Indonesian Journal of Applied Technology, Computer and Science
Vol. 2, No. 1, Feb 2025, pp. 15-24
ISSN: 3063-8305

Classification of Defects in Arabica Coffee Beans Using the KNN (K-Nearest Neighbor) Method in the Gayo Farming Cooperative, Bebeseb District, Central Aceh

[1]Khairunnisa, [2]Martiano
Department of Information Systems, Universitas Muhammadiyah Sumatera Utara, Indonesia

ABSTRACT

This study examines the classification of defects in Arabica coffee beans using the K-Nearest Neighbor (KNN) method at the Koperasi Usaha Tani Gayo, Aceh Tengah. Arabica coffee, with its high economic value, is often subjectively evaluated by farmers using their senses, which is less effective. Therefore, this research employs the KNN method to enhance the accuracy of defect classification in coffee beans. The KNN method, a supervised algorithm, classifies objects based on the categories of their nearest neighbors. This study utilizes color digital images processed with web-tools Teachable Machine and the MNIST dataset. The dataset is divided into three parts: training, validation, and testing. Defect images of coffee beans are classified into 16 classes, such as Full Sour Bean, Full Black Bean, and others. Evaluation shows that the KNN model has high accuracy in classifying coffee bean defects, although it requires significant computational time. The research results are implemented in a mobile application based on Flutter and the Dart programming language, simplifying the process of classifying Arabica coffee bean defects at the Koperasi Tani Gayo, enhancing the quality and efficiency of coffee bean determination.

Keyword : Defect classification; Arabica coffee beans; K-Nearest Neighbor (KNN); digital image; Teachable Machine; MNIST dataset.

Corresponding Author:
Khairunnisa
Department of Information Systems
Universitas Muhammadiyah Sumatera Utara
Jl. Kapten Mukhtar Basri No 3 Medan, 20238, Indonesia.
Email : knisa4806@gmail.com

Article history:
Received April, 2024
Revised April, 2024
Accepted Mei, 2024

1. INTRODUCTION

Indonesia is one of the countries that has a type of coffee that is quite famous for its distinctive taste and aroma. Most of the coffee produced in Indonesia consists of Arabica coffee and Robusta coffee. Arabica coffee grows in areas that have high land, while Robusta coffee is grown in low areas. Arabica coffee is one of the most famous variants because it has a unique taste and Arabica beans are also a global coffee bean product that has a distinctive taste, so it can refresh the body (Ficri, et al.2022).

Coffee is one of the plantation commodities that is very much needed by coffee farmers and has a fairly large economic value. Coffee is very popular among the community, both locally and internationally. Coffee has an important role as a contributor of foreign exchange for the country. In addition, coffee is also the main source of income for coffee farmers in Indonesia and opens up job opportunities for the community. Central Aceh and Bener Meriah Districts are the areas with the highest coffee production in Aceh Province. The area of coffee plantations in these two districts reaches 80% or around 96 thousand hectares. The existing coffee plantations as a whole are owned by the community, involving as many as 77 thousand heads of families (Noratun, et.al.2017).

Gayo Farming Cooperative is a business entity that focuses on the process and export of coffee quality commodities, not solely determined by its type. Coffee is also determined based on the quality of the coffee (Amelia, et al. 2023). The farmers are less careful in identifying the quality of Arabica coffee beans using the five senses such as hands, eyes, aroma, touch and smell. So the method of selecting the best Arabica coffee beans is less effective.

Digital image to find a matrix where the row and column indices indicate a location in the image. The elements in this matrix (elements in an image) represent the level of brightness at the location in question. In this study, color images were used. Color images consist of three basic colors, namely Red, Green, and Blue.

With this we can use the KNN (K-Nearest Neighbor) approach which is one of the techniques of supervised learning algorithms, where the results of query instances are grouped based on the majority category of the K nearest neighbors. This algorithm is designed to group new objects based on characteristics and examples taken from training data. The classification process is carried out by determining the nearest neighbor from the new test data sample. The proximity or distance of neighbors is calculated using the Euclidean distance. (Ficri, et.al, 2022). Using the KNN method as a machine learning technique to classify coffee beans based on certain visual characteristics and features.

Assess the effectiveness and accuracy of the KNN method in distinguishing good quality coffee beans from defective ones.

The purpose of this study is to identify and categorize various types of defects in Arabica coffee beans produced at the Gayo Agricultural Cooperative, Bebeseb District, Central Aceh, determine the severity of defects to assist in the selection process and determination of coffee bean quality, assist farmer cooperatives in ensuring that only high-quality coffee beans pass the production and distribution process, reduce errors in coffee bean selection by using a technology-based approach to improve quality consistency, reduce dependence on manual methods in classifying coffee bean defects, thereby accelerating the sorting process and increasing productivity, provide artificial intelligence-based technology solutions that can be applied by farmers or cooperatives on a larger scale.

2. RESEARCH METHOD/MATERIAL AND METHOD/LETERATURE REVIEW

A. KNN (K-Nearest Neighbor) Method

The K-Nearest Neighbor (KNN) method is a technique for classifying objects based on training data that has the closest distance to the object being analyzed. Nearest neighbor is a method used to find cases by measuring the similarity between new cases and existing cases, considering similar values and weights. The goal of the KNN (K-Nearest Neighbor) method is to classify new objects using attributes and training samples.

The basic principle of K-Nearest Neighbor (KNN) is to identify the closest distance between the data being analyzed and the K nearest neighbors in the training dataset. This method is included in the nonparametric classification category. The KNN method is quite easy to understand, operating by measuring the closest distance between the searched instance and the training sample to identify its KNN. (Aziz Ali Mahendra, et,al. 2023).

K-Nearest Neighbor, better known as KNN, is part of the instance-based learning group. This method is done by grouping a set of objects from training data that have the closest distance to the new or tested data. The KNN algorithm is a method used to group certain objects by considering the training data that has the highest similarity to the object. (Taufiq, et,al. 2022).

B. Data Analysis Techniques

The author applies data analysis techniques through the KNN method with the following steps:

1) Determine the parameters with the number of nearest neighbors.
2) Calculate the square of the distance of each object to the sample data that has been provided by utilizing the formula above.
3) Then group the objects based on the closest distance.
4) Collect the KNN classification categories.
5) By utilizing the most influential K-Nearest Neighbor category, the calculated query value can be predicted.

The author uses data analysis techniques with the KNN (K-Nearest Neighbor) method. The stages in data analysis are as follows:

1. Determining Parameters (Number of Nearest Neighbors)
   KNN requires a parameter K, which is the number of nearest neighbors to be used for classification or regression. The K value is selected based on experiments and validation to obtain the best performance.
2. Calculating Euclidean Distance
   The distance between two data points X=(X_1,X_2,….,X_n) and Y=(Y_1,Y_2,….,Y_n) can be calculated using the Euclidean distance formula as follows:

$$d(X,Y) = \sqrt{(X_1 - Y_1)^2 + (X_2 - Y_2)^2 + \cdots + (X_n - Y_n)^2}$$

Where d(X,Y) is the distance between points X and Y, and X_i and Y_i are the components of the data points X and Y respectively.

3. Sorting Objects by Distance
   After calculating the distance between the sample data point and all data points in the dataset, the next step is to sort all data points based on the distance from smallest to largest.

4. Determining Classification Based on K Nearest Neighbors
   From the sorted data, take the K nearest data points. Classification is done by selecting the category that appears most among the K nearest neighbors. For example, if out of 5 nearest neighbors, 3 of them are good quality coffee beans, then the sample will be classified as good quality coffee beans.
5. Model Validation
   Validation is done to test the accuracy of the model that has been built. A commonly used technique is k-fold cross validation. This process involves dividing the dataset into k subsets, then the model is trained using k-1 subsets and tested using the remaining subsets. This process is repeated k times with each subset used once as test data. The average accuracy of all iterations is calculated to get an idea of the model's performance.
6. Confusion Matrix
   Confusion Matrix is an analysis tool used to assess the effectiveness of a classification algorithm. This matrix depicts the number of correct and incorrect predictions produced by the model when compared to the actual values in the dataset.

3. RESULTS AND DISCUSSION

A. Dataset Preparation Stage

At this stage, the data in the form of images will be trained in order to produce an appropriate model. This process uses a web-tools teachable machine, a web-based application used to create a classification model. And the dataset used is MNIST, which consists of 28x28 pixel handwritten digit images. This data set is divided into three categories: training set, validation set, and test set. Training data is used to train the model, while validation data is used to test the model during the training session. While the test data is used to assess the extent of the model's performance after training is complete.

The image to be classified is a defect image on coffee which is divided into 16 classes Full Sour Bean, Full Black Bean, Dried Cherry Pod, Fungus Damage Bean, Foreign Matter, Severe Insect Damage, Partian Black Bean, Partial Sour Bean, Parchment Pergamino Bean, Floater Bean, Immature Unripe Bean, Whitered Bean, Shell, Broken Chipped Cut, Hull Husk, Slight Insect Damage. Grouping of defects in coffee that are categorized in the form of defects. So that it can distinguish the image objects that will be produced. The following is a display of the teachable machine web tools in Figure 1.

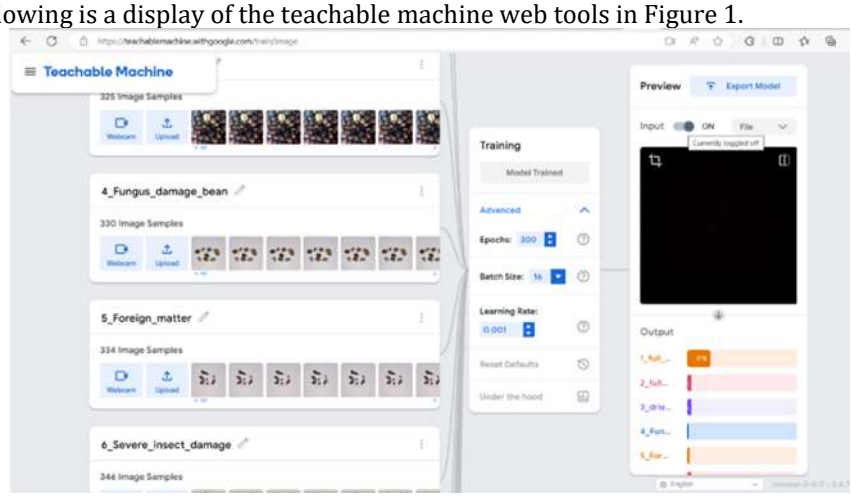

Fig 1. Teachable machine Web View

Based on Figure 1. it is a display of the results of the input dataset of defects in coffee that will be trained using the teachable machine web tools. In this process, the dataset is trained by considering the type of dataset, batch size and also the learning rate.

Dataset is a hyperparameter that determines how many times the deep learning algorithm works through the entire dataset, which means that one dataset is achieved when all batches have been passed through the neural network once. Batch size is the number of dataset samples that will be passed through at one time.

This batch size can determine the number of samples to be worked on. Learning rate is one of the training parameters for calculating the weight correction value during the training process. In this case, the learning rate is a parameter for how fast the training process will be carried out.

B. Influence of Dataset on Model

In the testing stage carried out by knowing the goodness of a neural network model created by comparing the dataset to the model so that it can be a reference material in determining the desired level of model accuracy. In this study, six different epoch categories were used, namely 50, 100, 150, 200, 250, 300 with a batch size of 16 with the same dataset. Epoch variations were carried out to find the best and ideal learning model in this case of coffee defect research. Low epochs have the potential to fail in model learning, while in higher epoch cases there is the potential for overfitting. The standard epoch in machine learning is 100 iterations, but in heavier learning, an addition of up to 500 epochs can be considered.

The results of the epoch variation will be shown in the form of training results in the form of train accuracy and train testing graphs which are commonly called validation. A good model is a model that reaches a level of convergence which is indicated by no change in the value of the accuracy parameter approaching 1. The results of the visualization of the training process with epoch variations are shown in Figure 2.

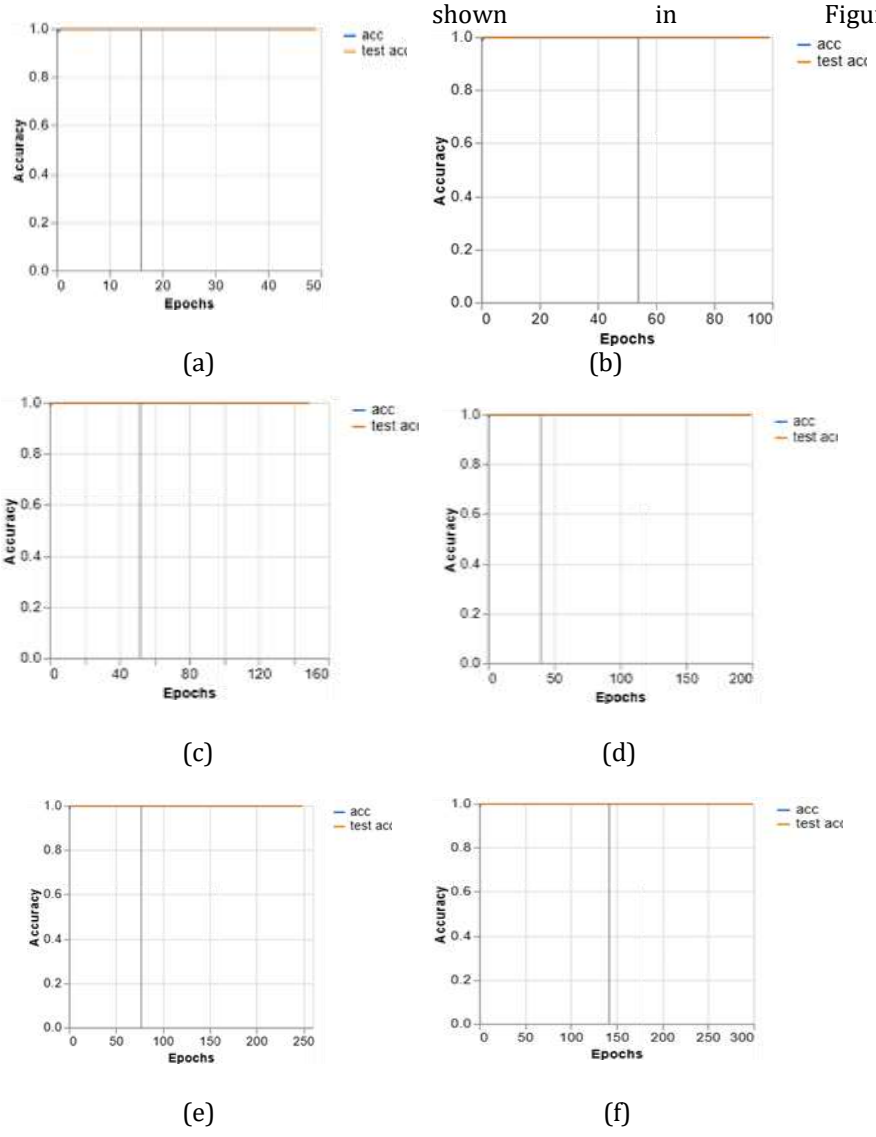

Fig 2. Training Dataset Results per Epoch, (a) epoch 50, (b) epoch 100, (c) epoch 150, (d) epoch 200, (e) epoch 250, (f) epoch 300.

At epoch 50, the model shows high training accuracy, as seen from the blue line reaching 100%. The testing accuracy also reaches the same value, which is 100%. This shows that there is no disparity between training and testing accuracy at this epoch, indicating that the model is able to generalize its knowledge well since the beginning of training. The model has successfully classified the training and testing data with perfect accuracy.

At epoch 100, a similar trend is seen where the training and testing accuracy both remain at 100%. There is no indication of overfitting because there is no difference between training and testing accuracy. This shows that the model remains stable and is able to maintain optimal performance in classifying training and testing data without any performance degradation or poor generalization.

At epoch 150, the situation remains the same with training and testing accuracy remaining at 100%. This indicates that the model is able to maintain optimal performance in classifying data in both training and testing. This consistency indicates that the model has reached a point where it can effectively learn from training data and apply its knowledge to testing data with perfect results.

At epoch 200, there is no change in training and testing accuracy, both of which remain at 100%. This indicates that the model has reached maturity in its learning process with optimal performance. At this stage, the model shows excellent generalization ability, with minimal difference between training and testing accuracy. This indicates that the model has stabilized and does not require further adjustment to improve its performance.

At epochs 250 and 300, the model still maintains training and testing accuracy at 100%. This indicates that the model has reached an optimal limit in the training process, where adding further epochs does not provide significant performance improvements. The model shows stable and accurate performance in its classification, with consistent and efficient generalization ability.

Based on the performance analysis above, the model shows optimal performance from the beginning to epoch 300, with consistent training and testing accuracy at 100%. This indicates that the model is able to generalize knowledge from training data to testing data very well without any signs of overfitting or underfitting. The model has achieved optimal efficiency in its learning process since the early epochs.

The figures from tables (a) to (f) show that from epoch 50 to 300, the training and testing accuracy are always at 100%, which means that the model consistently gives its best performance throughout training. There are no significant changes or disparities seen, indicating that the model is stable and very accurate in its classification since the beginning of training. This shows that the model has been well trained and is able to effectively deal with data complexity without losing the ability to generalize well on data that has never been seen before.

Overall, the model shows very good and stable performance throughout the training process, with the ability to classify training and testing data with perfect accuracy. This reflects the excellent quality of the algorithm and data used in training this model.

C. Analysis of Object Classification Class 1 Full Sour Bean

This study uses image classification on coffee that has defects, namely Full Sour Bean. The results of the study will be shown using a confusion matrix per object from the image classification. This study refers to the parameters of the object classification class 1 Full Sour Bean, therefore the parameters of the results of the confusion matrix are obtained from the training results on the web tools teachable machine based.

Based is the measurement result of Full Sour Bean object classification parameter to find out precision, recall and f1-score. Based on table 1, it can be seen that the average value of accuracy obtained in the Full Sour Bean part model is 97.75%

F1-score evaluation, the average value is 0.968 indicating a fairly high performance standard. The average value of precision and recall which are variables that affect the F1-score itself are 0.977 and 0.993. Based on the data Based on Table 1, the following comparison graph can be determined.

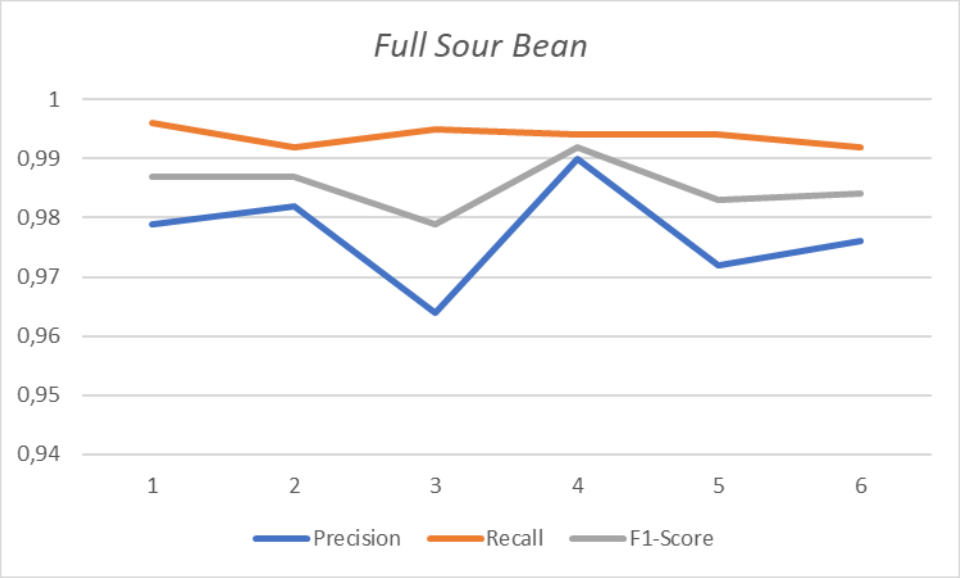

Fig 3. Model Performance on Object Classification Class 1 Full Sour Bean

Figure 3 is a performance graph showing f1-score and accuracy. Based on Figure 3, the results of the model performance, it can be concluded that the value of f1-score will be the same as accuracy. Based on the overall model obtained, it has an average accuracy of 0.9795, an average precision value of 0.977, an average recall value of 0.993, and an average f1-score value of 0.968, from the average value obtained in this study, it is concluded that the model obtained using the teachable machine web-tools has quite good learning and prediction results, and has reliability in detecting objects using an android camera, but in the process of taking the dataset there is incoming light noise but it can be used to recognize objects.

D. Visualization of Analysis Results

Epoch	*Accuracy* (%)	*Precision*	*Recall*	*F1-Score*
50	97,85	0,979	0,996	0,987
100	98,26	0,982	0,992	0,987
150	96,48	0,964	0,995	0,979
200	98,94	0,990	0,994	0,992
250	97,30	0,972	0,994	0,983
300	97,70	0,976	0,992	0,984
$\bar{X}$	**97,75**	**0,977**	**0,993**	**0,968**

Table 1. Object Classification Class 1 Full Sour Bean

Once the score map is calculated based on the average distance of the nearest neighbors for each pixel in the full image, it is visualized in several ways:

1. Displaying the original image for visual reference.
2. Displaying the score map with a color scale to indicate variations in similarity.
3. Displaying a binary score map based on a certain threshold to identify areas of high similarity.
4. Displaying the original image masked with the binary score map to highlight the relevant areas.

These visualizations provide a clear picture of the distribution of pixel similarities and help in identifying areas in the full image that have high similarity to the cropped image.

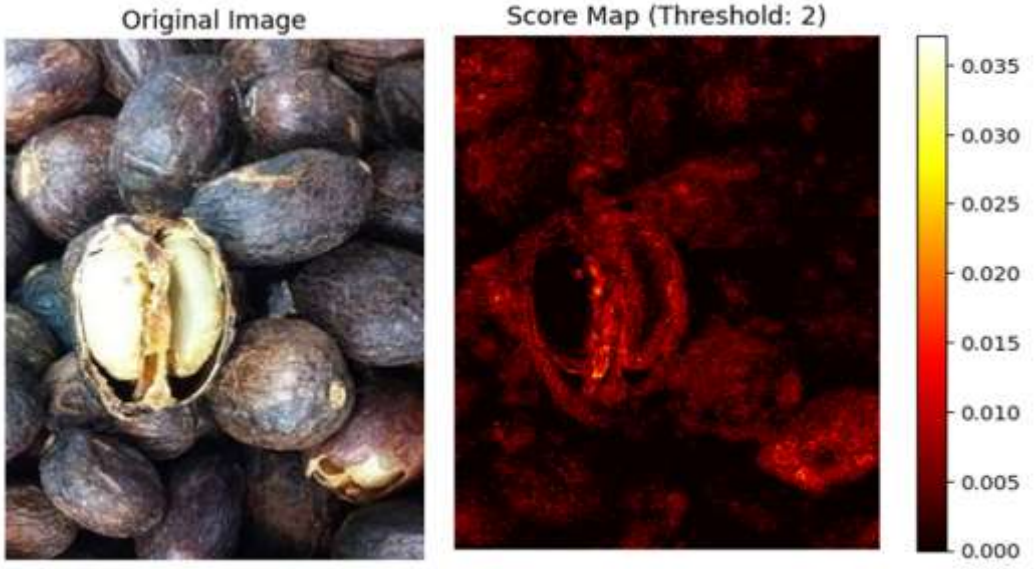

Fig 4. Image data processing using RGB

The image shown is a score map of an image based on pixel similarity with the image crop (crop1) using the K-Nearest Neighbors (KNN) model. This score map shows the intensity of pixel similarity in the original image in red and yellow. Areas with brighter colors indicate higher similarity. To see the mathematical calculations used in generating the score map with the K-Nearest Neighbors (KNN) model.

E. Displaying the Application Display

The application displayed is a mobile-based application designed using flutter as a framework and using the dart programming language to create the designed application.

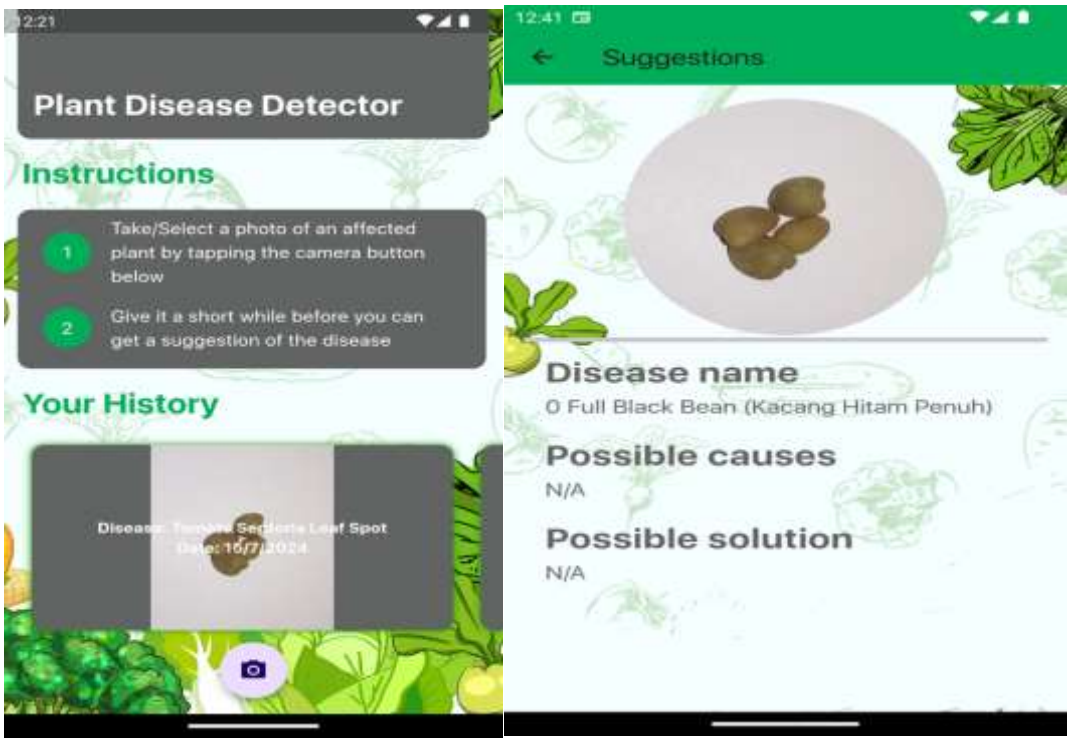

Fig 5. Coffee Disease Detector App View

The plant disease detector application on the homepage displays the title "Plant Disease Detector" at the top, with instructions for users to take or select a photo of the infected plant using the camera button below. After that, users are asked to wait a moment for the application to provide suggestions regarding the detected disease. At the bottom is a history of previously analyzed plant photos along with the diagnosis of the disease.

The second page of the application displays the title "Suggestions" at the top, with a close-up image of the infected plant. This section provides details about the name of the detected disease, such as "Full Black Bean". The application overall functions to detect diseases in plants by providing a diagnosis based on photos taken or selected by the user and displaying a history of previous diagnoses.

4. CONCLUSION

The conclusions obtained in this study based on the analysis and discussion include: This study has successfully applied the K-Nearest Neighbor (KNN) method for defect classification in Gayo Arabica coffee beans. From the results of the tests carried out, the KNN method showed optimal performance with consistent training and testing accuracy. This shows that the KNN model used is able to classify defects in coffee beans very well without any signs of overfitting or underfitting. The advantages of the KNN method applied in this study include its ability to provide accurate and efficient classification results. However, there are limitations related to the computing time required for the prediction process, especially on a large scale or in situations that require a fast response. This is an important consideration in applying this method in the field. The implementation of this defect classification system in Arabica coffee beans provides significant benefits for farmers. With this system, farmers can more easily and quickly determine the quality of their coffee beans, which in turn can increase work efficiency and the quality of the coffee products produced.

REFERENCES

Ariska, D. M., Astuti, I. F., & Agus, F. (2024). Rancangan dan Implementasi Sistem Pendukung Keputusan Pemilihan Bibit Tanaman Karet. IKRA-ITH Informatika : Jurnal Komputer Dan Informatika, 8(2), 82–91. https://doi.org/10.37817/ikraith-informatika.v8i2.2961

Chazar, C., & Rafsanjani, M. H. (2022). Penerapan Teachable Machine Pada Klasifikasi Machine Learning Untuk Identifikasi Bibit Tanaman. Prosiding Seminar Nasional Inovasi Dan Adopsi Teknologi (INOTEK), 2(1), 32–40. https://doi.org/10.35969/inotek.v2i1.207

Napoli, C. (2019). Beginning Flutter: A hands on guide to app development. Apress.

Sari, I.P., Hariani, P.P., Al-Khowarizmi, A., Ramadhani, F., Sulaiman, O.K., Satria, A, & Manurung, A.A. (2024). CLUSTERING HIV/AIDS DISEASE USING K-MEANS CLUSTERING ALGORITHM. Proceeding International Seminar on Islamic Studies 5 (1), 1668-1676

Fichri, M., Ramadhan, A., Arsyad, F., & Jumaryadi, Y. (2022). Implementasi Simple Additive Weighting Dalam Menentukan Biji Kopi Terbaik. Journal of Information System Research (JOSH), 3(3), 234–241. https://doi.org/10.47065/josh.v3i3.1505

Juliaviani, N., Sahara, S., & Winandi, R. (2017). Transmission Price of Gayo Arabica Coffee In Aceh Province. Jurnal Agribisnis Indonesia, 5(1), 39.

Sulaiman, O.K & Batubara, I.H. (2021). Implementation Data Mining For Level Analysis Traffic Violation By Algorithm Association Rule. Al'adzkiya International of Computer Science and Information Technology (AIoCSIT) Journal 2 (2), 128-135

Sari, I.P., Ramadhani, F., Satria, A., & Sulaiman, O.K. Leukocoria Identification: A 5-Fold Cross Validation CNN and Adaboost Hybrid Approach. 2023 6th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 486-491

Manurung, A.A., Nasution, M.D., & Sari, I.P. (2023). Implementation of Fuzzy K-Nearest Neighbor Method in Dengue Disease Classification. 2023 11th International Conference on Cyber and IT Service Management (CITSM), 1-4

Giordano, A. (2019). Flutter Projects: A practical, project-based guide to building real-world cross-platform mobile applications and games. Packt Publishing.

Kurosaki, K. (2020). Mastering Flutter: A step-by-step guide to designing and building Flutter apps. Independently published.

Sari, I.P., Ramadhani, F., Satria, A., & Apdilah, D. (2023). Implementasi Pengolahan Citra Digital dalam Pengenalan Wajah menggunakan Algoritma PCA dan Viola Jones. Hello World Jurnal Ilmu Komputer 2 (3), 146-157

Mahendra, A. A., Suranti, D., & Fredricka, J. (2023). Sistem Pendukung Keputusan Calon Penerima BPJS-PBI Pada Dinas Sosial Kota Bengkulu Menggunakan Metode K-Nearest Neighbor (KNN). Jurnal Media Infotama, 19(2), 391–400. https://doi.org/10.37676/jmi.v19i2.4290

Raup, A., Ridwan, W., Khoeriyah, Y., Supiana, S., & Zaqiah, Q. Y. (2022). Deep Learning dan Penerapannya dalam Pembelajaran. JIIP - Jurnal Ilmiah Ilmu Pendidikan, 5(9), 3258–3267. https://doi.org/10.54371/jiip.v5i9.805

Sari, I.P., Al-Khowarizmi, A, Sulaiman, O.K., & Apdilah, D. (2023). Implementation of Data Classification Using K-Means Algorithm in Clustering Stunting Cases. Journal of Computer Science, Information Technology and Telecommunication Engineering 4 (2), 402-412

Apdilah, D., & Sari, I.P. (2021). Optimization Of The Fuzzy C-Means Cluster Center For Credit Data Grouping Using Genetic Algorithms. Al'adzkiya International of Computer Science and Information Technology (AIoCSIT) Journal 2 (2), 156-163

Rezky Firmansyah. 2021. IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI BUNGA. Skripsi. UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA

Sari, I.P., Batubara, I.H., & Al-Khowarizmi, A. (2021). Sensitivity Of Obtaining Errors In The Combination Of Fuzzy And Neural Networks For Conducting Student Assessment On E-Learning. International Journal of Economic, Technology and Social Sciences (Injects) 2 (1), 331-338

Tolana, F., Muzakkir, I., & Riadi, A. (2023). Metode K-Nearest Neighbor (KNN) Untuk Sistem Pakar Diagnosa Penyakit Pneumonia Pada Balita. Jurnal Nasional Komputasi Dan Teknologi Informasi, 6(3), 363–373.

Varian, M. A. (2017). Analisa karakteristik kopi arabika dan metode penjemuran terhadap citarasa kopi arabika menggunakan analisa varian. 14–36.

Sari, I.P., Al-Khowarizmi, A., & Batubara, I.H. (2021). Cluster Analysis Using K-Means Algorithm and Fuzzy C-Means Clustering For Grouping Students' Abilities In Online Learning Process. Journal of Computer Science, Information Technology and Telecommunication Engineering 2 (1), 139-144

Yuhandri, Y., Ramadhanu, A., & Syahputra, H. (2022). Pengenalan Teknologi Pengolahan Citra Digital (Digital Image Processing) Untuk Santri Di Rahmatan Lil'Alamin International Islamic Boarding School. Community Development Journal : Jurnal Pengabdian Masyarakat, 3(2), 1239–1244. https://doi.org/10.31004/cdj.v3i2.5868

Biessek, M. (2019). Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart 2. Packt Publishing.

Indah Purnama Sari. Algoritma dan Pemrograman. Medan: UMSU Press, 2023, pp. 290.

Indah Purnama Sari. Buku Ajar Rekayasa Perangkat Lunak. Medan: UMSU Press, 2021, pp. 228.

Janner Simarmata Arsan Kumala Jaya, Syarifah Fitrah Ramadhani, Niel Ananto, Abdul Karim, Betrisandi, Muhammad Ilham Alhari, Cucut Susanto, Suardinata, Indah Purnama Sari, Edson Yahuda Putra. Komputer dan Masyarakat. Medan: Yayasan Kita Menulis, 2024, pp.162.

Mahdianta Pandia, Indah Purnama Sari, Alexander Wirapraja Fergie Joanda Kaunang, Syarifah Fitrah Ramadhani Stenly Richard Pungus, Sudirman, Suardinata Jimmy Herawan Moedjahedy, Elly Warni, Debby Erce Sondakh. Pengantar Bahasa Pemrograman Python. Medan : Yayasan Kita Menulis, 2024, pp.180

Zelvi Gustiana Arif Dwinanto, Indah Purnama Sari, Janner Simarmata Mahdianta Pandia, Supriadi Syam, Semmy Wellem Taju Fitrah Eka Susilawati, Asmah Akhriana, Rolly Junius Lontaan Fergie Joanda Kaunang. Perkembangan Teknologi Informatika. Medan: Yayasan Kita Menulis, 2024, pp.158