equation 5.[18].

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \times 100\% \tag{2}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{3}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{4}$$

$$F1\text{-}score = \frac{(2 \times Precision \times Recall)}{(Precision + Recall)} \tag{5}$$

RESULTS

Testing Results Convolutional Neural Network (CNN)

The CNN model building process uses the VGG16 base model from TensorFlow. This model was chosen due to its architecture that provided pre-training weights from the ImageNet dataset that could speed up convergence and improve accuracy. The model is set up with an input image, discards the last layer, and utilizes the weights from the ImageNet dataset. The weights of the VGG16 model cannot change during training to maintain consistency and ensure only additional layers will learn. Then, a new classification model is formed using the VGG16 model as the initial layer, followed by several additional layers for classification. These layers include a flatten layer to flatten the output, a dense layer with 512 units and ReLU activation, a dropout layer with a level of 0.5 to prevent overfitting, and a final dense layer with units corresponding to the number of classes and softmax activation to generate prediction probabilities. After model building, the next step is to compile the model with the Adam optimizer, due to its good ability to handle gradient descent on complex models. The loss function used was Sparse Categorical Cross-entropy, suitable for multi-class classification, and an accuracy metric was chosen to monitor performance during training. The model was trained using training data with 10 epochs and batch size 16, then validated using validation data. The results of the Confusion Matrix for CNN with VGG16 architecture can be seen in Figure 6.

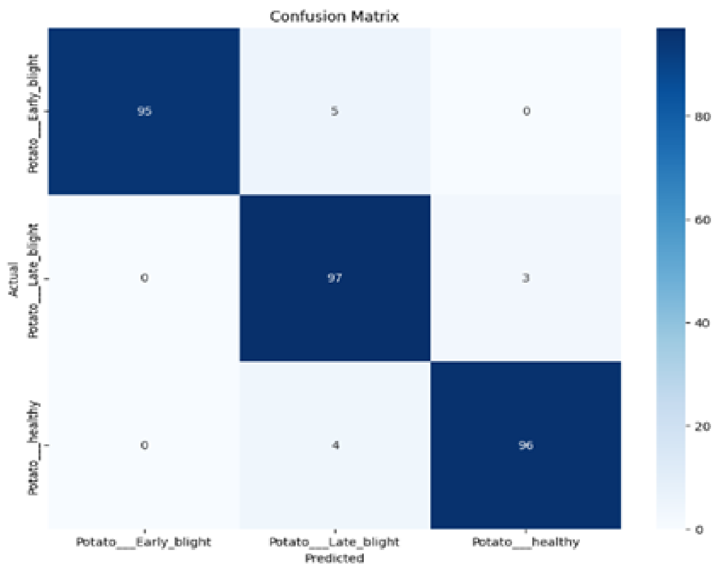

Figure 6. Confussion Matrix CNN

Figure 6 shows that CNN successfully classified 288 images out of 300 images. However, there are still prediction errors in each class of 12 images. Based on the above results, we can then calculate evaluation metrics such as precision, recall, and F1-score. The results can be seen in Table 2.

Table 2. Table Confusion matrix CNN

Class	Precision	Recall	f1-score	Accuracy
Early blight	1.00	0.95	0.97	96%
Late blight	0.92	0.97	0.94	
Healthy	0.97	0.96	0.96	

Table 2 shows that the VGG16 model achieved an accuracy of 96%, indicating its ability to generalize well to previously unseen test data. For the early blight class, the model achieved a perfect precision of 1.00, indicating no errors in predicting positive examples. Recall of 0.95 indicates that the model was able to correctly identify 95% of all Early Blight cases. The F1-score of 0.97 illustrates the strong balance between precision and recall. In the Late Blight class, although the precision was slightly lower at 0.92, the high recall of 0.97 shows that the model was able to capture most of this class well. The F1-score of 0.94 in this class reflects the model's performance in dealing with image complexity.

For the healthy class, the model shows a precision of 0.97, meaning there are only a few errors in predicting positive examples. A recall of 0.96 indicates the model's ability to find almost all healthy examples in the test data. An F1-score of 0.96 demonstrates the model's consistency in accuracy and completeness of predictions. This result confirms that the VGG16 model performs very well in classifying potato leaf images, even in situations with subtle differences between classes. The high values of precision, recall, and f1-score across all classes reflect the model's ability to capture the important characteristics of each class with low error, while also maintaining high generalization on previously unseen data.

Test Result K-Nearest Neighbors (KNN)

KNN training using the MobileNetV2 model loaded with pre-trained weights from the ImageNet dataset. MobileNetV2 was chosen for its computational efficiency and its ability to capture important features in images with fewer parameters compared to other architectures. Feature extraction is performed before the MobileNetV2 model process. These features are collected and organized into an array along with the labels and original images from each batch. After extraction, the image features undergo flattening to convert them into a two-dimensional array suitable for training the KNN model using K=3 to find a balance between the model's bias and variance. The results of the confusion matrix for the KNN algorithm can be seen in Figure 7.

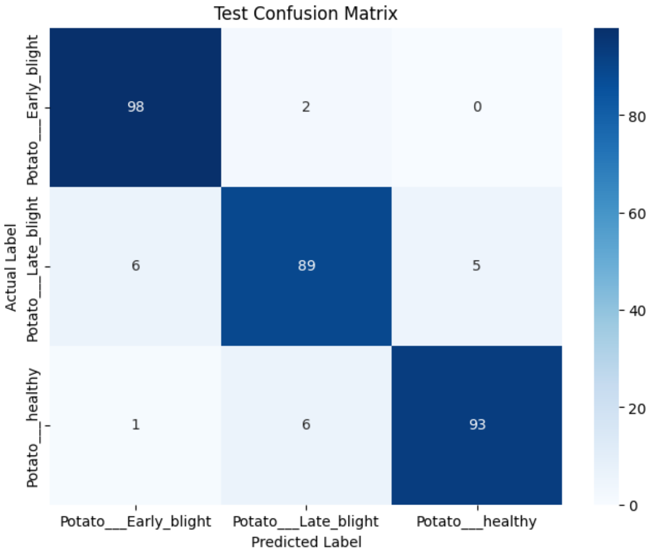

Figure 7. Confussion Matrix KNN

In Figure 7, it shows that the KNN algorithm successfully classified 280 images correctly out of 300 images. However, there are still prediction errors in each class amounting to 20 images. Based on the results above, we can then calculate evaluation metrics such as precision, recall, and F1-score. The results can be seen in Table 3.

Table 3. Table Confusion Matrix KNN

Class	Precision	Recall	F1-score	Accuracy
Early blight	0.93	0.98	0.96	93%
Late blight	0.92	0.89	0.90	
Healthy	0.95	0.93	0.94	

Table 3 shows excellent performance in identifying the Early Blight class with a Precision of 0.93 and a Recall of 0.98. The high Recall value indicates that the model is able to correctly detect almost all cases of Early Blight, although there are some images that have been misclassified. A high F1-Score (0.96) indicates a good balance between Precision and Recall, with an Accuracy of 93%, signifying the model's reliability in classification for this class.

In the Late Blight class, the model shows slightly less optimal results compared to other classes, with a Precision of 0.92 and a Recall of 0.89. The achieved F1-Score is 0.90, which still reflects good performance. The lower Recall compared to Precision indicates that this model tends to miss some actual Late Blight cases. In the Healthy class, the model achieves a Precision of 0.95 and a Recall of 0.93, with an F1-Score of 0.94. This shows that the model is capable of effectively recognizing data that falls into the Healthy category, and consistently identifies the majority of each class correctly.

CONCLUSION

From the test results, it is evident that the CNN with the VGG16 architecture outperforms KNN in classifying potato leaf images. CNN has a higher accuracy of 96% compared to KNN's 93% and a more consistent performance across all evaluation metrics, especially in terms of precision and recall across all classes. CNN is more capable of extracting complex features from images, which is important in dealing with the visual variations present in this dataset. The advantage of this CNN is evident in its ability to recognize deep patterns in data, which allows the model to capture subtle differences between classes more effectively. On the other hand, although KNN performs quite well in some classes, it shows limitations in dealing with feature complexity. These limitations are evident in its less optimal ability to identify the Late Blight class. KNN tends to only consider similarities between data in the feature space, without truly detecting the deep patterns that distinguish each class. This causes KNN to struggle to distinguish between classes with subtle differences, especially when the relevant features are not immediately apparent. As a result, KNN is less effective in producing accurate predictions for classes with more complex characteristics. In classification tasks like this, where the ability to capture details and subtle differences between classes is required, CNNs with architectures like VGG16 are more recommended compared to KNN. However, the use of KNN remains relevant when considering lower computational needs and simpler applications, especially when a lighter model is required.

REFERENCES

E. C. I. S. W. Nugraheni and E. Ratnasari, "Biobakterisida Ekstrak Daun Dewandaru (Eugenia uniflora) dalam Menghambat Bakteri Layu (Ralstonia solanacearum) pada Tanaman Kentang," *LenteraBio : Berkala Ilmiah Biologi*, vol. 10, no. 3, pp. 366–374, 2021, doi: https://doi.org/10.26740/lenterabio.v10n3.p366-374.

Badan Pusat Statistik, "Produksi Tanaman Sayuran 2021-2023." Accessed: Jun. 26, 2024. [Online]. Available: https://www.bps.go.id/id/statistics-table/2/NjEjMg==/produksi-tanaman-sayuran.html

P. T. Ompusunggu, "Klasifikasi Penyakit Tanaman Pada Daun Kentang Dengan Metode Convolutional Neural Network Arsitektur Mobilenet," *Jurnal Syntax Fusion*, vol. 2, no. 9, pp. 740–751, Sep. 2022, doi: 10.54543/fusion.v2i09.217.

A. Lesmana, R. P. Fadhillah, and C. Rozikin, "Identifikasi Penyakit pada Citra Daun Kentang Menggunakan Convolutional Neural Network (CNN)," *Jurnal Sains dan Informatika (JSI)*, vol. 8, no. 1, pp. 21–30, 2022, doi: https://doi.org/10.34128/jsi.v8i1.377.

Y. Muliani, R. Robana, and I. Mulyati, "Aplikasi Agensia Hayati Trichoderma harzianum Rifai. untuk Menekan Phytophtora infestans (Mont.) Penyebab Penyakit Busuk Daun pada Tanaman Kentang (Solanum tuberosum L.)," *AGROSCRIPT: Journal of Applied Agricultural Sciences*, vol. 5, no. 1, pp. 52–64, May 2023, doi: 10.36423/agroscript.v5i1.1245.

S. Lestari and K. I. Nauval, "Implementasi Deteksi Objek Penyakit Daun Kentang Dengan Metode Convolutional Neutral Network," *Jurnal Aplikasi Teknologi Informasi dan Manajemen (JATIM)*, vol. 3, no. 2, pp. 136–149, 2022, doi: https://doi.org/10.31102/jatim.v3i2.1576.

A. Saputro, S. Mu'min, M. Lutfi, and H. Putri, "Deep Transfer Learning Dengan Model Arsitektur VGG16 Untuk Klasifikasi Jenis Varietas Tanaman Lengkeng Berdasarkan Citra Daun," *Jurnal Mahasiswa Teknik Informatika*, vol. 6, no. 2, pp. 609–614, 2022, doi: https://doi.org/10.36040/jati.v6i2.5456.

M. K. Santosa, M. H. P. Swari, and A. N. Sihananto, "Implementasi Arsitektur Alexnet Dan Resnet34 Pada Klasifikasi Citra Penyakit Daun Kentang Menggunkan Transfer Learning," *Jurnal Mahasiswa Teknik Informatika*, vol. 7, no. 5, pp. 3293–3301, 2023, doi: https://doi.org/10.36040/jati.v7i5.7337.

R. Yati, T. Rohana, and A. Rizky Pratama, "Klasifikasi Jenis Mangga Menggunakan Algoritma Convolutional Neural Network," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 7, no. 3, pp. 1265–1275, 2023, doi: 10.30865/mib.v7i3.6445.

F. D. Utari, A. M. Siregar, and D. Wahiddin, "Implementasi Algoritme K-Nearest Neighboar (KNN) untuk Prediksi Hasil Produksi," *Scientific Student Journal for Information, Technology and Science*, vol. 1, no. 1, pp. 21–25, 2020, Accessed: Jun. 23, 2024. [Online]. Available: https://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/index

M. R. Ramadhan, Y. Cahyana, and A. R. Juwita, "Penerapan Algoritma K-Nearest Neighbor dan C4.5 untuk Klasifikasi Penyakit Kanker Serviks," *Scientific Student Journal for Information, Technology and Science*, vol. 5, no. 1, pp. 86–95, 2024, Accessed: Jun. 23, 2024. [Online]. Available: https://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/article/view/989

B. M. W. Tranose, H. H. Handayani, and S. A. P. Lestari, "Penerapan Algoritma K-Nearest Neighbor Dengan Pengolahan Citra Digital Untuk Mengidentifikasi Jenis Kayu," *Scientific Student Journal for Information, Technology and Science*, vol. 4, no. 2, pp. 102–110, 2023, Accessed: Jun. 23, 2024. [Online]. Available: https://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/article/view/973

S. Nurjanah, A. M. Siregar, and D. S. Kusumaningrum, "Penerapan Algoritma K – Nearest Neighbor (Knn) Untuk Klasifikasi Pencemaran Udara Di Kota Jakarta," *Scientific Student Journal for Information, Technology and Science*, vol. 1, no. 2, pp. 71–76, Jul. 2020, Accessed: Jul. 03, 2024. [Online]. Available: http://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/issue/view/53

K. L. Kohsasih, M. D. A. Rizky, T. Fahriyani, V. Wijaya, and R. Rosnelly, "Analisis Perbandingan Algoritma Convolutional Neural Network dan Algoritma Multi-layer Perceptron Neural dalam Klasifikasi Citra Sampah," *Times*, vol. 10, no. 2, pp. 22–28, 2021, [Online]. Available: http://ejournal.stmik-time.ac.id

J. R. Aisya and A. Prasetiadi, "Klasifikasi Penyakit Daun Kentang dengan Metode CNN dan RNN," *Jurnal Tekno Insentif*, vol. 17, no. 1, pp. 1–10, Apr. 2023, doi: 10.36787/jti.v17i1.888.

Moch. R. Yuliansyah, M. B, and A. Franz, "Perbandingan Metode K-Nearest Neighbors dan Naïve Bayes Classifier Pada Klasifikasi Status Gizi Balita di Puskesmas Muara Jawa Kota Samarinda," *Adopsi Teknologi dan Sistem Informasi (ATASI)*, vol. 1, no. 1, pp. 08–20, Jun. 2022, doi: 10.30872/atasi.v1i1.25.

H. H. Sutarno, R. Latuconsina, and A. Dinimaharawati, "Prediksi Stunting Pada Balita Dengan Menggunakan Algoritma Klasifikasi K-Nearest Neighbors," *eProceedings of Engineering*, vol. 8, no. 5, pp. 6657–6661, 2021, Accessed: Jun. 26, 2024. [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/16456

O. V. Putra, M. Z. Mustaqim, and D. Muriatmoko, "Transfer Learning untuk Klasifikasi Penyakit dan Hama Padi Menggunakan MobileNetV2," *Techno.Com*, vol. 22, no. 3, pp. 562–575, Aug. 2023, doi: 10.33633/tc.v22i3.8516.