DIGIVENTURE
Journal of Applied Digital Business Research

Vol. 1, No. 2, (2025), pp. 142-167
e-ISSN: 3123-5212
Website: https://journal.unesa.ac.id/digiventure

DEVELOPMENT OF A RUPIAH BANKNOTE IDENTIFICATION MODEL IN THE CURRENCY SENSE APPLICATION FOR VISUALLY IMPAIRED

Putri Rigita Cahyani[1]

[1] Faculty of Economics and Business, Universitas Negeri Surabaya, Indonesia,
putririgita.21025@mhs.unesa.ac.id

ABSTRACT

This study develops and evaluates an Android based assistive application, *Currency Sense*, designed to help visually impaired individuals recognize Indonesian Rupiah banknote denominations independently. The application employs a Convolutional Neural Network to perform image based currency classification and delivers the recognition results through audio feedback. The research methodology includes the development of a CNN based classification model, system implementation on the Android platform, and performance evaluation using a confusion matrix. The experimental results demonstrate that the proposed model achieves an accuracy of 89.28%, with the highest recognition performance observed for the Rp50,000 and Rp100,000 denominations. Several misclassifications were identified among visually similar denominations, primarily influenced by variations in lighting conditions, image orientation, and image noise. In addition to model evaluation, black box testing was conducted to assess application functionality across four main interface pages using ten test scenarios, all of which produced valid results. These findings indicate that the application functions reliably and meets functional requirements. The proposed system contributes to artificial intelligence–based accessibility solutions by offering a practical tool to enhance autonomy in financial transactions for visually impaired users.

***Keywords**: Convolutional Neural Network, Currency Sense, Visual Impairment, Image Classification, Assistive Technology*

***Article History:** (filled by Journal Manager)*
Received : 10 November 2025
Revised : 30 November 2025
Accepted : 10 December 2025
Available online : 26 December 2025

https://doi.org/.....................

I. INTRODUCTION

Visual impairment remains a significant public health issue in Indonesia. According to data from the Indonesian Ophthalmologists Association (*Perhimpunan Dokter Spesialis Mata Indonesia*), in 2017 approximately 8 million people experienced visual impairment, consisting of about 1.6 million individuals with total blindness and 6.4 million individuals with moderate to severe visual impairment (Kemenkes, 2021). This condition significantly limits affected individuals in performing daily activities independently. One of the most challenging tasks for visually impaired individuals is identifying and distinguishing banknote during everyday transactions(Hafiar et al., 2020). Conventional methods, such as folding banknotes or relying on tactile features, are prone to errors due to memory limitations, banknote wear, and the inability to verify denominations accurately, increasing the risk of financial loss and fraud (Ikhsan & Sari, 2018). These challenges highlight the urgent need for accessible and reliable assistive technologies that enable visually impaired users to recognize currency independently (Saputra et al., 2024).

Several assistive solutions have been proposed, including mobile applications and hardware based devices. One example is *Cash Reader*, which recognizes banknote denominations using vibration feedback however, its free version supports only limited denominations and lacks audio output, requiring users to memorize vibration patterns (Cashreader, 2019). Another solution, *Kasentra*, is a prototype wearable device using a color sensor and microcontroller to identify banknotes and provide audio feedback, but its design is less practical for daily mobility (Restuputri et al., 2018). These limitations indicate the need for a more accessible, accurate, and practical mobile based banknote recognition system for visually impaired users.

Motivated by these limitations, this study proposes the development of *Currency Sense*, an Android based assistive application for recognizing Indonesian Rupiah banknote denominations using a Convolutional Neural Network. CNN is selected for its proven capability in extracting and learning complex visual patterns from images, enabling robust currency classification under varying conditions (Ibrahim et al., 2023). The application is designed to provide audio based feedback and a simple interface tailored to visually impaired users, ensuring ease of use in real world scenarios. This research aims to design and implement a CNN based classification model for Indonesian banknotes, integrate the model into an Android application, evaluate its performance using a confusion matrix, and assess system functionality through black box testing. The study contributes to the advancement of artificial intelligence based accessibility technologies and supports inclusive development by enhancing financial independence for visually impaired individuals.

II. LITERATURE REVIEW

2.1 Assistive Technology

Technology has become an integral part of human life, supporting and simplifying daily activities. In the context of special education and disability support, adaptive tools and assistive media play a crucial role in addressing the needs of individuals with disabilities. Assistive technology refers to devices or systems that are specifically designed or modified to enhance the functional capabilities of persons with disabilities, enabling them to perform daily activities and participate in learning processes more effectively. The use of assistive technology contributes significantly to the development of independent living skills, allowing individuals with disabilities to carry out daily tasks with reduced reliance on external assistance. Consequently, assistive technology is widely recognized as a key component in improving autonomy, accessibility, and quality of life for people with special needs (Damastuti, 2021).

2.2 Visually Impaired

Visually impaired individuals experience limitations in visual function. According to Dermawan (2018), visual impairment is defined as a visual acuity of less than 6/60 even with visual aids, or a complete loss of vision. Due to these limitations, visually impaired individuals primarily rely on tactile and auditory senses in learning and daily activities. Visual impairment is generally classified into two categories, total blindness and low vision. Total blindness refers to the absence of light perception and affects a relatively small proportion of visually impaired individuals, while low vision describes a condition in which residual vision remains but cannot be adequately corrected through conventional treatments. Despite retaining partial vision, individuals with low vision still face significant challenges in performing vision dependent daily activities (Lee et al., 2017).

2.3 Machine Learning

Machine learning is a branch of artificial intelligence and computer science that focuses on enabling systems to learn from data and improve performance over time. It operates by training models to identify patterns and make predictions based on available datasets. This approach is well suited for image recognition tasks, as it allows models to handle variations in lighting, image quality, and environmental conditions. Machine learning models rely on diverse training data rather than single images, enabling more robust pattern recognition. Prior to model development, data preprocessing is essential to address noise, missing values, and inconsistent formats that may negatively affect model performance (Erlin et al., 2021).

2.4 Convolutional Neural Network

A Convolutional Neural Network is a type of artificial neural network specifically designed for processing image data. CNN are widely used for image classification, identification, and pattern recognition tasks due to their ability to extract hierarchical visual features. The CNN architecture enables the model to capture spatial patterns in images by applying convolution operations using multiple kernels. These operations are performed on two dimensional data, such as images, allowing the network to learn meaningful visual representations efficiently. Consequently, CNN are particularly effective for tasks involving structured two dimensional data (Anhar, 2023).

2.5 Software Development Life Cycle Prototyping

The prototyping model is a development approach used to rapidly gather user requirements by creating an early version of the system that can be directly viewed and tested by users. This approach emphasizes system functionality and interface aspects, allowing users to understand the system workflow and provide immediate feedback. The resulting prototype is iteratively evaluated and refined to clarify and validate system requirements more accurately. By actively involving users throughout the development process, the prototyping model facilitates smoother system implementation and improves usability (Kurniyanti & Murdiani, 2022). In this study, the prototyping approach is selected due to the nature of the system, which is an assistive application tailored to the specific needs of visually impaired users.

2.6 Black Box Testing

Black box testing is a software testing method that focuses on verifying system functionality based on input and output behavior without requiring knowledge of the internal program structure or source code. In this approach, testers evaluate how the system responds to various inputs and whether the resulting outputs conform to predefined requirements and specifications. Black box testing simulates end user interactions, making it effective for validating that application features function as intended. This method is typically applied in the final stage of software development, when the system is fully implemented and ready for functional evaluation. Through this testing, developers can identify functional failures, incorrect input handling, interface inconsistencies, and unexpected system responses. By emphasizing the users perspective, black box testing plays an important role in ensuring software usability, reliability, and overall quality prior to deployment (Fedianto et al., 2024).

2.7 Confusion Matrix

A confusion matrix is a commonly used evaluation method to measure the performance of a classification system. It is presented in the form of a table that displays the number of test data instances that are correctly and incorrectly classified, making it easier to evaluate the accuracy of a classification system. By using a confusion matrix, the performance of a classification system can be analyzed in detail and areas where misclassification occurs can be clearly identified (Nurhidayat & Dewi, 2023). A confusion matrix represents classification results using four main components (Grandis et al., 2021):

a. True Positive (TP): The number of data instances that belong to the positive class and are correctly predicted as positive.
b. True Negative (TN): The number of data instances that belong to the negative class and are correctly predicted as negative.
c. False Positive (FP): The number of data instances that do not belong to the positive class but are incorrectly predicted as positive.
d. False Negative (FN): The number of data instances that belong to the positive class but are incorrectly predicted as negative.

III. METHOD

Type of Research

This study employs a quantitative research approach to design, develop, and evaluate an Android based assistive application for recognizing Indonesian Rupiah banknote denominations using a Convolutional Neural Network. Quantitative methods are applied to objectively measure system performance through numerical evaluation metrics derived from classification results. This research is classified as field research, as it involves direct development and real world testing of an application intended for visually impaired users.

System Development

System development followed the System Development Life Cycle (SDLC) using a prototyping approach. This method enables iterative refinement of the application based on user feedback. Functional requirements were identified through user input, followed by prototype development and repeated testing until the final version of the application was achieved. Application functionality was evaluated using black box testing.

Model Development

A Convolutional Neural Network (CNN) model was developed due to its effectiveness in image based feature extraction and classification. The model was trained using preprocessed and augmented images to enhance robustness against variations in lighting conditions and image quality. Standard training parameters were applied without extensive hyperparameter tuning, as the primary focus of this study is system implementation rather than model optimization. Model performance evaluation focuses on measurable metrics, including accuracy, precision, recall, and F1-score, calculated using a confusion matrix to assess the classification capability of the CNN model.

Data Collection

Data collection consists of primary and secondary data. Primary data were obtained through online interviews, observations, and questionnaires involving five visually impaired participants. Interviews and observations were conducted to identify user needs and interaction constraints, while questionnaires were used during black box testing to evaluate application functionality. Secondary data consist of Indonesian Rupiah banknote image datasets obtained from Kaggle, covering the 2016 and 2022 emissions for model training and validation. The dataset comprises 3,500 images evenly distributed across seven denomination classes, with 500 images per class. An 80:20 split was applied for training and validation to ensure balanced learning and reliable performance evaluation.

Population

The population in this study includes all visually impaired individuals in Indonesia who require assistance in identifying currency for daily activities. This population consists of individuals with total or partial visual impairments who use cash as a medium for financial transactions.

Sample

The sample was selected from the population using a purposive sampling technique. Purposive sampling is a sampling technique that is not based on any specific level, but rather selected according to certain criteria and objectives relevant to the research (Akbar, 2014). The criteria for sample selection in this study are as follows: (1) visually impaired students at Universitas Negeri Surabaya who require assistance in identifying banknote denominations; (2) willingness to use the *Currency Sense* application and provide feedback during the testing process; (3) access to a smartphone compatible with the application; (4) basic ability to operate smartphone accessibility features such as screen readers (e.g., TalkBack); and (5) active involvement in daily cash transactions.

The selection of visually impaired students from Universitas Negeri Surabaya as research participants was based on several considerations. In terms of accessibility, the university environment enables the researcher to reach respondents efficiently, facilitating effective data collection and application testing. Additionally, students are considered to have relatively higher digital literacy compared to the general population, allowing them to operate smartphone based applications and provide meaningful feedback on the applications functionality and usability. Furthermore, visually impaired students are generally within a productive age range and actively engage in daily financial transactions, making the need for a banknote identification assistive tool highly relevant in both academic and everyday contexts.

IV. RESULTS AND DISCUSSION

Results of Model and Application Development Using the SDLC Prototyping Method

This section presents the results of each stage of system development conducted using the Software Development Life Cycle (SDLC) with a prototyping approach. The SDLC Prototyping method emphasizes the creation of an initial system model (prototype) to obtain user feedback before full system implementation. This approach allows developers to identify user needs more accurately and refine system functionality iteratively.

The SDLC Prototyping process consists of several stages, including requirement analysis, design, prototyping, development, and testing. Each stage is described in detail to illustrate the systematic process of developing the model and application until a final version that is ready for user implementation is achieved. The following subsections explain each development stage using the SDLC Prototyping method.

1. Requirement Analysis

The requirement analysis stage in this study aims to identify functional requirements, non-functional requirements, and system requirements for designing the classification model and the *Currency Sense* application as an assistive tool for recognizing Indonesian Rupiah banknote denominations for visually impaired users.

Based on the interview results, this study involved five visually impaired respondents with diverse characteristics. The respondents consisted of both male and female participants with different visual impairment conditions, including one individual with low vision and four individuals with total blindness. All respondents were active smartphone users in their daily activities and were familiar with accessibility support technologies such as TalkBack. This indicates that the respondents possess a good level of technology acceptance and digital literacy, particularly in utilizing mobile devices as assistive tools for daily activities, including cash based transactions.

These respondent characteristics provide a relevant foundation for the system requirement analysis, ensuring that the identified requirements accurately reflect the needs and conditions of visually impaired users as the target users of the developed application. Based on the interview analysis, the system requirements are categorized into functional and non-functional requirements, as described below.

a. Functional Requirements:
   The application must be able to detect and classify banknote denominations in real time using the smartphone camera. It must provide clear audio feedback to users after the detection results are generated. In addition, the application should support offline usage to ensure accessibility under various conditions without requiring an internet connection.

b. Non-Functional Requirements:
   The classification model is required to achieve a high level of accuracy in recognizing banknote denominations. Furthermore, the application interface must be designed to be simple, intuitive, and user friendly for visually impaired users.

Based on the results of the functional and non-functional requirements analysis, software and development environment support is required to accommodate system needs, including image processing, model development, and application implementation on Android devices.

2. Develop

After user requirements were identified through interviews and needs analysis, the next stage involved designing the initial system and application architecture. This initial design serves as a reference for defining the overall system structure before proceeding to the prototyping phase. In this study, the design stage consists of the following components.

a. User Flowchart

In the early design phase of the Currency Sense application, a user flowchart was employed as a visual tool to represent the overall user interaction flow. The diagram illustrates how users navigate between application screens, including available decision points and navigation paths. The user flowchart functions as a primary guideline that explains the logical sequence of application usage, starting from the initial launch to the completion of the process. In the context of Currency Sense, the flow includes the splash screen, home page, banknote scanning process, result display, and navigation to the history page. Each node and transition in the diagram represents a user action or system response required to achieve specific application objectives.

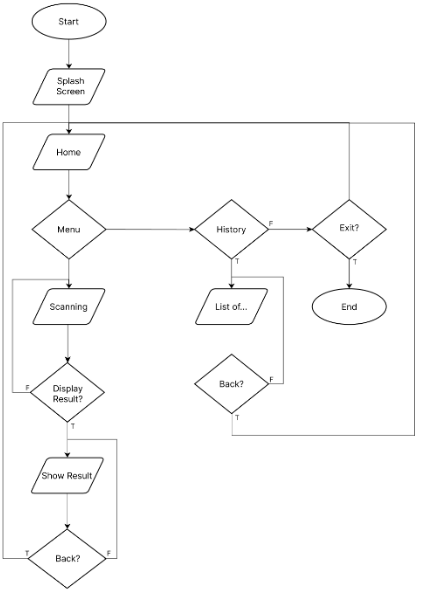

Figure 1 User Flowchart

b. Use Case Diagram

The use case diagram is utilized to describe the interaction between system actors and the main functionalities provided by the application. This diagram facilitates an understanding of how each actor interacts with the system and what functions are accessible to them. In the Currency Sense application, the primary actor is the visually impaired user, who can perform core functions such as scanning banknotes and accessing detection history through the history menu. In addition, an Admin actor is defined, responsible for updating the CNN model used by the system. Model updates are conducted outside the mobile application environment with the objective of improving detection accuracy and performance using newly available training data. Consequently, the Admin focuses on technical model management, while visually impaired users interact directly with the applications primary features.

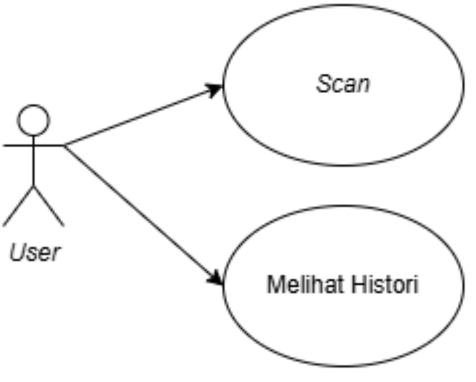

Figure 2 Use Case Diagram

c. Activity Diagram

The activity diagram illustrates the main workflow involved in both the development and usage of the Currency Sense application, involving three main components: the user, the system, and the admin. This diagram provides a comprehensive overview of how each component interacts to execute essential system functions, from the banknote scanning process to CNN model updates performed externally.

From the user perspective, the activity begins when the application is launched and the main menu is displayed. Users may choose either to scan a banknote or view detection history. When the scan option is selected, the system activates the camera to capture an image of the banknote, which is then processed using a Convolutional Neural Network to identify the Rupiah denomination. The detection result is presented in both text and audio formats to ensure accessibility for visually impaired users. If the history option is selected, the system displays a list of previous detection results. The activity concludes when the user exits the application.

From the admin perspective, activities focus on maintaining and improving model performance. The admin uploads new or additional banknote image datasets, retrains the CNN model to enhance robustness under various conditions (e.g., different lighting or folded banknotes), and updates the trained model for use in the application. All model update processes are performed outside the mobile application environment.

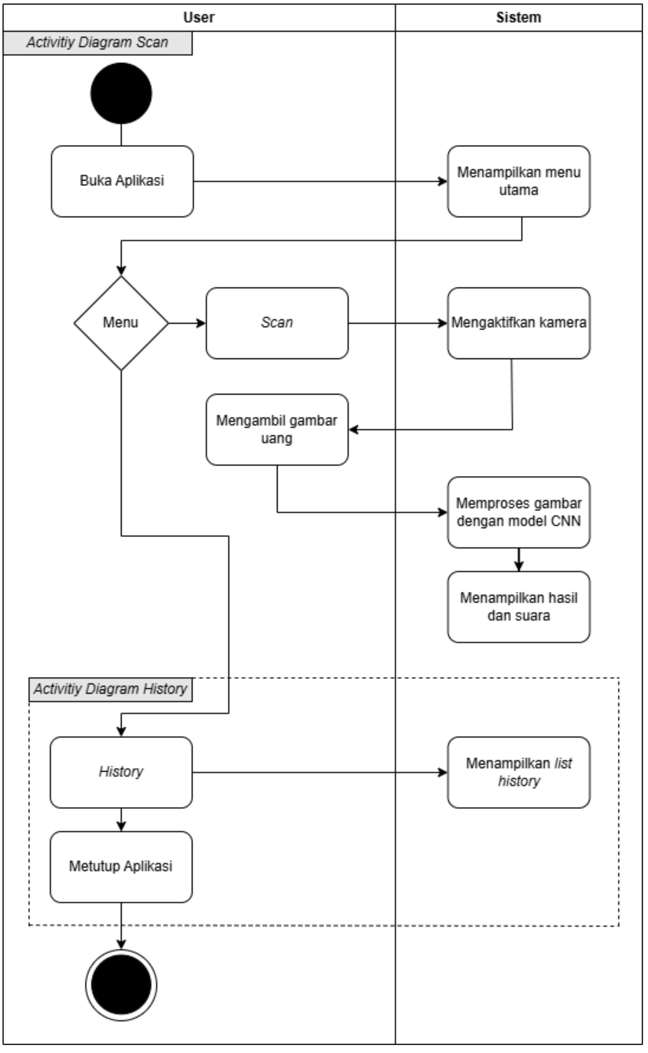

Figure 3 Activity Diagram

d. Sequence Diagram

The sequence diagram is used to represent interactions between system components based on chronological order. It illustrates how messages or data are exchanged between objects during a specific process. This diagram assists developers in understanding logical interactions among components within a processing cycle, thereby supporting structured and efficient system implementation.

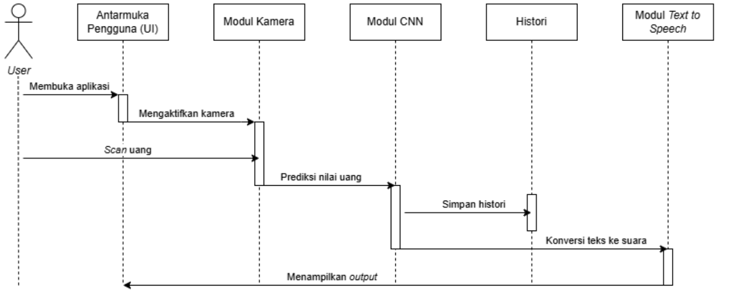

Figure 4 Sequence Diagram

3. Prototyping

The development process proceeded to the prototyping stage, where the previously defined system design was implemented into an initial working model. Prototyping was used to visualize the system workflow and to evaluate the alignment between the proposed design and the identified user requirements. This stage enables early validation of system functionality and usability before full implementation.

a. Low-Fidelity Wireframe Design Using Figma

A low-fidelity prototype was developed to represent the basic structure of the Currency Sense application interface. At this stage, the design focused on the layout of interface elements, relative sizing, and user interaction flow, without emphasizing color schemes, typography, or detailed visual aesthetics. The low-fidelity prototype served as a wireframe to ensure that navigation flow and interface structure met user requirements. The wireframe design was created using Figma, and each application screen is described as follows.

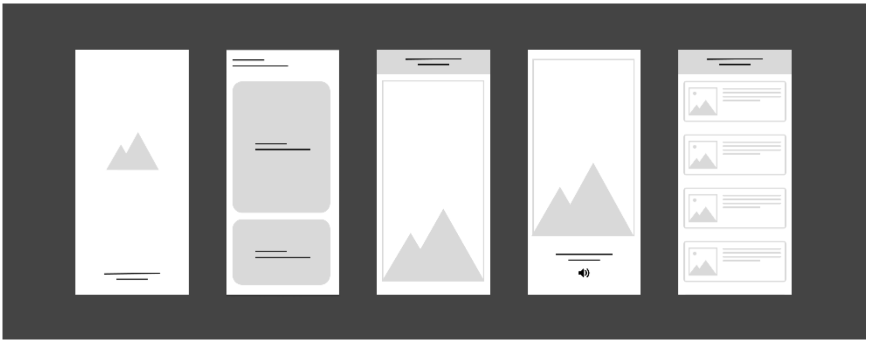

Figure 5 Wireframe Aplikasi Currency Sense

The Currency Sense application consists of several main interface screens designed to support accessible user interaction. The splash screen displays the application logo at the center of the screen and serves as an initialization stage before the main interface is loaded. The home screen provides navigation to the core application features through a minimalistic layout with large buttons to ensure accessibility for visually impaired and low vision users. The scan screen features a large rectangular area representing the camera viewfinder, emphasizing the primary function of capturing banknote images for further processing. The result screen presents the captured image in the upper section and displays the classification output in the form of the recognized banknote denomination in the lower section. Finally, the history screen shows a list of previously identified banknotes, allowing users to review past detection results without performing a new scanning process.

b. High-Fidelity Prototype Design Using Figma

The next stage involved designing a high-fidelity prototype using Figma. This design aims to provide a detailed visual representation of the application workflow prior to implementation in code. The resulting design served as a reference for developing the application layout in Android Studio, facilitating a smooth transition from design to implementation. The prototype developed in this study is non interactive and focuses solely on visual interface representation.

Figure 6 Prototype Aplikasi Currency Sense

The high-fidelity prototype consists of the following application screens. The splash screen is displayed immediately after the application is launched and presents the application logo and name, Currency Sense, as the initial identity of the application. This screen allows the system to perform initialization processes before displaying the main interface. The home screen is designed with two primary buttons, namely Scan and History, using a high contrast black and green color scheme as the applications visual identity. Large button sizes are applied to enhance accessibility and ease of interaction. The scan screen provides a banknote scanning feature with a large rectangular guide area to assist users in positioning the banknote correctly. A “SCAN” label is displayed at the top of the screen to indicate the page function. This screen captures banknote images through the device camera, which are then classified by the CNN model to identify the Rupiah denomination. The result screen displays the captured banknote image along with the recognized denomination in textual form, such as “Rp100.000,” emphasizing clarity of information and supported by audio feedback. Finally, the history screen presents a list of previous scanning results, where each item includes the banknote image, the recognized denomination, and the scanning date. This feature enables users to review past identification results without performing a new scanning process.

4. Customer Evaluation

The customer evaluation stage was conducted after the application prototype had been fully developed and all core features were functioning properly. This evaluation involved visually impaired users as the primary target users of the application. The assessment focused on usability, navigation clarity, and application effectiveness when used in conjunction with TalkBack, the standard Android accessibility service for visually impaired users.

During the evaluation process, participants were asked to test two main features, the banknote scanning feature and the scan history feature. Throughout the testing session, the researcher observed how users operated the application using TalkBack, their responses to the navigation flow, and any obstacles encountered during interaction. The results indicated that the application was generally usable with TalkBack however, several issues were identified regarding the applications built in audio instructions.

Initially, the application provided automatic voice instructions on certain screens, such as guidance on the home screen and prompts on the scan screen. User feedback revealed that these audio instructions caused confusion because they overlapped with TalkBack screen reading output. Since TalkBack automatically reads interface elements including button labels, page titles, and other interactive components, the additional audio instructions were considered redundant and negatively affected the user experience. As a result, users expressed a preference for relying solely on TalkBack for navigation guidance.

An exception was identified on the result screen. During testing, TalkBack occasionally exhibited a delay in reading the classification output due to announcing the application name before reading the result, even when the name was not visually present on the screen. This behavior caused a delay before users received the essential information. To address this issue, a dedicated audio output was retained on the result screen to immediately announce the detected banknote denomination, ensuring timely delivery of critical information.

5. Review and Refine

The review and refine stage was conducted after the customer evaluation phase identified several findings and user feedback from visually impaired participants. This stage represents an iterative process aimed at reexamining the system design, interaction flow, and application functionality, followed by refinements to improve usability and better align the application with user needs.

Based on the evaluation results, several aspects required improvement, particularly regarding the interaction between the application and TalkBack, the accessibility service used by visually impaired users. The primary feedback indicated that the applications built in audio instructions caused confusion due to overlapping with TalkBack voice output. Consequently, the following refinements were implemented.

First, all built in audio instructions were removed from the application. Previously, automatic audio prompts were played on multiple screens, such as navigation instructions on the home screen and scanning prompts on the scan screen. These instructions were eliminated because TalkBack already reads button labels and page titles effectively. Removing the redundant audio improved clarity, prevented overlapping outputs, and enhanced the overall navigation experience.

Second, the interface structure was optimized to be more compatible with TalkBack. All interface elements were assigned consistent and descriptive labels, enabling TalkBack to accurately convey each components function. Additionally, the arrangement of interface elements was refined to ensure a more logical and intuitive reading order during screen navigation.

Third, button accessibility was enhanced by increasing button sizes and expanding interaction areas to reduce the risk of accidental input. On the scan screen, the entire screen area was designated as a single scanning trigger, allowing users to initiate the scanning process without searching for a specific button.

The refinement process was conducted through multiple iterations until a stable prototype was achieved that was easy to operate and fully compatible with TalkBack. This stage ensured that the application not only functioned correctly from a technical perspective but also provided an optimal user experience for individuals with visual impairments. Upon completion of this stage, the prototype was deemed ready to serve as the foundation for further development.

6. Development

The development stage focuses on implementing the refined prototype into a functional system through coding and system integration. At this stage, the application logic, machine learning model, and user interface design were translated into an operational Android based assistive application. The development process emphasized functionality, accessibility, and system stability rather than extensive optimization.

The system was developed using a combination of machine learning and mobile application technologies. The CNN based banknote identification model was implemented using Python with deep learning libraries such as TensorFlow and Keras. The model was trained using preprocessed and augmented Rupiah banknote images to enhance robustness against variations in lighting conditions and image quality. After the training process was completed, the trained model was converted into a mobile compatible format to enable deployment on Android devices.

The Android application was developed using Android Studio as the integrated development environment. The application handles image acquisition through the device camera, invokes the CNN model for inference, and presents the classification results to the user. The development process followed the refined design obtained from previous stages, ensuring that interface components, navigation flow, and interaction patterns were consistent with accessibility requirements identified during user evaluation.

A key aspect of this stage was the integration of accessibility features, particularly compatibility with TalkBack. Interface elements were implemented with clear and consistent labels to ensure that TalkBack could accurately interpret and announce each component. Audio output was selectively implemented only for the classification result screen to provide immediate feedback on the detected banknote denomination, while redundant instructional audio was removed to prevent overlapping voice outputs.

The overall system workflow during development consists of capturing a banknote image via the smartphone camera, preprocessing the image, performing denomination classification using the CNN model, and delivering the result in textual and audio formats. The development stage resulted in a fully functional prototype that integrates image processing, machine learning inference, and accessible user interaction within a single mobile application. This implementation serves as the foundation for subsequent testing, evaluation, and real world usage.

7. Testing

7.1 Black Box Testing Results

The testing phase was conducted to ensure that all application functionalities operated in accordance with the user requirements identified during the analysis stage. Testing was performed using the Black Box Testing method, which focuses on evaluating system functionality and output without considering internal logic or source code implementation. In this approach, testers provide specific inputs and observe whether the resulting outputs meet the expected outcomes. Several test scenarios were designed to cover the core features of the application, including banknote scanning, prediction result display, and storage of identification history. The testing process aimed to verify that each function performed as intended and that the application met the defined functional requirements.

a. Home Screen

Table 1 Black Box Testing Results of the Home Screen

No	Test Case	Test Steps	Expected Result	Actual Result	Status
1	Scan Button	Click the scan button	The application opens the device camera	The camera is successfully opened	Valid
2	History Button	Click the history button	The application opens the history screen	The history screen is displayed	Valid

b. Scan Screen

Table 2 Black Box Testing Results of the Scan Screen

No	Test Case	Test Steps	Expected Result	Actual Result	Status
1	Camera Access	Click the scan button	The camera opens and is ready to capture images	The camera is activated	Valid
2	Banknote Detection	Point the camera at a banknote and tap the center area	The system recognizes the banknote and displays the result	The banknote denomination is detected and audio is played	Valid
3	Back Navigation	Click the back button	The system returns directly to the camera screen	The camera screen is reactivated	Valid
4	Rescan	Point the camera at a banknote and tap the center area	The system performs a new scanning process	The banknote denomination is detected and audio is played	Valid

c. Result Screen

Table 3 Black Box Testing Results of the Result Screen

No	Test Case	Test Steps	Expected Result	Actual Result	Status
1	Display Banknote Denominati	After the scan result is	The system displays the detected	The banknote denomination text is	Valid

	on	generated	banknote denomination	displayed	
2	Audio Output of Denomination	After the scan result is generated	The system produces audio corresponding to the detected denomination	The audio output matches the detected denomination	Valid

d. History Screen

Table 4 Black Box Testing Results of the History Screen

No	Test Case	Test Steps	Expected Result	Actual Result	Status
1	History Data Storage	Perform a scan	The history displays the most recent scan data	The data appears with the detected denomination and date	Valid
2	Display Format and Data Order	View the history list	Each item is displayed with a banknote image, denomination, and scan date	The data is neatly displayed and sorted in descending order by date	Valid

7.2 Summary of Black Box Testing Results

Table 5 presents the results of Black Box Testing conducted by five participants to evaluate the main functionalities on each application screen. This testing was performed to ensure that each feature operates in accordance with the predefined usage scenarios.

Table 5 Black Box Testing Conducted with Participants

No	**Test Class**	**Test Case**	**P1**	**P2**	**P3**	**P4**	**P5**
1	Home	Scan Button	✓	✓	✓	✓	✓
		History Button	✓	✓	✓	✓	✓
2	Scan	Camera Access	✓	✓	✓	✓	✓
		Banknote Detection	✓	✓	✓	✓	✓
		Back Navigation	✓	✓	✓	✓	✓
		Rescan	✓	✓	✓	✓	✓
3	Result	Display Banknote Denomination	✓	✓	✓	✓	✓
		Audio Output of Denomination	✓	✓	✓	✓	✓
4	History	History Data Storage	✓	✓	✓	✓	✓

		Display Format and Data Order	✓	✓	✓	✓	✓

Note:
P: Participants
✓: Test passed

7.3 Calculation of Effectiveness Scores for Each Table

Table 6 presents a summary of the Black Box Testing results conducted to assess the success rate of each application module. The testing covered four main screens, Home, Scan, Result, and History with a total of 10 test cases evaluated. Each test case was assessed by five participants, and the testing outcomes were classified into two categories Valid and Not Valid.

Table 6 Summary of Black Box Testing Results

No	Test Class	Valid	Not Valid	Total *Test Case*
1	Home Screen	2	0	2
2	Scan Screen	4	0	4
3	Result Screen	2	0	2
4	History Screen	2	0	2
	Total	10	0	10

Effectiveness calculation in the Black Box Testing stage was conducted to determine the success rate of each test case in meeting the predefined functional requirements. Effectiveness was calculated based on the number of successful test cases compared to the total number of test cases on each application screen (Suharyono et al., 2024). The formula used in this calculation is as follows:

$$\text{Effectiveness (\%)}\ \frac{\text{Number of Valid Test Cases}}{\text{Total Test Case}} \times 100\% \qquad (1)$$

Table 7 Effectiveness of Each Testing Module

Test Class	Calculation Results
Home Screen	$\left(\frac{2}{2}\right) \times 100\%$ = 100%
Scan Screen	$\left(\frac{4}{4}\right) \times 100\%$ =100%
Result Screen	$\left(\frac{2}{2}\right) \times 100\%$ =100%
History Screen	$\left(\frac{2}{2}\right) \times 100\%$ =100%

Based on the Black Box Testing results, all test cases across each application module were successfully executed and categorized as valid. The effectiveness calculation shows that each testing module achieved an effectiveness score of 100%, indicating that all application functionalities operated in accordance with the predefined functional requirements.

8. Release

This stage represents the final phase of the system development lifecycle and marks the completion of the application design and implementation process. After all core functionalities were successfully implemented and tested, the application was released in the form of an Android Package (APK). The APK format was selected to allow direct installation and execution on Android devices without requiring additional setup from users. The APK was generated using Android Studio through the *Build Bundle(s) / APK(s)* feature, which compiles the complete source code and the integrated TensorFlow Lite model into a single application package. Upon completion of the release stage, the Currency Sense application fulfilled the research objective as an Android based assistive system capable of independently supporting visually impaired users in identifying banknote denominations and ready for real world deployment.

Classification Model Performance Results Based on Confusion Matrix Evaluation

As an evaluation context, it is important to emphasize that the performance assessment of the CNN model was conducted after the completion of the entire training process. This evaluation aims to measure how accurately the trained model, using predefined parameters such as epochs, batch size, and learning rate, classifies images of Indonesian Rupiah banknotes. The evaluation process began with monitoring the stability of the learning process through accuracy and loss learning curves. Subsequently, the model was tested using a testing dataset to assess its generalization capability. The prediction results obtained from this testing phase were then summarized in a confusion matrix, which compares the actual labels with the labels predicted by the model. Based on the four main components of the confusion matrix, True Positive, True Negative, False Positive, and False Negative, quantitative performance metrics, including accuracy, precision, recall, and F1-score, were calculated to provide a comprehensive evaluation of the models reliability.

1. Learning Curve Analysis

During the training process, the accuracy and loss learning curves were monitored to evaluate the stability and convergence of the models learning behavior.

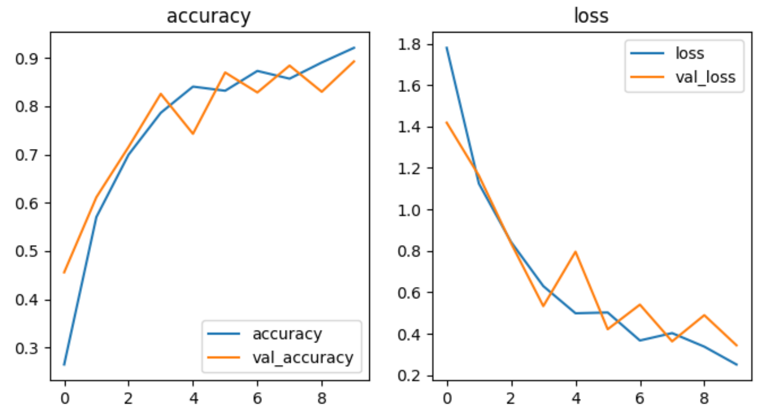

Figure 7 Learning Curve Results

Based on the generated learning curves, the training accuracy and validation accuracy exhibit closely aligned trends, with the validation accuracy reaching a peak of approximately 0.90 and the training accuracy around 0.93 at the final epoch. This stability indicates that the model learns effectively and is able to generalize knowledge from the training data to the validation data. Meanwhile, the loss curves show a consistent decreasing trend for both training and validation loss, although slight fluctuations are observed in the validation loss. This decline confirms that the models prediction error decreases throughout the training process, demonstrating a well controlled learning behavior.

2. Confusion Matrix Evaluation

After the training process was completed, the model was evaluated using a testing dataset to assess its generalization capability on unseen data. The performance of the classification model in this study was evaluated using a confusion matrix to determine the accuracy of the model in classifying Indonesian Rupiah banknote denominations. The confusion matrix presents the number of correct and incorrect predictions produced by the model for each class. Overall, the confusion matrix provides quantitative evidence that the model has successfully learned the discriminative features of different Rupiah banknote denominations and demonstrates strong generalization performance on the testing data. These findings are essential to ensure that the model not only achieves high accuracy but also remains robust and reliable when applied in real world conditions. Figure 8 illustrates the confusion matrix results obtained from testing the CNN model on 700 test images that have undergone preprocessing and data augmentation.

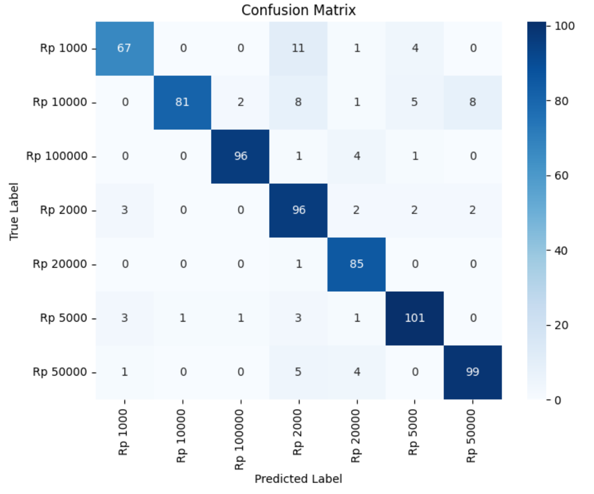

Figure 8 Confusion Matrix Results

Based on Figure 8, the diagonal values (67, 81, 96, 96, 85, 101, and 99) represent the number of correct predictions for each currency denomination class, while the off diagonal values indicate misclassifications or confusion between classes. Out of a total of 700 test samples, the model correctly predicted 625 instances, while 75 instances were misclassified. Using the accuracy formula, the overall model accuracy was calculated as 89.28%.

The evaluation results indicate that the CNN model is capable of correctly recognizing most of the currency denominations. The highest accuracy was observed in the Rp5,000 and Rp50,000 classes, with 101 and 99 correct predictions, respectively, suggesting that the visual characteristics of these denominations are sufficiently distinct for the model to recognize them accurately. The error patterns observed in some classes reveal certain visual similarities between denominations. For example, the Rp1,000 class was occasionally misclassified as Rp2,000 or Rp5,000, likely due to similar dominant colors, image noise, poor lighting conditions, or banknote orientation during image capture.

Overall, the confusion matrix demonstrates that most classes exhibit stable performance, although some still require improvement. Performance enhancement can be achieved by increasing training data variation, applying more suitable augmentation techniques, or developing deeper CNN architectures to capture complex visual features. These findings indicate that the model has successfully learned discriminative features from Indonesian banknote images and exhibits strong generalization capabilities on test data, while also providing a quantitative basis to identify areas for optimization to achieve higher and more consistent accuracy.

3. Evaluation Metrics Calculation

The confusion matrix consists of four main components: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). Based on the confusion matrix shown in Figure 4.16, the summary of TP, TN, FP, and FN for each class is as follows:

Table 8 Confusion Matrix per Currency Denomination

Class	True Positive	True Negative	False Positive	False Negative
Rp 1.000	67	229	7	16
Rp 2.000	96	167	29	9
Rp 5.000	101	160	12	9
Rp 10.000	81	247	1	24
Rp 20.000	85	201	13	1
Rp 50.000	99	180	10	10
Rp 100.000	96	231	3	6

This summary serves as the basis for calculating various evaluation metrics, including accuracy, precision, recall, and F1-score, each of which provides a specific insight into the models capability in classifying each currency denomination. Based on these values, the evaluation metrics are calculated to obtain a comprehensive understanding of the models overall performance. The results for each class are presented in Figure 9.

	Nominal	Accuracy	Precision	Recall	F1-Score
0	Rp 1000	0.807229	0.905405	0.807229	0.853503
1	Rp 10000	0.771429	0.987805	0.771429	0.866310
2	Rp 100000	0.941176	0.969697	0.941176	0.955224
3	Rp 2000	0.914286	0.768000	0.914286	0.834783
4	Rp 20000	0.988372	0.867347	0.988372	0.923913
5	Rp 5000	0.918182	0.893805	0.918182	0.905830
6	Rp 50000	0.908257	0.908257	0.908257	0.908257
7	Rata-rata	0.892704	0.900045	0.892704	0.892546

Figure 9 Evaluation Metrics Results

a. Accuracy

Accuracy represents the proportion of correct predictions out of all samples. The highest value was observed for the Rp20,000 class (0.9883), indicating that nearly all predictions for this class were correct, while the lowest value was found in the Rp10,000 class (0.7714), suggesting more misclassifications for this class compared to others. The overall average accuracy of the model was 0.8927, indicating satisfactory performance.

b. Precision

Precision indicates the proportion of true positive predictions among all positive predictions. The highest precision was achieved by Rp10,000 (0.9878), meaning almost all predictions for this denomination were correct, while the lowest was for Rp2,000 (0.7680), indicating several false positive predictions for this class. The models average precision was 0.9000, showing generally good capability in predicting positive classes.

c. Recall

Recall represents the proportion of actual positive samples correctly identified by the model. The highest recall was observed for Rp20,000 (0.9883), indicating that nearly all samples of this class were correctly detected, while the lowest recall was for Rp10,000 (0.7714), suggesting some samples were missed. The average recall was 0.8927, indicating satisfactory detection performance across all classes.

d. F1-Score

The F1-Score combines precision and recall to assess the balance between them. The highest F1-Score was obtained for Rp100,000 (0.9552), indicating excellent classification performance, while the lowest was for Rp2,000 (0.8348), which is still acceptable. The average F1-Score was 0.8925, reflecting a balanced performance between precision and recall.

Overall, the model demonstrates consistent and good performance, with average accuracy, precision, recall, and F1-Score ranging from 0.89 to 0.90. Certain classes, such as Rp10,000 and Rp2,000, have relatively lower metrics, indicating potential misclassifications due to visual similarities with other denominations. In contrast, classes with high precision and recall, such as Rp20,000 and Rp100,000, show that the model can classify these denominations very effectively.

V. CONCLUSION AND RECOMMENDATION

Conclusion

Based on the study on the development of a Rupiah banknote denomination identification model in the *Currency Sense* application for visually impaired users, the following conclusions can be drawn:

1. Application Development Using the SDLC Prototyping Method Was Successful
   The SDLC Prototyping approach was successfully implemented, providing an iterative development workflow. Through requirements analysis, prototype creation, testing, and system refinement, the *Currency Sense* application was developed according to user needs. Prototyping facilitated feature validation, interface improvement, and enhanced user experience based on feedback.
2. System Implementation and Application Features Functioned According to Requirements
   The application successfully performed key functions, including capturing banknote images, processing them using a CNN model, and presenting denomination outputs both visually and audibly. Real time predictions

performed well, indicating that the application met functional requirements as defined during requirements analysis.

3. CNN Model Successfully Developed Using Preprocessed Rupiah Data
The classification model was developed using Convolutional Neural Networks, supported by preprocessing and data augmentation. Training results indicated that the model could effectively learn visual patterns of various denominations. CNN was proven suitable for capturing visual features such as color, texture, and unique patterns of Rupiah banknotes.

4. Model Performance Based on Confusion Matrix Shows High Accuracy
Evaluation using the confusion matrix indicated 625 correct predictions out of 700 test samples, yielding an overall accuracy of 89.28%. The diagonal values show that most classes have stable accuracy, particularly Rp5,000 and Rp50,000. Some classes, such as Rp1,000 and Rp2,000, still experienced misclassification due to color similarities and variations in image conditions. These results demonstrate good generalization capability of the model, although there is room for further optimization.

Recommendation

Based on the findings and evaluation, the following recommendations are proposed for further development:

1. Expand the dataset with more diverse images, including worn, folded, shadowed notes, or images taken from various angles, to reduce misclassifications and enhance model reliability in real world conditions.
2. Augmentation techniques can be expanded to include extreme rotations, light intensity changes, blur, or other perspectives. The CNN architecture can be upgraded to modern models such as EfficientNet or MobileNetV3 for improved performance and efficiency on mobile devices.
3. Add features such as interactive voice guidance, haptic feedback, and options for zoom or high contrast for low vision users to enhance accessibility and user experience.
4. Test the trained model with new banknote images to validate the consistency of predicted denominations and model confidence before integrating it into the Android application.
5. Carry out field testing with a larger number of visually impaired participants using standardized methods, such as the System Usability Scale, to assess satisfaction and application effectiveness, providing a basis for the next development cycle.

REFERENCES

Akbar, A. F. (2014). *Respon Siswa (SSB) Terhadap Pelatih Yang Berlatar Belakang Akademik Dan Non Akademik*. 1–4.

Anhar, R. A. P. (2023). Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, *11*(2), 466. https://doi.org/10.26760/elkomika.v11i2.466

Cashreader. (2019). *Apa yang membuat pembaca uang ini spesial*. https://cashreader.app/id/

Damastuti, E. (2021). Teknologi Asistif. In *Proceedings of the National Academy of Sciences* (Vol. 3, Issue 1). http://dx.doi.org/10.1016/j.bpj.2015.06.056%0Ahttps://academic.oup.com/bioinformatics/article-abstract/34/13/2201/4852827%0Ainternal-pdf://semisupervised-3254828305/semisupervised.ppt%0Ahttp://dx.doi.org/10.1016/j.str.2013.02.005%0Ahttp://dx.doi.org/10.10

Dermawan, O. (2018). Strategi Pembelajaran Bagi Anak Berkebutuhan Khusus Di Slb. *Psympathic : Jurnal Ilmiah Psikologi*, *6*(2), 886–897. https://doi.org/10.15575/psy.v6i2.2206

Erlin, Sianturi, J., Hajjah, A., & Agustin. (2021). Analisis Sentimen Prosesor AMD Ryzen menggunakan Metode Support Vector Machine. *SATIN - Sains Dan Teknologi Informasi*, *7*(2), 129–141. https://doi.org/10.33372/stn.v7i2.804

Fedianto, M. H. S., Aditiawan, F. P., & Haromainy, M. M. Al. (2024). *Pengujian Sistem Jaringan Dokumentasi Dan Informasi Menggunakan Black Box Testing Dan White Box Testing*. *3*(1).

Grandis, G. F., Arumsari, Y., & Indriati. (2021). Seleksi Fitur Gain Ratio pada Analisis Sentimen Kebijakan Pemerintah Mengenai Pembelajaran Jarak Jauh dengan K-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *5*(8), 3507–3514.

Hafiar, H., Setianti, Y., Subekti, P., & Sani, A. (2020). *BLIND CODE PADA UANG KERTAS RUPIAH PESAN KOMUNIKASI DAN KOMUNIKASI PESAN KEPADA PUBLIK DISABILITAS NETRA*. *5415*.

Ibrahim, M. M., Rahmadewi, R., & Nurpulaela, L. (2023). Pendeteksian Nominal Uang Pada Gambar Menggunakan Convolutional Neural Network: Integrasi Metode Pra-Pemrosesan Citra Dan Klasifikasi Berbasis Cnn. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *7*(2), 1394–1400. https://doi.org/10.36040/jati.v7i2.6863

Ikhsan, & Sari, P. P. (2018). Sistem Pendeteksi Nominal Dan Keaslian Uang Kertas Rupiah Untuk Penyandang Tuna Netra Berbasis Arduino. *Jurnal Ilmiah Informatika*, *6*(02), 10.

Kemenkes. (2021). *Katarak Penyebab Terbanyak Gangguan Penglihatan di Indonesia*. https://sehatnegeriku.kemkes.go.id/baca/umum/20211012/5738714/katara k-penyebab-terbanyak-gangguan-penglihatan-di-indonesia/

Kurniyanti, V. A., & Murdiani, D. (2022). PERBANDINGAN MODEL WATERFALL DENGAN PROTOTYPE PADA PENGEMBANGAN SYSTEM INFORMASI BERBASIS WEBSITE. *JURNAL FUSION*, *2*(8.5.2017), 2003–2005.

Lee, S. Y., Gurnani, B., & Mesfin, F. B. (2017). *Blindness*. https://www.ncbi.nlm.nih.gov/books/NBK448182/

Nurhidayat, R., & Dewi, K. E. (2023). Penerapan Algoritma K-Nearest Neighbor Dan Fitur Ekstraksi N-Gram Dalam Analisis Sentimen Berbasis Aspek. *Komputa : Jurnal Ilmiah Komputer Dan Informatika*, *12*(1), 91–100. https://doi.org/10.34010/komputa.v12i1.9458

Restuputri, D. P., Mas'udin, I., Fadholi, N. M. S., & Wijaya, Y. A. (2018). Alat Bantu Mengenali Uang Kertas Dengan Kasentra (Kacamata Sensor Tunanetra). *Seminar Dan Konferensi Nasional IDEC*, 7–8.

Saputra, A. N., Handayani, H. H., Sukmawati, C. E., & Siregar, A. M. (2024). Model Klasifikasi Nominal Mata Uang Kertas Republik Indonesia Menggunakan Convolutional Neural Network. *Journal of Information System Research (JOSH)*, *6*(1), 176–184. https://doi.org/10.47065/josh.v6i1.5927

Suharyono, F. W. G., Kartini, & Junaidi, A. (2024). *PENERAPAN METODE BOUNDARY VALUE ANALYSIS DAN EQUIVALENCE PARTITIONING DALAM PENGUJIAN BLACK BOX UNTUK APLIKASI SIADITA*. *8*(1), 1013–1020.