

**METODE *CONVOLUTIONAL NEURAL NETWORK* (CNN)
UNTUK MENDETEKSI NOMINAL UANG KERTAS RUPIAH
SEBAGAI ALAT BANTU TUNANETRA PADA APLIKASI
BERBASIS *MOBILE***

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan mendapatkan gelar Strata
Satu Program Studi Informatika**



**Disusun oleh :
ROBBI SHURNII AL'AQIILAH
22EO10008**

**PROGRAM STUDI INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU KOMPUTER
UNIVERSITAS NAHDLATUL ULAMA AL GHAZALI
CILACAP
2026**

**METODE *CONVOLUTIONAL NEURAL NETWORK* (CNN)
UNTUK MENDETEKSI NOMINAL UANG KERTAS RUPIAH
SEBAGAI ALAT BANTU TUNANETRA PADA APLIKASI
BERBASIS *MOBILE***

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan mendapatkan gelar Strata
Satu Program Studi Informatika**



**Disusun oleh :
ROBBI SHURNII AL'AQILAH
22EO10008**

**PROGRAM STUDI INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU KOMPUTER
UNIVERSITAS NAHDLATUL ULAMA AL GHAZALI
CILACAP
2026**

HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya menyatakan bahwa penulisan skripsi dengan judul “Metode *Convolutional Neural Network* (CNN) Untuk Mendeteksi Nominal Uang Kertas Rupiah Sebagai Alat Bantu Tunanetra Pada Aplikasi Berbasis *Mobile*” adalah hasil karya saya dengan arahan dari pembimbing dan belum diajukan kepada pihak manapun. Sumber informasi yang dikutip dalam skripsi ini telah dicantumkan dalam Daftar Pustaka.

Demikian pernyataan ini dibuat dengan sebenarnya. Apabila di kemudian hari terdapat ketidaksesuaian dalam pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.



LEMBAR PENGESAHAN

Skripsi Saudara

Nama : Robbi Shurnii Al'Aqilah
NIM : 22EO10008
Fakultas/Prodi : Fakultas MIKOM / Prodi Informatika
Judul : Metode *Convolutional Neural Network* (CNN) untuk Mendeteksi Nominal Uang Kertas Rupiah sebagai Alat Bantu Tunanetra pada Aplikasi Berbasis *Mobile*

Telah disidangkan oleh Dewan Penguji Fakultas Matematika dan Ilmu Komputer Universitas Nahdlatul Ulama Al Ghazali (UNUGHA) Cilacap pada hari / tanggal :
Jumat, 3 Juli 2026

Dan dapat diterima sebagai pemenuhan tugas akhir mahasiswa Program Strata Satu (S.1) Program Studi Informatika (S.Kom.) Fakultas Matematika dan Ilmu Komputer (FMIKOM) pada Universitas Nahdlatul Ulama Al Ghazali (UNUGHA) Cilacap.

Cilacap, 9 Juli 2026

Dewan Sidang

Ketua

Tri Anggoro, M.Kom.
NIDN. 0601128103

Sekretaris

Abdul Haq, S.E., M.Cs.
NIDN. 0606067701

Penguji 1

Tri Anggoro, M.Kom.
NIDN. 0601128103

Penguji 2

Verry, S.T., M.Kom.
NIDN. 0628068001

Pembimbing I

Abdul Haq, S.E., M.Cs.
NIDN. 0606067701

Pembimbing II

Koko Edy Yulianto, M.Kom.
NIP. 230714144

Mengetahui,

Dekan Fakultas Matematika dan Ilmu Komputer



M. A. Aziz Zein S.Si.M.Kom.
NIDN. 2125098601

HALAMAN NOTA KONSULTAN

NOTA KONSULTAN

Hal : Skripsi Saudari Robbi Shurnii Al'Aqilah
Lampiran : -

Kepada
Yth. Dekan Fakultas Matematika dan Ilmu Komputer
Universitas Nahdlatul Ulama Al Ghazali Cilacap
di Cilacap

Assalamu'alaikum Wr. Wb.

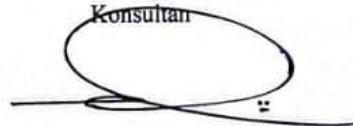
Setelah membaca, memeriksa dan melakukan perbaikan seperlunya maka skripsi saudara/i:

Nama : Robbi Shurnii Al'Aqilah
NIM : 22EO10008
Fakultas / Prodi : Fakultas Matematika dan Ilmu Komputer / Prodi Informatika
Judul : Metode *Convolutional Neural Network* (CNN) Untuk Mendeteksi Nominal Uang Kertas Rupiah Sebagai Alat Bantu Tunanetra Pada Aplikasi Berbasis *Mobile*

Dapat diajukan ke Fakultas Matematika dan Ilmu Komputer, Universitas Nahdlatul Ulama Al Ghazali Cilacap untuk memenuhi syarat memperoleh gelar Strata Satu (S1).

Wassalamu'alaikum Wr. Wb.

Cilacap, 9 Juli 2026
Konsultan



Tri Anggoro, M.Kom
NIDN. 0601128103

HALAMAN NOTA PEMBIMBING

NOTA PEMBIMBING

Cilacap, 24 Juni 2026

Kepada Yth :
Fakultas Matematika dan Komputer (FMIKOM)
UNUGHA Cilacap
di Cilacap

Assalamu'alaikum Wr. Wb.

Setelah melakukan bimbingan, telaah, arahan dan koreksi tahap penulisan skripsi
saudari:

Nama : Robbi Shurnii Al'Aqilah
NIM : 22EO10008
Fakultas / Prodi : Fakultas Matematika dan Ilmu Komputer / Prodi
Informatika
Judul : Metode *Convolutional Neural Network* (CNN) Untuk
Mendeteksi Nominal Uang Kertas Rupiah Sebagai Alat
Bantu Tunanetra Pada Aplikasi Berbasis *Mobile*

Kami berpendapat bahwa skripsi tersebut sudah dapat diajukan ke sidang skripsi.
Bersamaan ini kami kirimkan skripsi tersebut, semoga dapat segera disidangkan.

Atas perhatiannya kami ucapkan terima kasih.

Wassalamu'alaikum Wr. Wb.

Mengetahui,

Pembimbing I



Abdul Haq, S.E., M.Cs.
NIDN. 0606067701

Pembimbing II



Koko Edy Yulianto, M.Kom.
NIP. 23071400144

HALAMAN MOTO

“seumur hidup bakal dihadapi sama kegagalan - kegagalan lain. kalau tetap takut dan tidak mau mencoba, bakal hidup seumur hidup sebagai pecundang.”

(yukatheo)

“doakan usahamu, usahakan doamu.”

(maspheyy)

“rencana hanya kanvas, niat adalah kuasnya, dan tinta nyalinya”

(yb)

HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT, Rabb semesta alam yang senantiasa memberikan karunia sehingga penulis mampu menyelesaikan penulisan skripsi ini. Karya ini saya persembahkan kepada :

1. Almarhum Bapak Ismail, S.T., sosok yang paling dirindukan. Berat sekali rasanya ditinggalkan sedari kecil. Terima kasih atas segala pengorbanan, kasih sayang, serta segala bentuk tanggung jawab atas kehidupan layak yang telah diberikan semasa bapak hidup. Terima kasih telah menjadi alasan terbesar untuk tetap semangat berjuang meraih gelar sarjana yang bapak impikan. Walaupun berat sekali harus melewati kerasnya kehidupan tanpa didampingi sosok bapak, rasa iri dan rindu sering kali membuat terjatuh. Semoga bapak bangga dengan usaha dan perjuangan selama ini.
2. Ibu Listianti tercinta dan tersayang, yang telah mengorbankan banyak waktu, tenaga, dan upaya untuk berjuang demi kehidupan anak – anaknya, menggantikan peran bapak menjadi suatu kebanggaan memiliki orang tua hebat yang selalu mendukung anak – anaknya untuk mencapai cita – cita. Beliau memang tidak sempat merasakan pendidikan sampai bangku perkuliahan, namun beliau mampu mendidik dan memotivasi tiada henti hingga anaknya mampu menyelesaikan studinya sampai sarjana.
3. Kakak kandung saya Khoirunnisa ‘Fathimatunnada dan adik kandung saya Janeeta Zain, yang telah memberikan doa, kasih sayang, dukungan, semangat, serta menjadi penyemangat sekaligus pengingat agar skripsi ini segera diselesaikan. Terima kasih karena selalu ada untuk mendengarkan keluh kesah, memberikan motivasi, menghibur di saat lelah, dan sesekali membelikkan segelas americano untuk menemani mengerjakan skripsi.
4. Bapak Abdul Haq, S.E., M.Cs dan Bapak Koko Edy Yulianto, M.Kom selaku dosen pembimbing yang selalu meluangkan waktu, memberikan arahan, serta membimbing dengan penuh kesabaran dan ketulusan. Semoga segala kebaikan yang diberikan dibalas dengan sebaik – baiknya oleh Allah SWT.
5. Untuk sahabatku Faisya, Rheika Mosa, dan Bela yang selalu ada dalam setiap cerita, berbagai tawa, keluh, dan saling sharing dalam perjalanan ini.

Kebersamaan yang sederhana namun penuh makna menjadi bagian yang tak tergantikan.

6. Teman – teman angkatan 22 yang selalu menjadi bagian dari perjalanan perkuliahan ini, berbagai cerita di ruang kelas, bertukar pikiran dalam tugas dan skripsi, serta saling menguatkan di setiap proses yang tidak selalu mudah.
7. Teman – teman HIMAFORTIC yang selalu menjadi tempat belajar, berkembang, dan berbagai pengalaman di luar ruang kelas. Setiap kebersamaan, kerja sama, dan cerita yang terjalin memberikan warna serta perjalanan berharga dalam perjalanan ini.
8. Teman – teman KKN Kelompok 12 Desa Pengadegan Majenang yang selalu berbagi cerita, pengalaman, dan kebersamaan dalam setiap proses pengabdian, serta yang suka mengajak nongkrong sekedar ngopi di sela – sela kesibukan.
9. Live streaming Maraphthon dan AAA Clan yang selalu menjadi hiburan di saat suntuk mengerjakan skripsi, menghadirkan suasana santai dan obrolan yang terasa seperti ikut nongkrong langsung.
10. Pemilik NIM 22AB10062 yang hadir tanpa rencana, namun selalu memberikan dukungan di setiap keadaan. Dengan caranya sendiri mampu menghadirkan kebahagiaan, bahkan di saat hal – hal terasa tidak mudah. Tanpa banyak kata, kehadirannya menjadi hal sederhana yang begitu berarti.
11. Diri saya sendiri, yang telah berjuang dalam diam, melewati banyak hal yang tidak selalu mudah, namun tetap memilih untuk bertahan. Yang tidak menyerah meski lelah, dan tetap melangkah meski ragu. Semua proses ini adalah bukti bahwa saya mampu dan akan selalu mampu untuk terus melangkah ke depan.

KATA PENGANTAR

Puji syukur kepada Allah SWT atas segala limpahan nikmat dan karuniaNya, sehingga penulis dapat menyelesaikan penulisan skripsi. Sholawat dan salam senantiasa penulis haturkan kepada Nabi Muhammad SAW sebagai pembimbing seluruh umat manusia.

Skripsi ini tidak akan selesai tanpa adanya bantuan dari banyak pihak, karena itu penulis menyampaikan terima kasih kepada :

1. Bapak Mochamad Taufiqurrochman Abdul Aziz Zein, M.Kom, Dekan Fakultas Matematika dan Ilmu Komputer (FMIKOM) Universitas Nahdlatul Ulama Al Ghazali Cilacap (UNUGHA).
2. Bapak Abdul Haq, S.E., M.Cs., pembimbing I yang telah dengan sabar memberikan bimbingan dan petunjuk dalam menyelesaikan skripsi ini.
3. Bapak Koko Edy Yulianto, M.Kom., pembimbing II yang telah dengan sabar memberikan bimbingan dan petunjuk dalam menyelesaikan skripsi ini.
4. Bapak-Ibu dosen Program Studi Informatika Fakultas Matematika dan Ilmu Komputer Universitas Nahdlatul Ulama Al Ghazali Cilacap.
5. Semua pihak yang tidak dapat kami sebutkan satu persatu

Semoga Allah SWT membalas jerih payah dan pengorbanan yang telah diberikan dengan balasan yang lebih baik. Amiin. Penulis berharap semoga karya kecil ini bermanfaat bagi pembaca.

Cilacap, Juni 2026

Penulis

ABSTRAK

ROBBI SHURNII AL'AQIILAH. Metode *Convolutional Neural Network* (CNN) untuk Deteksi Nominal Uang Kertas Rupiah Sebagai Alat Bantu Tunanetra Menggunakan Aplikasi *Mobile*. Dibimbing oleh Abdul Haq, S.E., M.Cs. dan Koko Edy Yulianto, M.Kom.

Penyandang tunanetra sering mengalami kesulitan dalam mengenali nominal uang kertas rupiah saat melakukan transaksi sehari – hari. Penelitian ini bertujuan untuk mengembangkan aplikasi mobile berbasis android yang dapat mendeteksi nominal uang kertas rupiah menggunakan metode *Convolutional Neural Network* (CNN) serta memberikan hasil deteksi dalam bentuk suara.

Metode yang digunakan adalah CNN dengan arsitektur *MobileNetV2*. Dataset terdiri dari 2.400 citra uang kertas rupiah emisi tahun 2022 yang mencakup nominal Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, Rp 100.000, dan kelas bukan uang. Model dikembangkan menggunakan TensorFlow, kemudian dikonversi ke format TensorFlow Lite dan diintegrasikan ke dalam aplikasi android berbasis Flutter yang dilengkapi fitur *Text-to-Speech* (TTS).

Hasil pengujian menggunakan *Confusion Matrix* pada 900 data testing menunjukkan tingkat akurasi sebesar 98.89%. Sistem mampu mendeteksi nominal uang dengan baik pada jarak 10 – 30 cm dan kondisi pencahayaan di atas 100 lux. Hasil penelitian menunjukkan bahwa metode CNN mampu diterapkan secara efektif untuk mendeteksi nominal uang kertas rupiah dan membantu penyandang tunanetra mengenali uang secara mandiri.

Kata kunci : *Convolutional Neural Network* (CNN), Deteksi Nominal Uang, Aplikasi *Mobile*, Pengolahan Citra Digital, Tunanetra.

ABSTRACT

ROBBI SHURNII AL'AQIILAH. Convolutional Neural Network (CNN) Method for Detecting the Denomination of Rupiah Banknotes as an Aid for the Visually Impaired Using a Mobile Application. Supervised by Abdul Haq, S.E., M.Cs. and Koko Edy Yulianto, M.Kom.

Visually impaired individuals often have difficulty recognizing the denominations of Indonesian banknotes during daily transactions. This research aims to develop an android-based mobile application that can detect the denomination of Indonesian banknotes using the Convolutional Neural Network (CNN) method and provide detection results in the form of sound.

The method used is CNN with the MobileNetV2 architecture. The dataset consists of 2.400 images of 2022 emission Indonesia banknotes, including denominations of Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, Rp 100.000, and a non-currency class. The model was developed using TensorFlow, then converted to TensorFlow Lite format and integrated into an android application based on Flutter equipped with Text-to-Speech (TTS) features.

The results of the testing using the Confusion Matrix on 900 testing data show an accuracy rate of 98.89%. The system is able to detect the nominal value of money well at a distance of 10 – 30 cm and lighting conditions above 100 lux. The research results indicate that the CNN method can be effectively applied to detect the nominal value of Indonesian banknotes and help visually impaired individuals recognize money independently.

Keywords : Convolutional Neural Network (CNN), Currency Denomination Detection, Mobile Application, Digital Image Processing, Blind.

DAFTAR ISI

HALAMAN PERNYATAAN KEASLIAN SKRIPSI	Error! Bookmark not defined.
LEMBAR PENGESAHAN	Error! Bookmark not defined.
HALAMAN NOTA KONSULTAN.....	Error! Bookmark not defined.
HALAMAN NOTA PEMBIMBING.....	Error! Bookmark not defined.
HALAMAN MOTO	vii
HALAMAN PERSEMBAHAN	viii
KATA PENGANTAR	x
ABSTRAK	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL.....	xvi
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Rumusan Masalah	2
C. Batasan Masalah.....	2
D. Tujuan Penelitian	3
E. Manfaat Penelitian	3
BAB II TINJAUAN PUSTAKA.....	4
A. Penelitian Terkait	4
B. Landasan Teori.....	8
a. Convolutional Neural Network (CNN).....	9
b. Pengolahan Citra Digital	9
c. <i>Machine Learning</i> dan <i>Deep Learning</i>	10
d. Uang Kertas Rupiah	10
e. Aplikasi Mobile.....	10
f. Android	10
g. Text-to-Speech (TTS)	10
BAB III METODOLOGI.....	12
A. Waktu dan Tempat Penelitian	12
B. Prosedur Penelitian.....	12
a. Identifikasi Masalah	12

b. Studi Literatur	13
c. Analisa Kebutuhan	13
d. Pengumpulan Dataset.....	14
e. Desain.....	14
f. Pengembangan Aplikasi Mobile	23
g. Pengujian Sistem.....	24
h. Analisis Hasil	30
C. Jadwal Penelitian.....	30
BAB IV HASIL DAN PEMBAHASAN	31
A. Hasil	31
a. Implementasi Sistem	31
b. Hasil Pengujian Model.....	38
c. Hasil Pengujian Akurasi Model	40
d. Hasil Pengujian Efektivitas Sistem	49
B. Pembahasan.....	51
a. Pengujian Fungsional Sistem	51
b. Analisis Performa Model CNN.....	53
BAB V KESIMPULAN DAN SARAN.....	57
A. Kesimpulan	57
B. Saran.....	57
DAFTAR PUSTAKA	58
LAMPIRAN.....	61

DAFTAR GAMBAR

Gambar 1 Mindmap Penelitian	8
Gambar 2 Arsitektur CNN	9
Gambar 3 Flowchart Penelitian.....	12
Gambar 4 Diagram Use Case	15
Gambar 5 Diagram Activity Splash	16
Gambar 6 Diagram Activity Masuk.....	17
Gambar 7 Diagram Activity Dashboard Scan.....	17
Gambar 8 Diagram Activity Panduan	18
Gambar 9 Diagram Activity Pengaturan.....	18
Gambar 10 Diagram Sequence Masuk.....	19
Gambar 11 Diagram Sequence Dashboard	19
Gambar 12 Diagram Sequence Panduan.....	20
Gambar 13 Diagram Sequence Pengaturan.....	20
Gambar 14 Mockup Splash.....	21
Gambar 15 Mockup Mulai	21
Gambar 16 Mockup Dashboard	21
Gambar 17 Mockup Scan.....	21
Gambar 18 Mockup Hasil Scan	22
Gambar 19 Mockup Panduan.....	22
Gambar 20 Mockup Pengaturan.....	22
Gambar 21 Mockup Informasi Aplikasi	22
Gambar 22 Mockup Tentang Kami	23
Gambar 23 Gantt Chart Jadwal Penelitian	30
Gambar 24 Confusion Matrix	41
Gambar 25 Classification Report	48
Gambar 26 Grafik Akurasi Model	55
Gambar 27 Grafik Loss Model	56

DAFTAR TABEL

Tabel 1 Penelitian Terkait	4
Tabel 2 Kebutuhan Perangkat Keras	13
Tabel 3 Kebutuhan Perangkat Lunak	13
Tabel 4 Pengujian Menu Splash	24
Tabel 5 Pengujian Menu Masuk	24
Tabel 6 Pengujian Menu Dashboard	24
Tabel 7 Pengujian Menu Panduan	25
Tabel 8 Pengujian Menu Pengaturan	25
Tabel 9 Pengujian Menu Informasi Aplikasi	25
Tabel 10 Pengujian Menu Tentang Kami	25
Tabel 11 Pengujian Sistem Berdasarkan Jarak	26
Tabel 12 Pengujian Sistem Berdasarkan Cahaya	27
Tabel 13 Pengujian Akurasi Model	29
Tabel 14 Hasil Pengujian Model Berdasarkan Jarak dan Cahaya	38
Tabel 15 Hasil Pengujian Model Berdasarkan Nominal	39
Tabel 16 Accuracy	42
Tabel 17 Hasil Pengujian Efektivitas	50

BAB I PENDAHULUAN

A. Latar Belakang

Uang adalah alat pembayaran yang resmi dan sah untuk melakukan jual beli. Uang dibedakan menjadi 3 jenis yaitu uang kertas, uang koin, dan uang digital. Uang kertas rupiah memiliki ciri khas tersendiri, seperti warna, benang pengaman, logo Bank Indonesia, *watermark*, dan lain – lain. Sedangkan, uang koin hanya dapat dibedakan dari diameternya dan bahan yang digunakan untuk membuat uang koin tersebut. Adapun uang digital merupakan bentuk uang yang tidak memiliki wujud fisik dan digunakan dalam transaksi elektronik melalui perangkat seperti *smartphone*, misalnya melalui *e-wallet* dan *mobile banking*[1].

Uang kertas rupiah adalah jenis uang yang berbentuk lembaran, terbuat dari bahan kertas atau bahan yang mirip kertas, yang dikeluarkan oleh pemerintah Indonesia. Uang kertas rupiah asli dapat dikenali berdasarkan ciri khas nya. Setiap karakter pada pecahan uang kertas atau pada proses pembuatannya, memiliki fungsi untuk membedakan satu pecahan dengan pecahan lainnya, serta berperan sebagai pengaman agar uang tersebut tidak dipalsukan. Namun, dalam praktiknya tidak semua orang dapat dengan mudah mengenali ciri – ciri tersebut, terutama bagi penyandang tunanetra[2].

Teknologi informasi berkembang dengan sangat cepat, dengan banyak penemuan baru yang menawarkan solusi dan kemudahan. Inovasi dan solusi ditujukan untuk semua orang, terutama mereka yang memiliki keterbatasan atau penyandang disabilitas. Namun, dalam kenyataannya tidak semua teknologi mampu menjawab kebutuhan dasar mereka secara optimal. Dalam beraktivitas, penyandang disabilitas, terutama tunanetra, memerlukan alat bantu. Penyandang tunanetra sulit menemukan dan membedakan uang, terutama saat melakukan transaksi jual beli yang membutuhkan uang[3].

Kondisi ini menunjukkan adanya permasalahan yang nyata karena hingga saat ini penyandang tunanetra masih mengandalkan metode konvensional, seperti meminta bantuan orang lain atau meraba tanda timbul pada uang. Metode tersebut memiliki keterbatasan karena bergantung pada orang lain menimbulkan risiko penipuan dan tanda fisik dapat rusak[4]. Kecuali uang tersebut masih sangat baru, rata, dan dapat dibandingkan dengan yang lain, para penyandang tunanetra sulit untuk mengetahui nilai uang[5].

Dalam hal daya ingat, mereka yang tunanetra mungkin mengalami kesulitan mengingat nominal atau kode karena mereka menggunakan uang yang sudah rusak dari tangan ke tangan[6]. Untuk membantu penyandang tunanetra, diperlukan pengembangan sistem deteksi nominal uang berdasarkan keterbatasan tersebut. Sistem ini akan dapat membedakan nominal uang menggunakan gambar dan menghasilkan suara[7].

Penelitian ini menggunakan metode *Convolutional Neural Network* (CNN) sebagai algoritma utama dalam proses pengenalan citra uang kertas rupiah. CNN merupakan salah satu metode dalam bidang *Deep Learning* yang dirancang khusus untuk mengolah data berbentuk citra atau gambar. Metode ini banyak digunakan pada

berbagai penelitian di bidang *computer vision* karena memiliki kemampuan yang sangat baik dalam mengenali objek, mendeteksi pola, serta melakukan klasifikasi citra dengan tingkat akurasi yang tinggi. Dibandingkan dengan metode pengolahan citra konvensional yang masih memerlukan proses ekstraksi fitur secara manual, CNN mampu mempelajari karakteristik suatu objek secara otomatis melalui proses pelatihan menggunakan data dalam jumlah besar. Oleh karena itu, CNN menjadi salah satu pendekatan yang paling populer dan banyak diterapkan dalam berbagai sistem pengenalan objek, termasuk pengenalan nominal uang kertas rupiah[8].

CNN tersusun atas beberapa layer yang bekerja secara berurutan, seperti *convolution layer*, *activation layer*, *pooling layer*, dan *fully connected layer*. Keunggulan utama CNN terletak pada kemampuannya dalam melakukan ekstraksi fitur secara otomatis. Hal ini memungkinkan sistem untuk mempelajari sendiri ciri – ciri yang membedakan setiap objek tanpa memerlukan proses perancangan fitur secara manual. Kemampuan tersebut menjadikan CNN lebih adaptif terhadap variasi bentuk, warna, tekstur, maupun pencahayaan pada citra dibandingkan metode klasifikasi tradisional. Selain itu, CNN juga mampu menghasilkan performa yang baik dalam mengenali objek meskipun terdapat sedikit perubahan posisi, orientasi, maupun kondisi lingkungan saat proses pengambilan gambar[9].

Secara singkat, CNN adalah teknik pengolahan gambar yang menggunakan konvolusi untuk menemukan fitur khusus pada data pembelajaran dan kemudian menggunakan fitur tersebut sebagai input untuk melakukan pembelajaran dengan *Neural Network*. Dengan demikian, komputer memiliki kemampuan untuk mengklasifikasi uang agar dapat dikenali sesuai nominalnya berdasarkan fitur khusus, yang berarti bahwa uang dapat dikenali dengan benar[10].

Tujuan dari penelitian ini adalah untuk membuat dan membangun suatu aplikasi yang akan membantu tunanetra mengenali dan membedakan berbagai uang kertas senilai Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, dan Rp 100.000 yang diterbitkan oleh Bank Indonesia tahun emisi 2022.

B. Rumusan Masalah

Rumusan masalah pada penelitian ini sebagai berikut :

- a. Bagaimana merancang dan mengimplementasikan aplikasi deteksi nominal uang kertas rupiah berbasis CNN pada perangkat *mobile*?
- b. Bagaimana menguji efektivitas dan akurasi sistem dalam mendeteksi uang secara real-time dengan output audio?

C. Batasan Masalah

Batasan masalah yang diambil pada penelitian ini sebagai berikut :

- a. Sistem yang dikembangkan hanya mendeteksi dan mengklasifikasikan nominal uang kertas rupiah emisi tahun 2022.
- b. Nominal uang yang digunakan dalam penelitian dibatasi pada beberapa pecahan tertentu, seperti Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, dan Rp 100.000.

- c. Sistem diimplementasikan pada perangkat Android dengan kamera sebagai media input dan suara atau audio yang menyebutkan nominal uang yang terdeteksi sebagai output.
- d. Proses pengambilan citra dilakukan menggunakan kamera bawaan standar pada perangkat Android tanpa menggunakan perangkat tambahan atau sensor khusus.

D. Tujuan Penelitian

Tujuan penelitian yang diinginkan peneliti dalam penelitian yaitu sebagai berikut :

- a. Merancang dan mengimplementasikan aplikasi deteksi nominal uang kertas rupiah berbasis *Convolutional Neural Network* (CNN) pada perangkat *mobile*.
- b. Menguji efektivitas dan tingkat akurasi sistem dalam mendeteksi nominal uang kertas secara real – time serta menghasilkan output audio yang sesuai.

E. Manfaat Penelitian

- 1. Bagi Teoritis (bagi peneliti)
 - a. Menambah wawasan dan pemahaman dalam penerapan metode *Convolutional Neural Network* (CNN) pada bidang pengolahan citra digital.
 - b. Meningkatkan kemampuan dalam mengembangkan aplikasi *mobile* berbasis *machine learning*.
 - c. Memberikan pengalaman dalam merancang, mengimplementasikan, dan menguji sistem deteksi berbasis kecerdasan buatan secara real – time.
- 2. Praktis (bagi pengguna)
 - a. Membantu penyandang tunanetra dalam mengenali nominal uang kertas secara mandiri melalui output audio.
 - b. Meningkatkan kemudahan dan kepercayaan diri pengguna dalam melakukan transaksi sehari – hari.
 - c. Memberikan solusi teknologi yang praktis, cepat, dan mudah digunakan dalam mendeteksi nominal uang kertas.

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

Penelitian terkait yang digunakan sebagai referensi berjarak lima tahun terhitung ke belakang dari tahun penelitian saat ini. Tabel 1 merupakan tabel penelitian terkait sebagai berikut :

Tabel 1 Penelitian Terkait

No.	Nama (Tahun)	Keterangan
(1)	(2)	(3)
1.	Maulana Azhar K, Santoso I, Yosua D, Soetrisno A (2021)	Judul : Implementasi Deep Learning Menggunakan Metode Convolutional Neural Network dan Algoritma YOLO Dalam Sistem Pendeteksi Uang Kertas Rupiah Bagi Penyandang Low Vision Metode : Convolutional Neural Network (CNN), algoritma YOLO Hasil Penelitian : Pada perhitungan nilai mAP sistem berdasarkan keseluruhan pengujian yang meliputi pengujian dengan berbagai variasi kondisi uang (seluruh bagian, bagian kiri, dan bagian kanan) serta penghunian pada variasi kondisi cahaya redup mencapai nilai mAP sebesar 88%. Rata – rata waktu respon program untuk dapat mendeteksi uang sebesar 1,28 detik.
2.	I Made Gende Indra Mahendra (2022)	Judul : Pengembangan Model Pembelajaran Mesin untuk Mengidentifikasi Nominal Uang Kertas Rupiah Menggunakan Algoritma CNN Dengan Library Keras Metode : Convolutional Neural Network (CNN) Hasil Penelitian : Algoritma CNN berbasis Keras berhasil digunakan untuk mengidentifikasi nominal uang rupiah dengan preprocessing yang tepat. Model arsitektur C2 memberikan hasil terbaik dengan prediksi paling akurat serta nilai precision, recall, dan F1-score tertinggi, serta dilengkapi GUI sederhana untuk memudahkan penggunaan.

No.	Nama (Tahun)	Keterangan
(1)	(2)	(3)
3.	Hermawan A, Lianata L, Junaedi, Maranto A (2022)	Judul : Implementasi Machine Learning Sebagai Pengenal Nominal Uang Rupiah dengan Metode YOLOv3 Metode : Aplikasi <i>Mobile</i>, YOLOv3 Hasil Penelitian : Menunjukkan hasil yang baik dengan akurasi hingga 98% pada kondisi pencahayaan yang optimal dan diimplementasikan dalam aplikasi Android yang mampu mendeteksi nominal uang secara real-time serta menghasilkan output suara. Namun, kinerjanya menurun pada kondisi pencahayaan rendah, dengan akurasi sekitar 77%, sehingga masih bergantung pada kualitas cahaya saat proses deteksi.
4.	Ayu P, Zakaria L, Zainal M (2023)	Judul : Pengenalan Nilai Mata Uang Kertas Untuk Tunanetra Berbasis Android Metode : Aplikasi <i>Mobile</i>, Template Matching Hasil Penelitian : Aplikasi android berhasil mendeteksi nominal uang Rp 1.000 – Rp 100.000 secara realtime dengan output suara untuk tunanetra. Namun, akurasi sangat dipengaruhi pencahayaan dan jarak kamera, pada cahaya tinggi dan jarak 200 cm deteksi gagal dengan akurasi rendah.
5.	Daniel Yc Raban R, Arnie R (2023)	Judul : Model Pendeteksi Nominal Uang Kertas Rupiah Menggunakan Teknologi Optical Character Recognition Metode : Aplikasi <i>Mobile</i>, Optical Character Recognition (OCR), algoritma Regular Expression Hasil Penelitian : Teknologi OCR berbasis Regular Expression berhasil mendeteksi nominal uang kertas dengan akurasi 100% pada 210 sampel. Sistem berpotensi dikembangkan secara komersial untuk membantu tunanetra, namun masih perlu pengujian pada uang logam, uang lusuh, dan pengguna low vision.

No.	Nama (Tahun)	Keterangan
(1)	(2)	(3)
6.	Malik Ibrahim M, Rahmadewi R, Nurpulaela L (2023)	Judul : Pendeteksian Nominal Uang Pada Gambar Menggunakan <i>Convolutional Neural Network</i>: Integrasi Metode Pra-Pemrosesan Citra dan Klasifikasi Berbasis CNN Metode : <i>Convolutional Neural Network</i> (CNN) Hasil Penelitian : Pengujian CNN pada 10 sampel per nominal menunjukkan hasil bervariasi, tertinggi Rp 50.000 (100%) dan Rp 100.000 (90%), namun beberapa nominal di bawah 50% sehingga dinilai kurang berhasil. Ketidakefisienan diduga akibat proses pelabelan manual yang kurang optimal.
7.	Muhammad Nur Hidayat A, Zakiyah I (2023)	Judul : Identifikasi Nominal Mata Uang Rupiah Bagi Penyandang Tunanetra Dengan Algoritma <i>Convolutional Neural Network</i> Berbasis Andorid Metode : <i>Convolutional Neural Network</i> (CNN), VGG-16 Hasil Penelitian : Penelitian menggunakan metode CNN (VGG-16) berbasis aplikasi android mampu mendeteksi nominal uang rupiah secara real-time dengan akurasi sekitar 83%, namun masih terdapat beberapa kesalahan deteksi akibat perbedaan nilai confidence sehingga hasil belum sepenuhnya akurat.
8.	Khairu D, Wantasen F, Yusupa A, Taringan V (2025)	Judul : Klasifikasi Jenis Uang Kertas Menggunakan <i>Convolutional Neural Network</i> Metode : <i>Convolutional Neural Network</i> (CNN), arsitektur VGG-16 Hasil Penelitian : Model CNN berhasil mengklasifikasikan nominal uang kertas dengan akurasi 100% pada epoch ke-13 dan loss yang terus menurun. Model menunjukkan kemampuan generalisasi yang baik dan berpotensi diterapkan pada aplikasi pengenalan uang, namun tetap memerlukan pengujian lebih lanjut pada kondisi nyata.

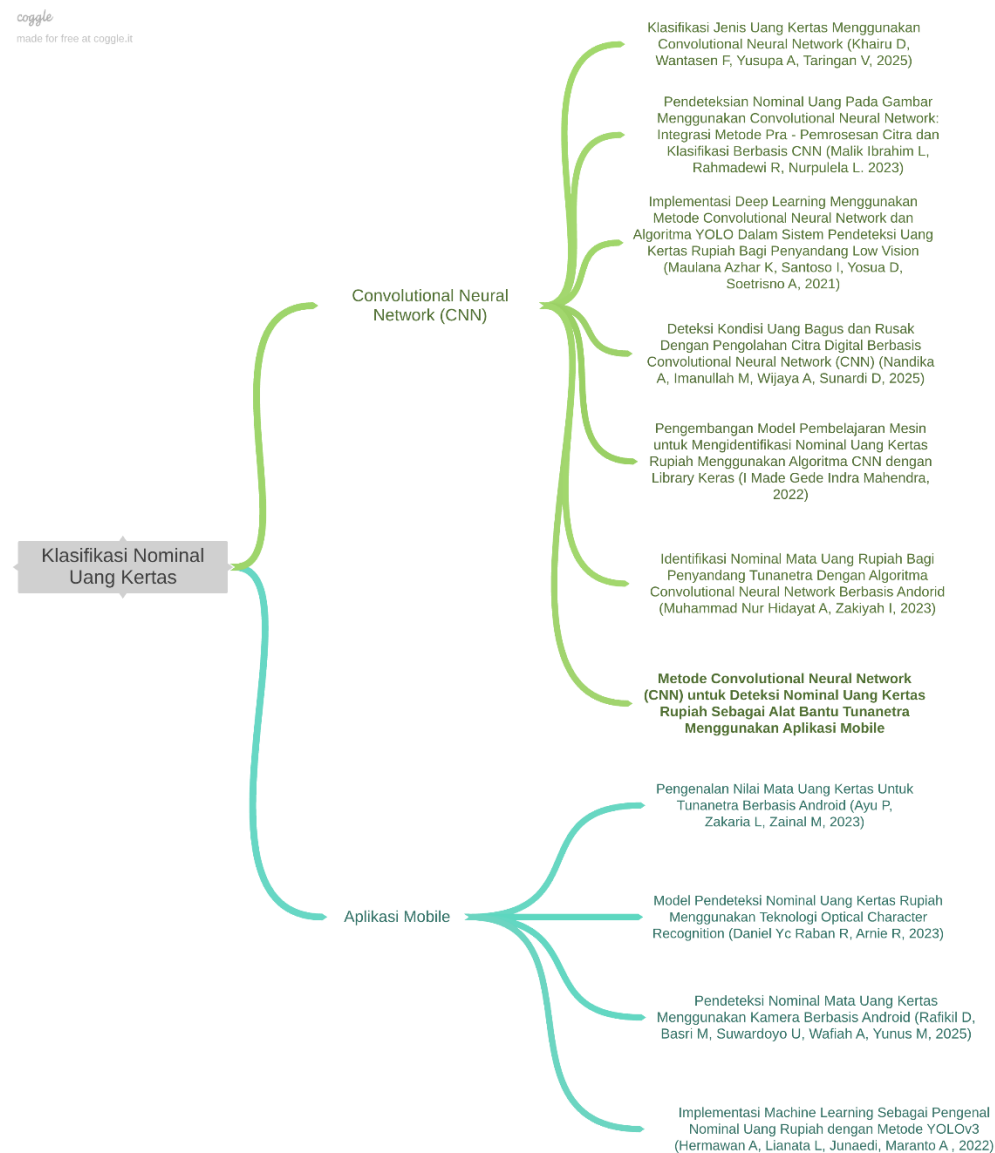
No.	Nama (Tahun)	Keterangan
(1)	(2)	(3)
9.	Nandika A, Imanullah M, Wijaya A, Sunardi D (2025)	Judul : Deteksi Kondisi Uang Bagus dan Rusak Dengan Pengolahan Citra Digital Berbasis <i>Convolutional Neural Network</i> (CNN) Metode : <i>Convolutional Neural Network</i> (CNN) Hasil Penelitian : Metode CNN mampu mengklasifikasikan kondisi uang kertas dengan akurasi validasi 93%, precision 97%, dan recall 96%. Sistem GUI berbasis MATLAB memudahkan penggunaan dan mendapat nilai 5/5 dari Bank Indonesia, sehingga berpotensi diterapkan dalam layanan perbankan.
10.	Rafikil D, Basri M, Suwardoyo U, Warfiah A, Yunus M (2025)	Judul : Pendeteksi Nominal Mata Uang Kertas Menggunakan Kamera Berbasis Android Metode : Aplikasi Mobile, YOLOv8 Hasil Penelitian : Aplikasi YOLOv8 di Android mampu mendeteksi nominal uang secara realtime pada kondisi normal dengan jarak ideal 10 – 15 cm. Namun, akurasi menurun saat uang terlipat tidak beraturan atau rusak parah sehingga berpotensi gagal deteksi.

Berdasarkan penelitian sebelumnya, terdapat perbedaan pendekatan dalam mendeteksi nominal uang kertas rupiah. Penelitian oleh Ayu P dkk. (2023) menggunakan metode *template matching* yang diimplementasikan pada aplikasi *mobile*, namun memiliki keterbatasan karena tingkat akurasi sangat dipengaruhi oleh kondisi pencahayaan dan jarak kamera. Sementara itu, penelitian oleh Khairu D dkk. (2025) menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur VGG-16 yang mampu menghasilkan akurasi tinggi, tetapi masih terbatas pada tahap klasifikasi dan belum diimplementasikan secara optimal dalam aplikasi *mobile* berbasis real-time. Hal ini menunjukkan adanya kesenjangan antara tingkat akurasi metode dan penerapannya dalam kondisi nyata.

Berdasarkan kesenjangan tersebut, penelitian ini menawarkan kebaruan berupa pengembangan sistem deteksi nominal uang kertas rupiah berbasis *Convolutional Neural Network* (CNN) yang diimplementasikan dalam aplikasi *mobile* berbasis android. Sistem yang dikembangkan tidak hanya menekankan pada tingkat akurasi,

tetapi juga pada kemampuan penggunaan secara real-time, serta dilengkapi dengan output suara untuk meningkatkan aksesibilitas bagi penyandang tunanetra. Selain itu, sistem diupayakan lebih adaptif terhadap variasi kondisi lingkungan, seperti pencahayaan dan jarak, sehingga lebih aplikatif dan relevan untuk digunakan dalam kehidupan sehari – hari.

Gambar 1 merupakan *mindmap* penelitian terkait sebagai referensi penulisan dalam bentuk gambar, yaitu sebagai berikut :



Gambar 1 *Mindmap* Penelitian

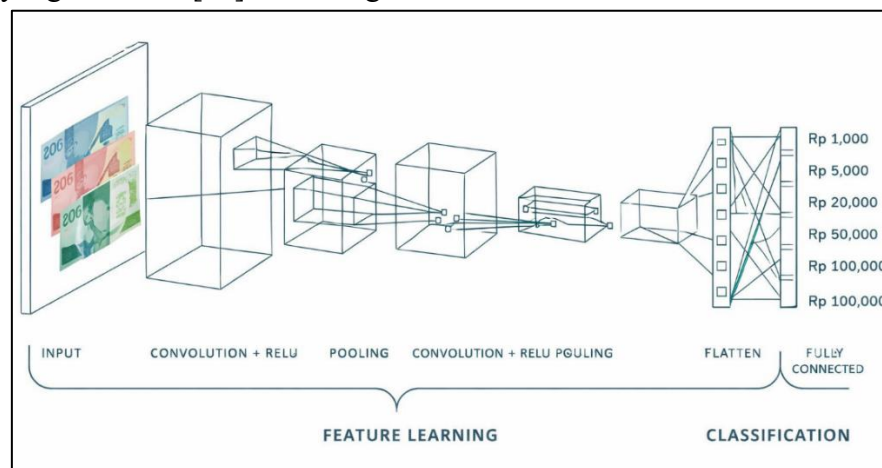
B. Landasan Teori

Landasan teori pada penelitian “Metode Convolutional Neural Network (CNN) untuk Deteksi Nominal Uang Kertas Rupiah Sebagai Alat Bantu Tunanetra Menggunakan Aplikasi Mobile” yaitu :

a. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu bagian dari ilmu Deep Neural. Metode CNN terinspirasi oleh cara saraf neuron otak manusia bekerja. Setiap objek visual yang diterima disimpan menjadi peta memori topografik yang disimpan di otak manusia, yang setiap levelnya difokuskan untuk mengenali karakteristik tertentu[9].

Algoritma CNN menggunakan konvolusi untuk mengidentifikasi fitur tertentu dalam data pelatihan. Fitur – fitur ini kemudian digunakan sebagai input untuk melatih jaringan saraf atau neural network untuk meningkatkan kemampuan pengenalan objek. Arsitektur CNN terdiri dari beberapa lapisan penting. Termasuk lapisan konvolusi, lapisan masukan, dan lapisan keluaran, serta lapisan tersembunyi seperti lapisan pooling, normalisasi, ReLU, fully connected layer, dan loss layer. Setiap lapisan memiliki tugas khusus yang membantu menjalankan kinerja jaringan secara keseluruhan dengan mengidentifikasi dan mengklasifikasikan karakteristik gambar yang diberikan[17]. Berikut gambar dari arsitektur CNN :



Gambar 2 Arsitektur CNN

b. Pengolahan Citra Digital

Citra adalah gambar yang dibuat dengan menggunakan titik, garis, bidang, dan warna untuk menciptakan gambaran dari benda hidup atau mati. Citra yang memiliki dua dimensi, seperti lukisan atau foto, atau tiga dimensi seperti patung. Citra sebagai keluaran suatu sistem perekaman data yang dapat bersifat optik yaitu berupa foto, bersifat analog berupa sinyal video seperti gambar pada monitor televisi atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan[18].

Digitalisasi citra analog terdiri dari kuantitas dan sampling. Sementara pengambilan sampel membagi gambar menjadi pixel atau elemen diskrit, kuantisasi menentukan intensitas warna untuk setiap pixel dengan nilai bulat. Setiap pixel memiliki nilai, yang dapat berupa angka atau nilai, yang menunjukkan intensitas keabuan pixel. Kode warna atau tingkat keabuan disebut skala gelap. Bit yang digunakan menentukan kisaran nilai dan menunjukkan resolusi skala abu – abu[19].

c. *Machine Learning* dan *Deep Learning*

Machine learning adalah bagian dari kecerdasan buatan yang berfokus pada aplikasi yang belajar sendiri dari data yang dilatih dengan algoritma yang efektif untuk memecahkan masalah. Data yang dimaksud dapat berupa angka, kata, gambar, dan lain – lain. Proses pembelajaran mesin terdiri dari proses pembelajaran terhadap mesin dari data yang kita berikan sehingga mesin tersebut dapat berpkiran dan berperilaku seperti manusia[20].

Deep learning menggunakan jaringan saraf dengan beberapa lapisan abstraksi untuk memproses data dengan cara yang mirip dengan proses kognitif manusia, sementara *machine learning* memungkinkan analisis data yang cepat dan akurat. Model – model seperti ini dapat menangani data tak terstruktur yang sering terjadi dalam transaksi keuangan, seperti teks dari korespondensi pelanggan atau kisah yang terjadi dalam transaksi[21].

d. Uang Kertas Rupiah

Uang kertas rupiah adalah uang kertas dalam bentuk lembaran yang dibuat dari kertas atau bahan lainnya yang mirip dengan kertas yang dikeluarkan oleh pemerintah Indonesia dan diizinkan untuk digunakan sebagai alat pembayaran di seluruh wilayah Republik Indonesia. Setiap orang di seluruh dunia menggunakan uang sebagai alat tukar utama untuk melakukan transaksi jual beli barang dan jasa[22].

e. Aplikasi Mobile

Aplikasi mobile adalah perangkat lunak yang dibuat dengan program aplikasi untuk tujuan tertentu. Aplikasi mobile yang baik mampu menyajikan informasi dan layanan secara efektif dan praktis, yang membedakannya dari aplikasi lainnya[23]. Aplikasi mobile memiliki keunggulan utama dibandingkan dengan aplikasi web dan desktop karena mereka memungkinkan akses kapan saja dan di mana saja melalui perangkat pintar yang sering digunakan setiap hari. Aplikasi mobile biasanya memiliki antarmuka pengguna (UI) yang lebih sederhana tetapi menarik. Selain itu, sebagian besar aplikasi mobile sangat sederhana untuk digunakan[24].

f. Android

Android yang dibuat oleh Google dan Open Handset Alliance adalah paket software perangkat bergerak yang baru. Android diharapkan akan menjadi platform utama untuk pengembang aplikasi dalam beberapa tahun mendatang, dengan ketersediaannya di jutaan ponsel dan perangkat bergerak lainnya. Saat ini, banyak platform mobile tersedia di pasar, termasuk Symbian, iPhone, Windows Mobile, Blackberry, Java Mobile Edition, dan Linux Mobile (LiMo)[25].

g. Text-to-Speech (TTS)

Teknologi Text-to-Speech (TTS) memungkinkan komputer mengubah teks tertulis menjadi ucapan. Merubah ucapan ke fonem dan teks ke fonem adalah dua prinsip utamanya. Pertama, rangkaian kode suara yang biasanya terdiri dari fonem,

durasi, dan nada, akan diubah menjadi sinyal suara yang sesuai dengan input. TTS telah banyak digunakan untuk membantu orang yang memiliki keterbatasan penglihatan dalam berbagai bidang utamanya[26].

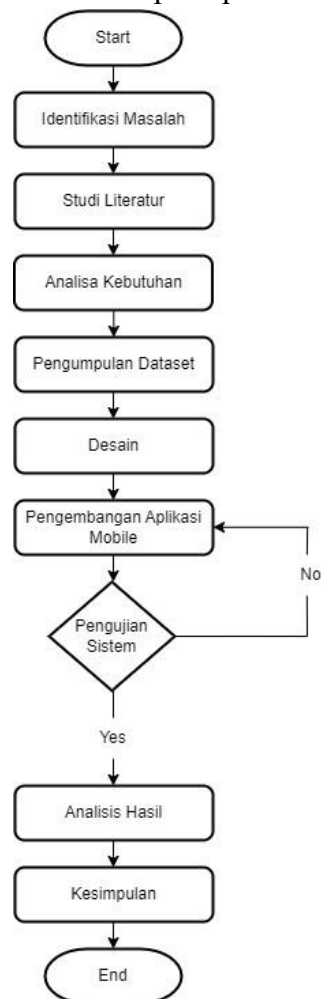
BAB III METODOLOGI

A. Waktu dan Tempat Penelitian

Waktu penelitian dilaksanakan selama satu semester di tahun 2026. Tempat penelitian dilaksanakan di Kabupaten Cilacap.

B. Prosedur Penelitian

Penelitian ini menerapkan beberapa tahapan yang harus dilaksanakan untuk menyelesaikan penelitian ini. Prosedur penelitian sebagai acuan dari latar belakang permasalahan dan penyelesaian masalah pada penelitian ini yaitu sebagai berikut :



Gambar 3 Flowchart Penelitian

Penjelasan dari setiap tahapan penelitian ini yaitu :

a. Identifikasi Masalah

Identifikasi masalah dilakukan untuk mengidentifikasi masalah yang dihadapi oleh penyandang tunanetra dalam mengenali nominal uang kertas secara mandiri. Hasilnya menunjukkan bahwa penyandang tunanetra mengalami kesulitan membedakan nominal uang karena keterbatasan penglihatan mereka, yang berpotensi menyebabkan kesalahan transaksi. Oleh karena itu, diperlukan solusi

berbasis teknologi yang dapat mendeteksi secara otomatis nominal uang kertas dan memberikan output suara untuk membantu.

b. Studi Literatur

Studi literatur dilakukan dengan cara pencarian buku, jurnal, atau lainnya yang berhubungan dengan penelitian. Tujuan dari studi literatur yaitu untuk mengkaji teori dan penelitian terdahulu yang relevan dengan penelitian ini. Pada tahap ini, peneliti mempelajari konsep dasar pengolahan citra digital, metode Convolutional Neural Network (CNN), serta implementasi *machine learning* pada aplikasi mobile. Selain itu, dilakukan penelaahan terhadap peneliti sebelumnya yang berkaitan dengan klasifikasi citra uang kertas dan teknologi bantu bagi penyandang tunanetra, Hasil studi literatur ini digunakan sebagai dasar dalam merancang model CNN dan mengembangkan sistem deteksi nominal uang kertas rupiah berbasis mobile dengan output suara.

c. Analisa Kebutuhan

Tahap analisa kebutuhan dilakukan untuk menentukan spesifikasi sistem yang diperlukan dalam pengembangan aplikasi deteksi nominal uang kertas rupiah berbasis Convolutional Neural Network (CNN). Analisa ini meliputi kebutuhan perangkat keras dan perangkat lunak agar sistem dapat berjalan sesuai tujuan penelitian, yaitu membantu penyandang tunanetra dalam mengenali nominal uang secara mandiri melalui output suara. Perangkat keras dan perangkat lunak yang dibutuhkan yaitu sebagai berikut :

i. Kebutuhan Perangkat Keras

Tabel 2 Kebutuhan Perangkat Keras

No	Perangkat Keras	Keterangan
1.	Laptop (Prosesor 12th Gen Intel(R) Core(TM) i5-1235U 1.30 GHz, RAM 16 GB)	Sebagai media pengembangan dan pelatihan model CNN.
2.	Smartphone Android (Versi 11, RAM 4 GB, Penyimpanan 64 GB)	Sebagai perangkat implementasi dan pengujian aplikasi deteksi nominal uang kertas rupiah berbasis mobile

ii. Kebutuhan Perangkat Lunak

Tabel 3 Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Keterangan
1.	Visual Studio Code (VS Code)	Untuk mengembangkan aplikasi mobile menggunakan framework Flutter serta untuk

No	Perangkat Lunak	Keterangan
		menulis kode program model CNN.
2.	Flutter SDK	Untuk membangun aplikasi mobile berbasis Android
3.	Python	Untuk proses pelatihan model Convolutional Neural Network (CNN).
4.	TensorFlow / Keras	Sebagai library dalam membangun dan melatih model CNN untuk klasifikasi nominal uang kertas rupiah.
5.	TensorFlow Lite Converter	Untuk mengonversi model hasil training ke format .tflite agar dapat diimplementasikan pada aplikasi mobile.
6.	Android SDK	Sebagai pendukung dalam proses build dan menjalankan aplikasi pada perangkat android.

d. Pengumpulan Dataset

Tahap pengumpulan dataset dilakukan untuk memperoleh data citra uang kertas rupiah yang akan digunakan dalam proses pelatihan dan pengujian model Convolutional Neural Network (CNN). Dataset yang digunakan berupa gambar uang kertas dari beberapa nominal yang beredar di Indonesia.

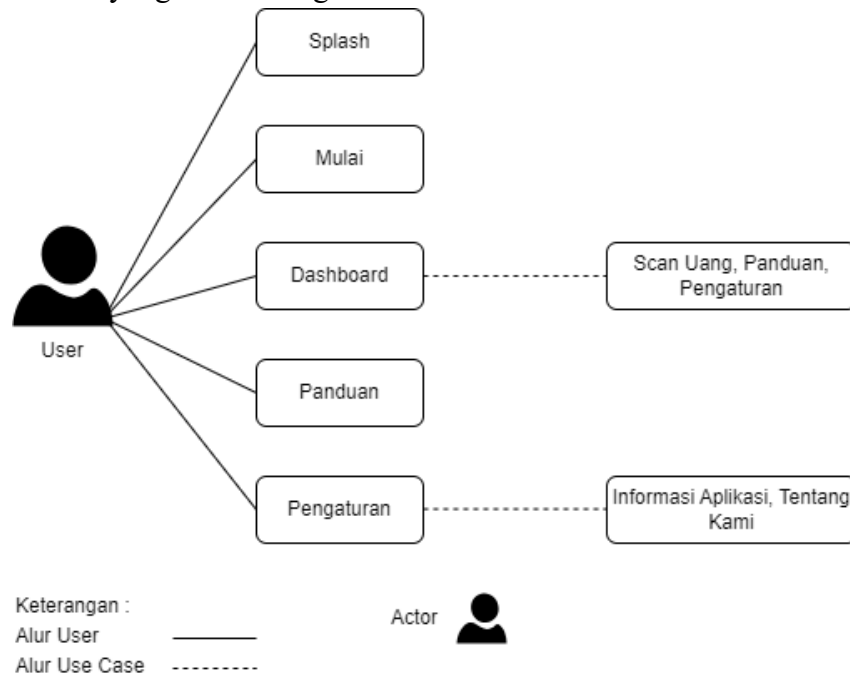
Pengambilan gambar dilakukan menggunakan kamera smartphone dengan berbagai variasi kondisi, seperti perbedaan pencahayaan, sudut pengambilan gambar, jarak kamera terhadap objek, dan latar belakang yang berbeda. Variasi tersebut bertujuan agar model CNN mampu mengenali nominal uang dengan baik dalam berbagai kondisi nyata. Dataset kemudian dikelompokkan berdasarkan masing – masing nominal, misalnya Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, dan Rp 100.000.

e. Desain

Desain aplikasi merupakan tahap perancangan sistem sebelum proses implementasi dilakukan. Pada tahap ini ditentukan bagaimana struktur aplikasi, tampilan antarmuka, alur kerja sistem, serta interaksi antara pengguna dan sistem agar aplikasi dapat berjalan sesuai dengan tujuan penelitian.

Dalam penelitian ini, desain aplikasi bertujuan untuk merancang sistem deteksi nominal uang kertas rupiah berbasis Convolutional Neural Network (CNN) yang mudah digunakan oleh penyandang tunanetra. Desain dilakukan dengan memperhatikan aspek fungsionalitas, kemudahan pengguna, dan aksesibilitas, sehingga aplikasi dapat memberikan output dalam bentuk teks dan suara secara

efektif. Perancangan yang dibuat untuk penggambaran aplikasi mobile melalui diagram *use case* yang mana sebagai berikut :



Gambar 4 Diagram *Use Case*

Berikut penjelasan dari diagram *use case* diatas :

i. *Splash*

Splash screen merupakan halaman awal yang muncul saat aplikasi dijalankan. Halaman ini berfungsi sebagai tampilan pembuka sekaligus untuk melakukan proses inisialisasi sitem sebelum *user* masuk ke halaman utama. Pada penelitian ini, *splash screen* digunakan untuk menampilkan identitas aplikasi dan mempersiapkan komponen sistem seperti model CNN dan fitur *text-to-speech*.

ii. *Mulai*

Halaman masuk merupakan halaman yang ditampilkan setelah *splash screen* sebagai tahap awal sebelum *user* mengakses *dashboard* aplikasi. Pada halaman ini terdapat tombol “Start” yang berfungsi untuk melanjutkan ke halaman utama aplikasi. Halaman masuk pada aplikasi ini tidak memerlukan proses login atau autentikasi pengguna. Tombol “Start” hanya berfungsi sebagai navigasi untuk berpindah dari halaman pembuka menuju *dashboard*.

iii. *Dashboard*

Halaman *dashboard* merupakan halaman utama aplikasi yang ditampilkan setelah *user* menekan tombol “Start”. Halaman ini berfungsi sebagai pusat navigasi untuk mengakses seluruh fitur yang tersedia dalam aplikasi seperti, scan uang, panduan, dan pengaturan.

Dashboard dilengkapi dengan fitur output suara yang memberikan informasi ketika halaman dibuka maupun saat tombol ditekan.

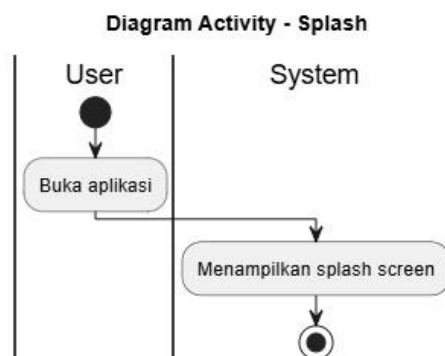
iv. Panduan

Halaman panduan merupakan halaman yang berisi informasi dan petunjuk penggunaan aplikasi. Halaman ini bertujuan untuk membantu user memahami cara mengoperasikan fitur yang tersedia, khususnya dalam melakukan proses scan uang dan mendengarkan hasil deteksi. Selain menampilkan teks panduan, halaman ini juga dilengkapi dengan fitur suara yang secara otomatis membacakan isi panduan.

v. Pengaturan

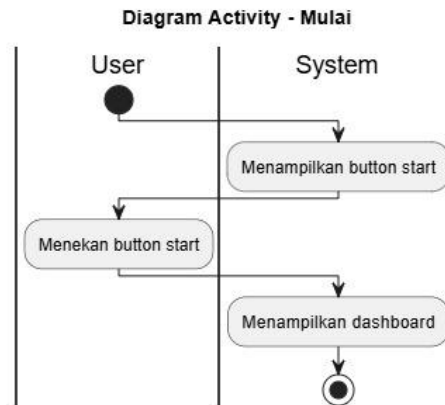
Halaman pengaturan merupakan halaman yang berisi informasi terkait aplikasi. Pada halaman ini terdapat menu Informasi Aplikasi dan Tentang Kami. Menu Informasi Aplikasi menampilkan penjelasan singkat mengenai fungsi dan tujuan aplikasi, sedangkan menu Tentang Kami berisi informasi mengenai pengembang aplikasi.

Dari keterangan pada diagram use case, terdapat diagram activity sebagai proses menggambarkan urutan aktivitas yang terjadi pada aplikasi deteksi uang kertas rupiah agar lebih mudah dipahami. Berikut adalah diagram activity pada aplikasi deteksi uang kertas rupiah :



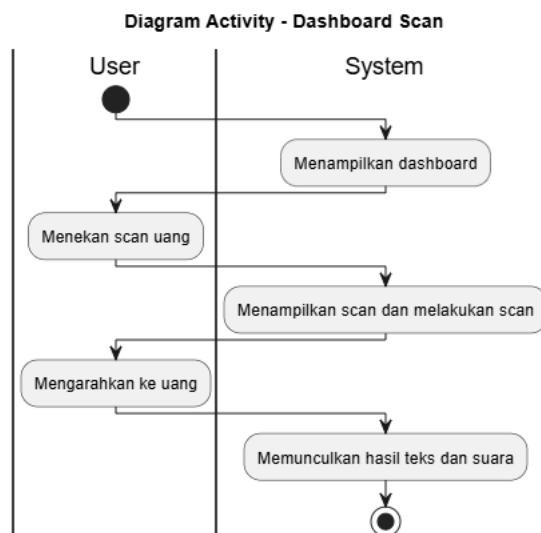
Gambar 5 Diagram Activity *Splash*

Gambar 5 yaitu diagram activity *splash* bagaimana alur *splash* berjalan. *User* memulai aplikasi pertama kali yang ditampilkan yaitu *splash* dengan delay tiga detik. Setelah delay waktu tiga detik selanjutnya yang ditampilkan yaitu halaman masuk.



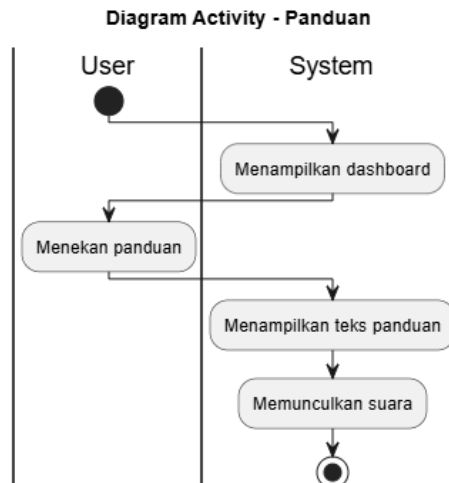
Gambar 6 Diagram Activity Masuk

Gambar 6 yaitu diagram activity mulai berjalan. Pada halaman mulai menampilkan *button start*. Kemudian user menekan *button start* yang langsung menampilkan *dashboard*.



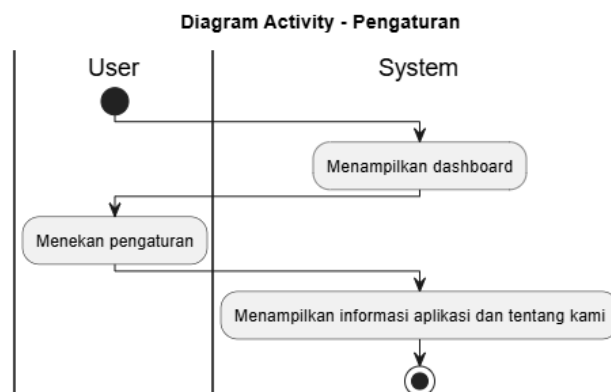
Gambar 7 Diagram Activity *Dashboard Scan*

Gambar 7 yaitu diagram activity dashboard scan bagaimana alur *scan* berjalan. Pada *dashboard* terdapat *button scan* uang, panduan, dan pengaturan. Ketika *user* menekan *button scan*, sistem akan mengaktifkan kamera dan menampilkan halaman *scan* untuk mendeteksi nominal uang kertas. Setelah *scan* dilakukan, maka akan menampilkan hasil berupa teks dan suara.



Gambar 8 Diagram Activity Panduan

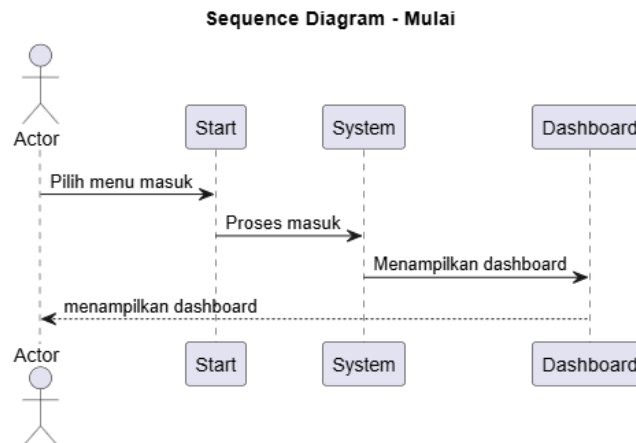
Gambar 8 yaitu diagram activity panduan berjalan. Ketika *user* membuka aplikasi dan berada di *dashboard*, kemudian memilih menu panduan. Sistem selanjutnya menampilkan halaman panduan yang berisi informasi cara penggunaan aplikasi, mulai dari proses *scan* hingga mendengarkan hasil deteksi suara.



Gambar 9 Diagram Activity Pengaturan

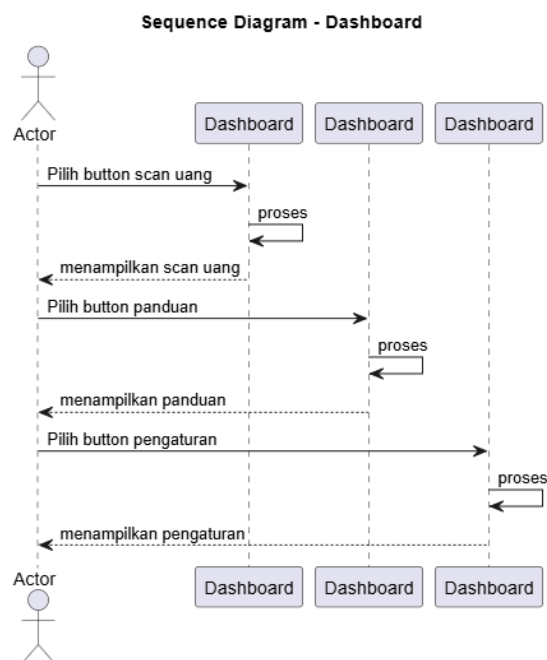
Gambar 9 yaitu diagram activity pengaturan berjalan. Pada *dashboard* terdapat menu *scan* uang, panduan, dan pengaturan. Sistem menampilkan menu informasi aplikasi dan tentang kami. Jika *user* memilih informasi aplikasi, sistem menampilkan detail aplikasi seperti versi dan deskripsi singkat. Jika *user* memilih tentang kami, sistem menampilkan informasi mengenai pengembang aplikasi.

Dari diagram activity, terdapat diagram sequence untuk mengetahui urutan kejadian yang dapat menghasilkan output yang diinginkan. Berikut diagram sequence sebagai berikut :



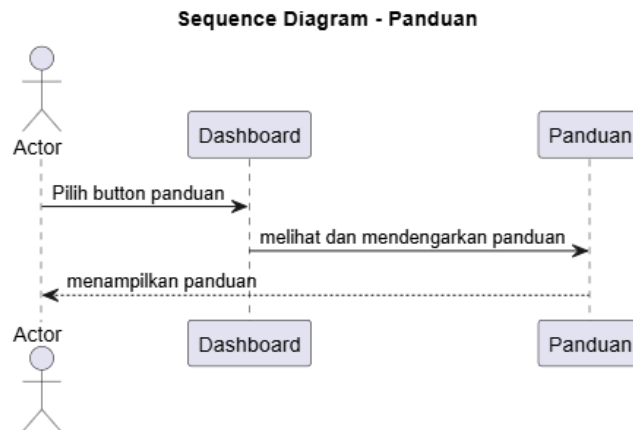
Gambar 10 Diagram Sequence Masuk

Diagram *sequence* pada Gambar 10 berupa diagram *sequence* masuk. Proses ini dimulai ketika *actor* memilih menu masuk. Permintaan tersebut dikirim ke sistem untuk di proses. Sistem kemudian melakukan proses masuk, dan setelah berhasil, sistem menampilkan halaman *dashboard* kepada *user*.



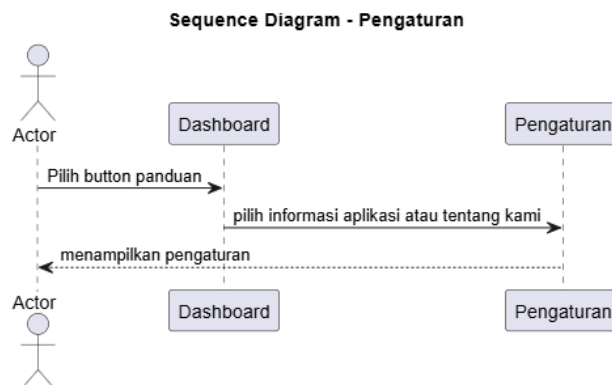
Gambar 11 Diagram Sequence Dashboard

Diagram *sequence* pada Gambar 11 yaitu diagram *sequence dashboard*. Proses ini menggambarkan interaksi antara *actor* dan sistem saat di halaman *dashboard*. Jika *user* memilih *button scan*, maka sistem akan memproses permintaan dan menampilkan halaman *scan* uang. Jika *user* memilih *button panduan*, maka sistem akan memproses permintaan dan menampilkan halaman panduan penggunaan aplikasi. Jika *user* memilih *button pengaturan*, maka sistem memproses dan menampilkan halaman pengaturan.



Gambar 12 Diagram Sequence Panduan

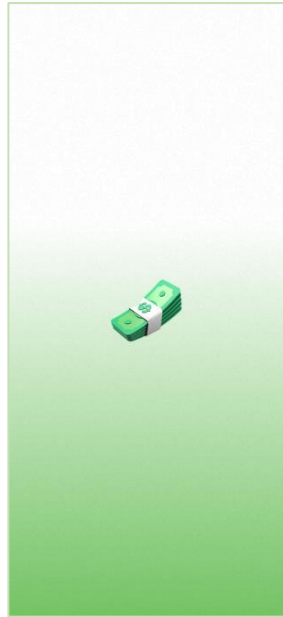
Diagram *sequence* pada Gambar 12 yaitu diagram *sequence* panduan. Proses dimulai saat *actor* memilih menu panduan pada *dashboard*. Sistem akan memproses dan menampilkan halaman panduan. Setelah halaman panduan ditampilkan, *user* dapat melihat dan mendengarkan informasi panduan yang tersedia dalam aplikasi.



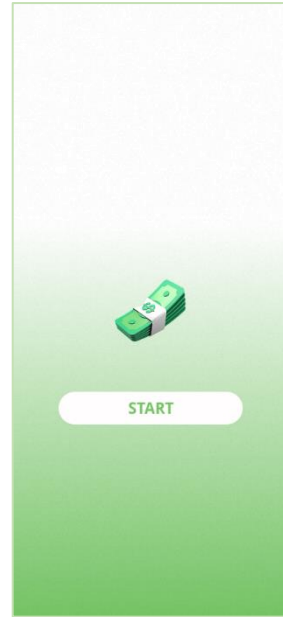
Gambar 13 Diagram Sequence Pengaturan

Diagram *sequence* pada Gambar 13 yaitu diagram *sequence* pengaturan. Proses dimulai saat *actor* memilih menu pengaturan pada *dashboard*. Di menu pengaturan menampilkan pilihan informasi aplikasi dan tentang kami. Jika *actor* memilih informasi aplikasi maka menampilkan detail aplikasi. Jika *actor* memilih tentang kami maka menampilkan informasi tentang pengembang aplikasi.

Dari diagram *use case*, diagram *activity*, dan diagram *sequence* selanjutnya yaitu rancangan *mockup* sebagai tampilan aplikasi deteksi uang kertas rupiah. Gambar 14 sampai Gambar 22 adalah *mockup* aplikasi deteksi uang kertas rupiah, yaitu sebagai berikut :



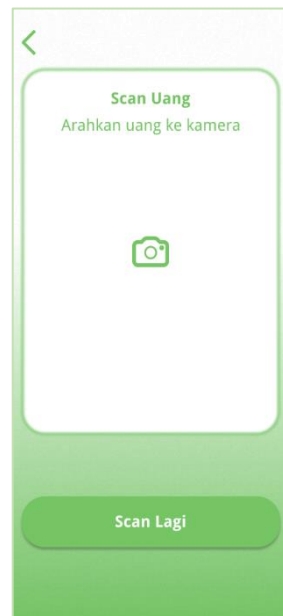
Gambar 14 Mockup Splash



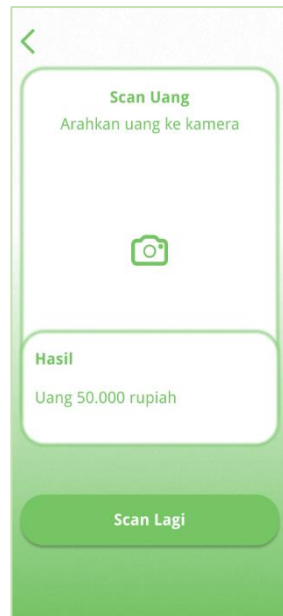
Gambar 15 Mockup Mulai



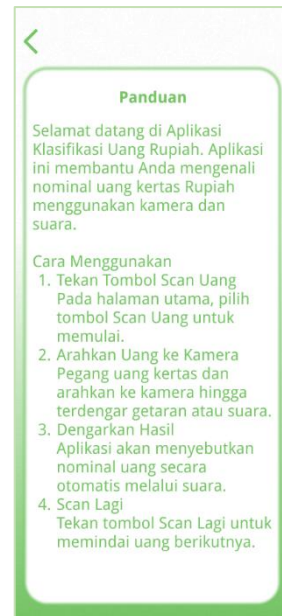
Gambar 16 Mockup Dashboard



Gambar 17 Mockup Scan



Gambar 18 Mockup Hasil Scan



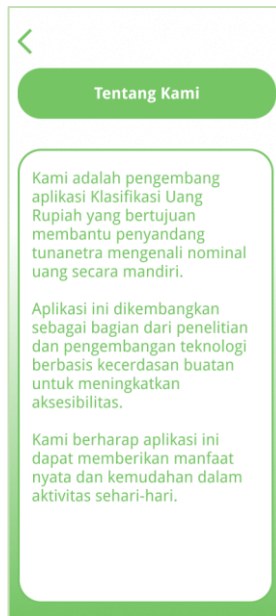
Gambar 19 Mockup Panduan



Gambar 20 Mockup Pengaturan



Gambar 21 Mockup Informasi Aplikasi



Gambar 22 Mockup Tentang Kami

f. Pengembangan Aplikasi Mobile

Pengembangan aplikasi mobile pada penelitian ini dilakukan untuk mengimplementasikan model Convolutional Neural Network (CNN) ke dalam sebuah aplikasi berbasis android yang dapat digunakan secara langsung oleh pengguna, khususnya penyandang tunanetra. Aplikasi dikembangkan menggunakan framework Flutter dengan bahasa pemrograman Dart, serta memanfaatkan Visual Studio Code sebagai code editor.

Tahap pengembangan dimulai dari perancangan struktur aplikasi yang terdiri dari splash screen, halaman masuk, dashboard, halaman scan, halaman panduan, dan halaman pengaturan. Setiap halaman dirancang dengan tampilan sederhana, serta kontras warna yang jelas guna meningkatkan aksesibilitas pengguna.

Selanjutnya dilakukan integrasi model CNN yang telah dilatih sebelumnya dan dikonversi ke dalam format TensorFlow Lite. Model ini ditanamkan ke dalam aplikasi agar proses klasifikasi citra dapat berjalan secara offline tanpa memerlukan koneksi internet. Fitur utama yang dikembangkan dalam aplikasi meliputi :

- i. Integrasi kamera untuk pengambilan gambar uang kertas.
- ii. Pemrosesan citra menggunakan model CNN untuk mengklasifikasikan nominal uang.
- iii. Penampilan hasil klasifikasi dalam bentuk teks.
- iv. Output suara menggunakan teknologi text-to-speech (TTS) untuk menyampaikan hasil deteksi kepada pengguna.
- v. Menu panduan dan pengaturan untuk mendukung kemudahan penggunaan.

Setelah tahap implementasi selesai, dilakukan pengujian aplikasi pada perangkat smartphone android untuk memastikan seluruh fitur berjalan dengan baik, responsif, dan sesuai dengan kebutuhan pengguna. Pengujian dilakukan untuk

mengevaluasi performa deteksi nominal uang, kejelasan suara, serta kemudahan navigasi aplikasi.

g. Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui tingkat kinerja dan keakuratan aplikasi dalam mendeteksi nominal uang kertas rupiah menggunakan metode Convolutional Neural Network (CNN). Tahap ini bertujuan untuk memastikan bahwa sistem berjalan sesuai dengan kebutuhan fungsional dan dapat digunakan secara optimal oleh pengguna, khususnya penyandang tunanetra.

i. Pengujian fungsional aplikasi

Pengujian pada menu splash

Tabel 4 Pengujian Menu Splash

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Tampilan splash screen	Run aplikasi	Menampilkan logo aplikasi sebagai pembuka aplikasi mulai berjalan	[] Berhasil [] Tidak Berhasil

Pengujian pada menu masuk

Tabel 5 Pengujian Menu Masuk

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses button masuk	Klik button 'Start'	Masuk berhasil dan mengakses dashboard	[] Berhasil [] Tidak Berhasil

Pengujian pada menu dashboard

Tabel 6 Pengujian Menu Dashboard

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses tampilan dashboard	Masuk berhasil dan menampilkan dashboard	Menampilkan menu utama	[] Berhasil [] Tidak Berhasil
Akses scan uang	Klik button 'scan uang'	Menampilkan kamera, mengeluarkan hasil teks dan suara	[] Berhasil [] Tidak Berhasil

Akses panduan	Klik button 'panduan'	Menampilkan teks dan suara panduan	[] Berhasil [] Tidak Berhasil
Akses pengaturan	Klik button 'pengaturan'	Menampilkan button 'Informasi Aplikasi' dan 'Tentang Kami'	[] Berhasil [] Tidak Berhasil

Pengujian pada menu panduan

Tabel 7 Pengujian Menu Panduan

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses panduan	Klik button 'Panduan'	Menampilkan teks dan suara panduan	[] Berhasil [] Tidak Berhasil

Pengujian pada menu pengaturan

Tabel 8 Pengujian Menu Pengaturan

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses pengaturan	Klik button 'Pengaturan'	Menampilkan button 'Informasi Aplikasi' dan 'Tentang Kami'	[] Berhasil [] Tidak Berhasil

Pengujian pada menu informasi aplikasi

Tabel 9 Pengujian Menu Informasi Aplikasi

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses informasi aplikasi	Klik button 'Informasi Aplikasi'	Menampilkan teks dan suara informasi aplikasi	[] Berhasil [] Tidak Berhasil

Pengujian pada menu tentang kami

Tabel 10 Pengujian Menu Tentang Kami

Skenario Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses tentang kami	Klik button 'Tentang Kami'	Menampilkan teks dan suara tentang kami	[] Berhasil [] Tidak Berhasil

ii. Pengujian fungsional sistem

Pengujian sistem dilakukan untuk mengetahui kinerja aplikasi dalam mendeteksi nominal uang kertas rupiah pada berbagai kondisi penggunaan. Pengujian dilakukan berdasarkan dua skenario, yaitu kondisi pencahayaan dan jarak kamera terhadap objek. Pada pengujian pencahayaan, sistem diuji pada kondisi cahaya terang, normal, redup, dan sangat rendah untuk mengetahui pengaruh intensitas cahaya terhadap akurasi deteksi. Sedangkan pada pengujian jarak, sistem diuji pada beberapa variasi jarak antara kamera dan uang kertas untuk mengetahui jarak optimal agar sistem dapat mendeteksi nominal uang dengan baik.

Tabel 11 Pengujian Sistem Berdasarkan Jarak

No	Jenis Pengujian	Tujuan Pengujian	Skenario Pengujian	Metrik Evaluasi
1	Pengujian Jarak Dekat	Mengetahui kemampuan sistem dalam mendeteksi nominal uang pada jarak dekat	Kamera diarahkan ke uang kertas pada jarak ± 10 cm dari kamera	Sistem mampu mendeteksi nominal uang dengan benar dan menampilkan hasil dalam bentuk teks dan suara.
2	Pengujian Jarak Sedang	Mengetahui kemampuan sistem dalam mendeteksi uang pada jarak sedang	Kamera diarahkan ke uang kertas pada jarak ± 20 cm dari kamera	Sistem mampu mengenali uang dengan tingkat akurasi yang baik
3	Pengujian Jarak Jauh	Mengetahui kemampuan sistem dalam mendeteksi nominal uang pada jarak yang lebih jauh	Kamera diarahkan ke uang kertas pada jarak ± 30 cm dari kamera	Sistem masih dapat mendeteksi nominal uang meskipun akurasi dapat menurun
4	Pengujian Jarak Maksimal	Mengetahui batas maksimal jarak kamera dalam	Kamera diarahkan ke uang kertas pada jarak lebih dari 40 cm	Sistem dapat atau tidak dapat mendeteksi nominal uang secara akurat

No	Jenis Pengujian	Tujuan Pengujian	Skenario Pengujian	Metrik Evaluasi
		mendeteksi uang		

Tabel 12 Pengujian Sistem Berdasarkan Cahaya

No	Jenis Pengujian	Rentang Cahaya	Tujuan Pengujian	Skenario Pengujian	Metrik Evaluasi
1	Pengujian Cahaya Terang	> 500 lux	Mengetahui performa sistem dalam kondisi pencahayaan yang terang	Proses <i>scanning</i> uang dilakukan di ruangan dengan pencahayaan terang	Sistem mampu mendeteksi nominal uang dengan akurasi tinggi
2	Pengujian Cahaya Normal	100 – 500 lux	Menguji performa sistem pada kondisi pencahayaan normal	Proses <i>scanning</i> dilakukan di ruangan dengan pencahayaan standar	Sistem dapat mendeteksi nominal uang dengan baik
3	Pengujian Cahaya Redup	10 – 100 lux	Mengetahui kemampuan sistem dalam kondisi pencahayaan rendah	Proses <i>scanning</i> dilakukan di ruangan dengan cahaya redup	Sistem masih dapat mendeteksi nominal uang namun akurasi dapat menurun
4	Pengujian Cahaya Sangat Rendah	< 10 lux	Mengetahui batas kemampuan sistem dalam kondisi minim cahaya	Proses <i>scanning</i> dilakukan pada kondisi cahaya sangat rendah	Sistem kesulitan mendeteksi nominal uang atau hasil tidak akurat

iii. Pengujian akurasi model

Pengujian akurasi pada penelitian ini dilakukan menggunakan metode Confusion Matrix untuk mengevaluasi performa model Convolutional Neural Network (CNN) dalam mengklasifikasikan nominal uang kertas rupiah. Confusion Matrix merupakan metode evaluasi yang membandingkan antara *actual class* dengan *predicted class*, sehingga dapat diketahui tingkat ketepatan dan kesalahan klasifikasi secara rinci pada setiap kelas.

Pada klasifikasi multi-kelas, Confusion Matrix berbentuk tabel berukuran $N \times N$, di mana N adalah jumlah kelas. Setiap elemen dalam matriks menunjukkan jumlah data yang diprediksi sebagai kelas tertentu terhadap kelas sebenarnya. Berdasarkan Confusion Matrix, diperoleh empat komponen utama, yaitu :

- True Positive (TP) : data yang diprediksi benar sesuai dengan kelas sebenarnya.
- True Negative (TN) : data yang diprediksi bukan kelas tersebut dan memang bukan kelas tersebut.
- False Positive (FP) : data yang diprediksi sebagai suatu kelas, tetapi sebenarnya bukan kelas tersebut.
- False Negative (FN) : data yang sebenarnya termasuk suatu kelas, tetapi diprediksi sebagai kelas lain.

Dari komponen tersebut dihitung beberapa metrik evaluasi sebagai berikut :

a. Accuracy

Accuracy digunakan untuk mengukur tingkat ketepatan keseluruhan model dalam melakukan klasifikasi.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Pada klasifikasi multi-kelas, accuracy dapat dihitung dengan :

$$Accuracy = \frac{\sum TP}{Total\ seluruh\ data}$$

b. Precision

Precision mengukur tingkat ketepatan model dalam memprediksi suatu kelas tertentu.

$$Precision = \frac{TP}{TP + FP}$$

Nilai precision yang tinggi menunjukkan bahwa prediksi model terhadap suatu kelas memiliki kesalahan yang kecil.

c. Recall

Recall mengukur kemampuan model dalam mendeteksi seluruh data yang benar – benar termasuk dalam suatu kelas.

$$Recall = \frac{TP}{TP + FN}$$

Nilai recall yang tinggi menunjukkan bahwa model mampu mengenali sebagian besar data yang seharusnya terklasifikasi pada kelas tersebut.

d. F1-Score

F1-Score merupakan rata – rata harmonis antara precision dan recall, yang digunakan untuk menilai keseimbangan performa model.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Penggunaan Confusion Matrix dalam penelitian ini bertujuan untuk memberikan evaluasi performa model secara menyeluruh, tidak hanya berdasarkan accuracy tetapi juga berdasarkan kemampuan model dalam mengurangi kesalahan klasifikasi antar nominal uang yang memiliki kemiripan visual.

Tabel 13 Pengujian Akurasi Model

Skenario Pengujian	Deskripsi	Parameter yang Diukur	Output yang Diharapkan
Pengujian data uji	Menguji model menggunakan dataset testing yang tidak digunakan saat training	Accuracy	Nilai akurasi tingi > 85%
Evaluasi per kelas	Menghitung precision dan recall tiap nominal	Precision, Recall	Kesalahan antar nominal rendah
Evaluasi keseluruhan	Menghitung F1-Score	F1-Score	Performa model seimbang

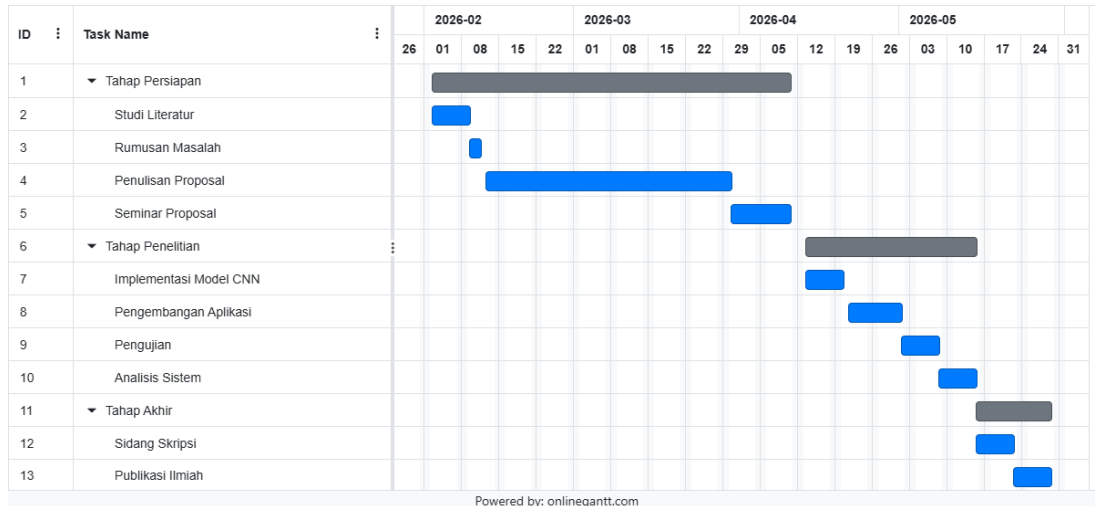
h. Analisis Hasil

Analisis hasil penelitian ini adalah untuk mengetahui seberapa baik sistem deteksi nominal uang kertas rupiah berbasis Convolutional Neural Network (CNN) dapat bekerja. Analisis melihat bagaimana sistem bekerja secara keseluruhan, baik dari fungsi aplikasi maupun kemampuan model untuk mengklasifikasi gambar. Secara umum, diharapkan bahwa sistem dapat mengenali nominal uang dengan tingkat ketepatan yang tinggi, menampilkan hasil klasifikasi dengan jelas, dan memberikan output suara yang sesuai dengan hasil deteksi. Selain itu, aplikasi diharapkan berjalan lancar pada perangkat android tanpa mengalami masalah atau kesalahan sistem.

Selain itu, analisis memasukkan pengamatan terhadap variabel yang dapat memengaruhi hasil deteksi, seperti kondisi pencahayaan, sudut pengambilan gambar, dan kualitas kamera perangkat. Faktor – faktor ini dipertimbangkan saat mengevaluasi kinerja sistem secara keseluruhan. Setelah analisis hasil, akan diketahui apakah sistem yang dikembangkan telah memenuhi tujuan penelitian, yaitu membantu penyandang tunanetra dengan menggunakan aplikasi mobile berbasis CNN untuk mengenali nominal uang kertas rupiah secara mandiri.

C. Jadwal Penelitian

Jadwal penelitian dibuat untuk mengatur waktu agar penelitian terarah dan terstruktur, jadwal penelitian dibuat dalam bentuk Gantt Chart berikut Gantt Chart jadwal penelitian dapat dilihat pada Gambar 23.



Gambar 23 Gantt Chart Jadwal Penelitian

BAB IV HASIL DAN PEMBAHASAN

A. Hasil

Pada penelitian ini telah menghasilkan sebuah aplikasi mobile berbasis Android yang menggunakan metode *Convolutional Neural Network* (CNN) untuk mendeteksi nominal uang kertas rupiah. Aplikasi ini dirancang untuk membantu penyandang tunanetra dalam mengenali nominal uang secara mandiri melalui output suara.

a. Implementasi Sistem

Implementasi sistem dilakukan menggunakan *Google Colaboratory* sebagai lingkungan pengembangan model *Convolutional Neural Network* (CNN). Tahapan implementasi dimulai dengan mempersiapkan lingkungan kerja melalui koneksi *Google Drive*, pemanggilan library yang dibutuhkan, penentuan lokasi dataset, serta proses preprocessing data. Koneksi *Google Drive* dilakukan untuk memudahkan akses dataset dan penyimpanan hasil pelatihan model. Selanjutnya dilakukan pemanggilan berbagai library seperti TensorFlow, NumPy, Matplotlib, Seaborn, dan Scikit-Learn yang digunakan dalam proses pembangunan, pelatihan, dan evaluasi model.

```
from google.colab import drive
drive.mount('/content/drive')

import tensorflow as tf
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import os

from sklearn.metrics import confusion_matrix,
classification_report

from tensorflow.keras.preprocessing.image import
ImageDataGenerator

data_dir = "/content/drive/MyDrive/dataset"

preprocess =
tf.keras.applications.mobilenet_v2.preprocess_inpu
t
```

```

datagen = ImageDataGenerator(
    preprocessing_function=preprocess,
    validation_split=0.2,
    rotation_range=25,
    zoom_range=0.2,
    shear_range=0.2,
    horizontal_flip=True,
    brightness_range=[0.5, 1.5]
)

train_data = datagen.flow_from_directory(
    data_dir,
    target_size=(224,224),
    batch_size=32,
    class_mode='categorical',
    subset='training'
)

val_data = datagen.flow_from_directory(
    data_dir,
    target_size=(224,224),
    batch_size=32,
    class_mode='categorical',
    subset='validation',
    shuffle=False
)

```

Dataset yang digunakan merupakan kumpulan citra uang kertas rupiah yang telah dikelompokkan berdasarkan nominalnya. Sebelum digunakan dalam proses pelatihan, data terlebih dahulu melalui tahap preprocessing menggunakan fungsi bawaan MobileNetV2. Selain itu dilakukan augmentasi data berupa rotasi, zoom, shear, horizontal flip, dan perubahan tingkat kecerahan. Penerapan augmentasi bertujuan untuk meningkatkan variasi data sehingga model dapat mengenali uang kertas dalam berbagai kondisi pencahayaan dan posisi pengambilan gambar. Setelah proses augmentasi selesai dilakukan, dataset dibagi menjadi data training sebesar 80% dan data validation sebesar 20%.

```

labels = list(train_data.class_indices.keys())

base_model = tf.keras.applications.MobileNetV2(
    input_shape=(224,224,3),
    include_top=False,
    weights='imagenet'
)

for layer in base_model.layers[-30:]:
    layer.trainable = True

x = base_model.output
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = tf.keras.layers.Dense(128,
    activation='relu')(x)
x = tf.keras.layers.Dropout(0.5)(x)
outputs = tf.keras.layers.Dense(
    len(labels),
    activation='softmax'
)(x)

model = tf.keras.Model(
    inputs=base_model.input,
    outputs=outputs
)

```

Setelah proses persiapan data selesai dilakukan, sistem menghasilkan label kelas berdasarkan nama folder dataset. Label tersebut disimpan ke dalam file `labels.txt` yang nantinya digunakan untuk menerjemahkan hasil prediksi model menjadi nominal uang pada aplikasi android. Tahap berikutnya adalah pembangunan model CNN menggunakan arsitektur MobileNetV2 yang telah dilatih sebelumnya menggunakan dataset ImageNet. Penggunaan MobileNetV2 dipilih karena memiliki ukuran model yang ringan sehingga cocok diterapkan pada perangkat mobile. Untuk meningkatkan kemampuan model dalam mengenali karakteristik uang kertas rupiah, dilakukan proses fine tuning pada 30 layer terakhir. Selanjutnya ditambahkan beberapa layer klasifikasi berupa Global Average Pooling Layer, Dense Layer, Dropout Layer, dan Output Layer dengan fungsi aktivasi Softmax.

```

model.compile(
    optimizer=tf.keras.optimizers.Adam(
        learning_rate=0.0001
    ),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)

callbacks = [
    tf.keras.callbacks.EarlyStopping(
        patience=5,
        restore_best_weights=True
    )
]

history = model.fit(
    train_data,
    validation_data=val_data,
    epochs=25,
    callbacks=callbacks
)

```

Kemudian model dikompilasi menggunakan optimizer Adam dengan learning rate sebesar 0,0001 dan fungsi loss categorical crossentropy. Untuk mengurangi risiko overfitting digunakan metode Early Stopping yang akan menghentikan proses pelatihan apabila tidak terjadi peningkatan performa model dalam beberapa epoch berturut – turut. Setelah seluruh konfigurasi selesai dilakukan, proses training model dijalankan menggunakan data training dan data validation yang telah dipersiapkan sebelumnya.

```

model.save("/content/drive/MyDrive/model_uang.h5")

Y_pred = model.predict(val_data)
y_pred = np.argmax(Y_pred, axis=1)

cm = confusion_matrix(
    val_data.classes,
    y_pred
)

print(
    classification_report(
        val_data.classes,
        y_pred,
        target_names=labels
    )
)

```

Setelah proses pelatihan selesai dilakukan, model disimpan dalam format H5. Selain itu sistem juga menghasilkan grafik akurasi dan grafik loss yang digunakan untuk memantau perkembangan performa model selama proses training. Evaluasi model dilakukan menggunakan confusion matrix dan classification report untuk mengetahui tingkat ketepatan klasifikasi pada setiap kelas nominal uang.

```

converter =
tf.lite.TFLiteConverter.from_keras_model(model)

converter.optimizations = [
    tf.lite.Optimize.DEFAULT
]

tflite_model = converter.convert()

with open(
    "/content/drive/MyDrive/model_uang.tflite",

```

```
"wb"
) as f:
    f.write(tflite_model)
```

Tahap terakhir adalah mengonversi model ke format TensorFlow Lite agar dapat digunakan pada aplikasi android. Proses konversi dilakukan dengan menerapkan optimasi model sehingga ukuran file menjadi lebih kecil dan lebih efisien saat dijalankan pada perangkat mobile. Hasil konversi berupa file `model_uang.tflite` yang selanjutnya diintegrasikan ke dalam aplikasi Flutter.

Pengembangan aplikasi dilakukan menggunakan framework Flutter dengan bahasa pemrograman Dart. Flutter dipilih karena mampu menghasilkan aplikasi yang ringan, responsif, dan mendukung pengembangan antarmuka pengguna secara fleksibel. Selain itu, Flutter juga mendukung integrasi berbagai package yang dibutuhkan dalam pengembangan sistem deteksi uang kertas rupiah berbasis kecerdasan buatan.

Sistem deteksi nominal uang pada penelitian ini memanfaatkan metode Convolutional Neural Network (CNN) sebagai model klasifikasi citra. CNN digunakan karena memiliki kemampuan yang baik dalam mengenali pola visual pada gambar, seperti warna, tekstur, bentuk, dan ciri khas dari masing – masing nominal uang kertas rupiah. Model CNN terlebih dahulu dilatih menggunakan dataset citra uang kertas rupiah yang terdiri dari beberapa kelas nominal, yaitu Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, Rp 100.000, serta kelas tambahan bukan uang.

```
await Tflite.loadModel(
    model: "assets/model.tflite",
    labels: "assets/labels.txt",
);
```

Kode program diatas digunakan untuk memuat model CNN dan file label klasifikasi ke dalam aplikasi android sebelum proses deteksi dilakukan. Setelah model berhasil dimuat, sistem dapat melakukan proses klasifikasi citra uang secara langsung pada perangkat mobile.

Model dikonversi ke dalam format TensorFlow Lite (.tflite) agar dapat diterapkan pada perangkat android dengan performa yang lebih ringan dan efisien. Implementasi model TensorFlow Lite pada aplikasi dilakukan menggunakan package `tflite_v2`.

```

final cameras = await availableCameras();

cameraController = CameraController(
    cameras[0],
    ResolutionPreset.medium,
    enableAudio: false,
);

```

Kode program diatas digunakan untuk menginisialisasi kamera smartphone sehingga aplikasi dapat mengambil citra uang kertas secara langsung. Selain implementasi model CNN, aplikasi juga memanfaatkan kamera smartphone sebagai media input citra uang kertas. Kamera diakses menggunakan package camera untuk melakukan proses scanning uang secara real-time.

Aplikasi yang dikembangkan terdiri dari beberapa halaman utama, yaitu splash screen, halaman mulai, dashboard, scan uang, panduan, dan pengaturan. Splash screen berfungsi sebagai tampilan pembuka aplikasi sekaligus proses inisialisasi sistem sebelum pengguna masuk ke halaman utama. Setelah itu pengguna diarahkan menuju halaman mulai yang memiliki tombol “Start” untuk masuk ke dashboard aplikasi.

Pada halaman dashboard tersedia beberapa menu utama, yaitu menu scan uang, panduan, dan pengaturan. Fitur scan uang merupakan fitur inti aplikasi yang digunakan untuk mendeteksi nominal uang kertas secara real-time menggunakan kamera smartphone. Ketika kamera diarahkan ke uang kertas, sistem akan melakukan proses klasifikasi menggunakan model CNN untuk menentukan nominal uang yang terdeteksi.

```

Future<void> initAll() async {
    _speech = stt.SpeechToText();
    await _speech.initialize();

    await tts.setLanguage("id-ID");
    await tts.setSpeechRate(0.5);
    await tts.awaitSpeakCompletion(true);

    await speak("Aplikasi siap. Ucapkan halo
    aplikasi untuk mulai");

    startListening();
}

```

Kode program diatas digunakan untuk membacakan hasil deteksi nominal uang melalui suara sehingga pengguna, khususnya penyandang tunanetra dapat mengetahui hasil klasifikasi tanpa harus melihat layar smartphone. Hasil deteksi nominal uang ditampilkan dalam bentuk teks pada layar aplikasi. Selain itu, aplikasi juga dilengkapi dengan fitur Text-to-Speech (TTS) menggunakan package flutter_tts untuk mengubah hasil deteksi menjadi output suara.

Selain fitur utama deteksi uang, aplikasi juga dilengkapi dengan menu panduan yang berisi informasi mengenai cara penggunaan aplikasi. Menu ini bertujuan untuk membantu pengguna memahami langkah – langkah penggunaan aplikasi dengan lebih mudah. Selanjutnya terdapat menu pengaturan yang berisi informasi aplikasi dan informasi pengembang.

Berdasarkan hasil implementasi, seluruh fitur utama aplikasi berhasil berjalan dengan baik pada perangkat android. Proses deteksi nominal uang dapat dilakukan secara real-time dengan waktu respon yang relatif cepat. Fitur Text-to-Speech juga berhasil menghasilkan suara sesuai dengan hasil klasifikasi model CNN.

b. Hasil Pengujian Model

Pengujian dilakukan untuk mengetahui kemampuan aplikasi dalam mengenali nominal uang kertas rupiah secara langsung menggunakan kamera smartphone. Pengujian dilakukan pada berbagai nominal uang dengan kondisi pencahayaan dan jarak yang berbeda. Setiap nominal diuji sebanyak 20 kali untuk mengetahui tingkat keberhasilan sistem dalam memberikan hasil deteksi yang sesuai.

Tabel 14 Hasil Pengujian Model Berdasarkan Jarak dan Cahaya

Nominal	10 cm	20 cm	30 cm	> 500 lux	100 – 500 lux	< 100 lux
Rp 1.000	✓	✓	✓	✓	✓	✓
Rp 2.000	✓	✓	✓	✓	✓	✓
Rp 5.000	✓	✓	✓	✓	✓	✓
Rp 10.000	✓	✓	✓	✓	✓	✓
Rp 20.000	✓	✓	✓	✓	✓	✓
Rp 50.000	✓	✓	✓	✓	✓	×
Rp 100.000	✓	✓	✓	✓	✓	×
bukan_uang	✓	✓	✓	✓	✓	✓

Berdasarkan hasil pengujian tabel diatas, sistem mampu mengenali seluruh nominal uang kertas rupiah dengan baik pada jarak 10 cm, 20 cm, dan 30 cm. Hasil

pengujian juga menunjukkan bahwa sistem bekerja optimal pada kondisi pencahayaan di atas 100 lux. Pada kondisi pencahayaan kurang dari 100 lux, sistem mengalami kesalahan deteksi pada nominal Rp 50.000 dan Rp 100.000 karena citra yang diterima kamera menjadi kurang jelas sehingga fitur – fitur penting pada uang tidak dapat dikenali secara optimal oleh model. Secara keseluruhan, sistem menunjukkan performa yang baik dalam mengenali nominal uang pada berbagai kondisi pengujian.

Tabel 15 Hasil Pengujian Model Berdasarkan Nominal

Nominal	Jumlah Pengujian	Berhasil	Gagal	Presentase
Rp 1.000	20	20	0	100%
Rp 2.000	20	20	0	100%
Rp 5.000	20	20	0	100%
Rp 10.000	20	20	0	100%
Rp 20.000	20	20	0	100%
Rp 50.000	20	18	2	90%
Rp 100.000	20	18	2	90%
bukan_uang	20	20	0	100%
Total	160	156	4	97.5%

Perhitungan :

$$\begin{aligned}
 \text{Presentase Keberhasilan} &= \frac{\text{Jumlah Berhasil}}{\text{Jumlah Pengujian}} \times 100\% \\
 &= \frac{156}{160} \times 100\% \\
 &= 97.5\%
 \end{aligned}$$

Berdasarkan tabel diatas, pengujian dilakukan sebanyak 20 kali pada setiap kelas, diperoleh total 160 kali pengujian. Nominal Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, dan bukan_uang berhasil dikenali dengan tingkat keberhasilan 100%. Sementara itu, nominal Rp 50.000 dan Rp 100.000 masing – masing mengalami dua kali kegagalan deteksi sehingga memperoleh tingkat keberhasilan sebesar 90%.

Kegagalan tersebut terjadi pada kondisi pencahayaan rendah yaitu < 100 lux. Kondisi pencahayaan yang kurang memadai menyebabkan fitur – fitur visual pada

uang tidak dapat ditangkap secara optimal oleh kamera sehingga memengaruhi hasil klasifikasi model.

Secara keseluruhan, sistem berhasil mengenali 156 data dari total 160 pengujian dengan tingkat keberhasilan sebesar 97.5%. Hasil tersebut menunjukkan bahwa sistem deteksi nominal uang yang dikembangkan mampu bekerja dengan baik dan memiliki tingkat akurasi yang tinggi pada pengujian secara langsung menggunakan aplikasi android.

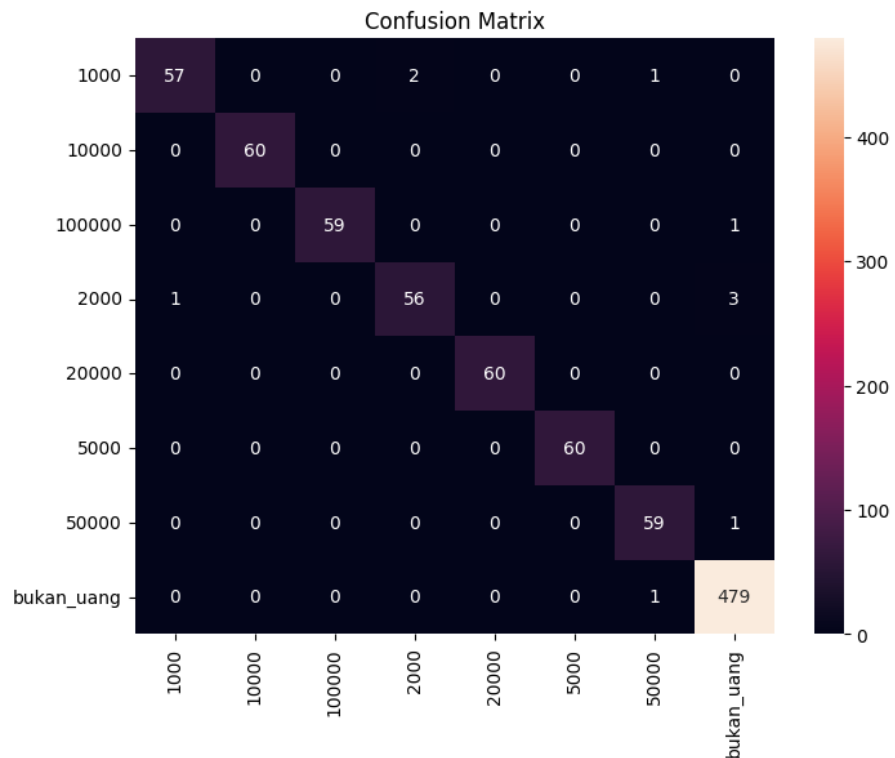
c. Hasil Pengujian Akurasi Model

Pengujian akurasi model dilakukan untuk mengetahui tingkat performa metode *Convolutional Neural Network (CNN)* dalam mengklasifikasi nominal uang kertas rupiah. Pengujian dilakukan menggunakan data testing yang tidak digunakan pada proses training model. Evaluasi model menggunakan metode Confusion Matrix dan Classification Report dengan parameter accuracy, precision, recall, dan F1-score.

Pada penelitian ini digunakan sebanyak 900 data testing yang terdiri dari beberapa kelas nominal uang kertas rupiah, yaitu Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, Rp 100.000 dan satu kelas tambahan yaitu bukan uang.

1. Hasil Confusion Matrix

Berdasarkan hasil pengujian model CNN menggunakan data testing, diperoleh confusion matrix yang menunjukkan hubungan antara data aktual dengan hasil prediksi model. Nilai pada diagonal utama menunjukkan jumlah data yang berhasil diprediksi dengan benar oleh model, sedangkan nilai di luar diagonal menunjukkan jumlah data yang mengalami kesalahan klasifikasi.



Gambar 24 Confusion Matrix

Berdasarkan hasil confusion matrix, sebagian besar data testing berhasil diklasifikasikan dengan benar oleh model CNN. Hal ini terlihat dari dominasi nilai pada diagonal utama confusion matrix. Kelas Rp 10.000, Rp 20.000, dan Rp 5.000 menunjukkan hasil klasifikasi yang sangat baik karena seluruh data pada kelas tersebut berhasil dikenali dengan benar tanpa kesalahan prediksi.

Pada kelas Rp 1.000, model berhasil mengklasifikasikan 57 data dengan benar dari total 60 data testing. Terdapat dua data yang salah diprediksi sebagai Rp 2.000 dan satu data yang salah diprediksi sebagai Rp 50.000. Selanjutnya pada kelas Rp 2.000, sebanyak 56 data berhasil dikenali dengan benar, sedangkan satu data salah diklasifikasikan sebagai Rp 1.000 dan tiga data lainnya diprediksi sebagai kelas bukan_uang.

Kelas Rp 50.000 dan Rp 100.000 masing – masing memiliki satu kesalahan klasifikasi, yaitu diprediksi sebagai kelas bukan_uang. Sementara itu, kelas bukan_uang menunjukkan performa yang sangat baik dengan 479 data berhasil dikenali dengan benar dari total 480 data testing, dan hanya satu data yang salah diprediksi sebagai Rp 50.000.

Hasil tersebut menunjukkan bahwa model CNN mampu mengenali karakteristik visual masing – masing nominal uang kertas rupiah dengan sangat baik. Kesalahan klasifikasi yang masih terjadi diduga dipengaruhi oleh variasi pencahayaan, posisi objek saat pengambilan gambar, serta kemiripan pola visual pada beberapa nominal uang.

2. Perhitungan Accuracy

Accuracy digunakan untuk mengetahui tingkat ketepatan model secara keseluruhan dalam melakukan klasifikasi terhadap seluruh data testing.

Tabel 16 Accuracy

Kelas	Prediksi Benar	Data Testing
Rp 1.000	57	60
Rp 2.000	56	60
Rp 5.000	60	60
Rp 10.000	60	60
Rp 20.000	60	60
Rp 50.000	59	60
Rp 100.000	59	60
bukan_uang	479	480
Total	890	900

Rumus :

$$Accuracy = \frac{\sum TP}{Total\ seluruh\ data}$$

Perhitungan :

$$Accuracy = \frac{890}{900} \times 100\% = 98.89\%$$

Jadi, nilai accuracy model adalah 98.89%

3. Perhitungan Precision

Precision digunakan untuk mengetahui ketepatan model saat memprediksi suatu kelas.

Rumus :

$$Precision = \frac{TP}{TP + FP}$$

- Precision kelas Rp 1.000

Diketahui :

TP : 57

FP : 1

Perhitungan :

$$Precision = \frac{57}{57 + 1} = 0.98$$

Precision Rp 1.000 = 0.98 atau 98%

- Precision kelas Rp 2.000

Diketahui :

TP : 56

FP : 2

Perhitungan :

$$Precision = \frac{56}{56 + 2} = 0.97$$

Precision Rp 2.000 = 0.97 atau 97%

- Precision kelas Rp 5.000

Diketahui :

TP : 60

FP : 0

Perhitungan :

$$Precision = \frac{60}{60 + 0} = 1.00$$

Precision Rp 5.000 = 1.00 atau 100%

- Precision kelas Rp 10.000

Diketahui :

TP : 60

FP : 0

Perhitungan :

$$Precision = \frac{60}{60 + 0} = 1.00$$

Precision Rp 10.000 = 1.00 atau 100%

- Precision kelas Rp 20.000

Diketahui :

TP : 60

FP : 0

Perhitungan :

$$Precision = \frac{60}{60 + 0} = 1.00$$

Precision Rp 20.000 = 1.00 atau 100%

- Precision kelas Rp 50.000

Diketahui :

TP : 59

FP : 2

Perhitungan :

$$Precision = \frac{59}{59 + 2} = 0.97$$

Precision Rp 50.000 = 0.97 atau 97%

- Precision kelas Rp 100.000

Diketahui :

TP : 59

FP : 0

Perhitungan :

$$Precision = \frac{59}{59 + 0} = 1.00$$

Precision Rp 100.000 = 1.00 atau 100%

- Precision kelas bukan_uang

Diketahui :

TP : 479

FP : 5

Perhitungan :

$$Precision = \frac{479}{479 + 5} = 0.99$$

Precision bukan_uang = 0.99 atau 99%

4. Perhitungan Recall

Recall digunakan untuk mengetahui kemampuan model mengenali seluruh data pada suatu kelas.

Rumus :

$$Recall = \frac{TP}{TP + FN}$$

- Recall kelas Rp 1.000

Diketahui :

TP : 57

FN : 3

Perhitungan :

$$Recall = \frac{57}{57 + 3} = 0.95$$

Recall Rp 1.000 = 0.95 atau 95%

- Recall kelas Rp 2.000

Diketahui :

TP : 56

FN : 4

Perhitungan :

$$Recall = \frac{56}{56 + 4} = 0.93$$

Recall Rp 2.000 = 0.93 atau 93%

- Recall kelas Rp 5.000

Diketahui :

TP : 60

FN : 0

Perhitungan :

$$Recall = \frac{60}{60 + 0} = 1.00$$

Recall Rp 5.000 = 1.00 atau 100%

- Recall kelas Rp 10.000

Diketahui :

TP : 60

FN : 0

Perhitungan :

$$Recall = \frac{60}{60 + 0} = 1.00$$

Recall Rp 10.000 = 1.00 atau 100%

- Recall kelas Rp 20.000

Diketahui :

TP : 60

FN : 0

Perhitungan :

$$Recall = \frac{60}{60 + 0} = 1.00$$

Recall Rp 20.000 = 1.00 atau 100%

- Recall kelas Rp 50.000

Diketahui :

TP : 59

FN : 1

Perhitungan :

$$Recall = \frac{59}{59 + 1} = 0.98$$

Recall Rp 50.000 = 0.98 atau 98%

- Recall kelas Rp 100.000

Diketahui :

TP : 59

FN : 1

Perhitungan :

$$Recall = \frac{59}{59 + 1} = 0.98$$

Recall Rp 100.000 = 0.98 atau 98%

- Recall kelas bukan_uang

Diketahui :

TP : 479

FN : 1

Perhitungan :

$$Recall = \frac{479}{479 + 1} = 1.00$$

Recall bukan_uang = 1.00 atau 100%

5. Perhitungan F1-Score

F1-Score digunakan untuk mengetahui keseimbangan antara precision dan recall.

Rumus :

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- F1-Score kelas Rp 1.000

Diketahui :

Precision : 0.98

Recall : 0.95

Perhitungan :

$$F1 - Score = 2 \times \frac{0.98 \times 0.95}{0.98 + 0.95} = 0.97$$

F1-Score Rp 1.000 = 0.97 atau 97%

- F1-Score kelas Rp 2.000

Diketahui :

Precision : 0.97

Recall : 0.93

Perhitungan :

$$F1 - Score = 2 \times \frac{0.97 \times 0.93}{0.97 + 0.93} = 0.95$$

F1-Score Rp 2.000 = 0.95 atau 95%

- F1-Score kelas Rp 5.000

Diketahui :

Precision : 1.00

Recall : 1.00

Perhitungan :

$$F1 - Score = 2 \times \frac{1.00 \times 1.00}{1.00 + 1.00} = 1.00$$

F1-Score Rp 5.000 = 1.00 atau 100%

- F1-Score kelas Rp 10.000

Diketahui :

Precision : 1.00

Recall : 1.00

Perhitungan :

$$F1 - Score = 2 \times \frac{1.00 \times 1.00}{1.00 + 1.00} = 1.00$$

F1-Score Rp 10.000 = 1.00 atau 100%

- F1-Score kelas Rp 20.000

Diketahui :

Precision : 1.00

Recall : 1.00

Perhitungan :

$$F1 - Score = 2 \times \frac{1.00 \times 1.00}{1.00 + 1.00} = 1.00$$

F1-Score Rp 20.000 = 1.00 atau 100%

- F1-Score kelas Rp 50.000

Diketahui :

Precision : 0.97

Recall : 0.98

Perhitungan :

$$F1 - Score = 2 \times \frac{0.97 \times 0.98}{0.97 + 0.98} = 0.98$$

F1-Score Rp 50.000 = 0.98 atau 98%

- F1-Score kelas Rp 100.000

Diketahui :

Precision : 1.00

Recall : 0.98

Perhitungan :

$$F1 - Score = 2 \times \frac{1.00 \times 0.98}{1.00 + 0.98} = 0.99$$

F1-Score Rp 100.000 = 0.99 atau 99%

- F1-Score kelas bukan_uang

Diketahui :

Precision : 0.99

Recall : 1.00

Perhitungan :

$$F1 - Score = 2 \times \frac{0.99 \times 1.00}{0.99 + 1.00} = 0.99$$

F1-Score bukan_uang = 0.99 atau 99%

6. Hasil Evaluasi Model

CLASSIFICATION REPORT:				
	precision	recall	f1-score	support
1000	0.98	0.95	0.97	60
10000	1.00	1.00	1.00	60
100000	1.00	0.98	0.99	60
2000	0.97	0.93	0.95	60
20000	1.00	1.00	1.00	60
5000	1.00	1.00	1.00	60
50000	0.97	0.98	0.98	60
bukan_uang	0.99	1.00	0.99	480
accuracy			0.99	900
macro avg	0.99	0.98	0.98	900
weighted avg	0.99	0.99	0.99	900
AKURASI VALIDASI FINAL: 98.89%				

Gambar 25 Classification Report

Berdasarkan hasil pengujian menggunakan Confusion Matrix, model CNN berhasil mencapai tingkat akurasi sebesar 98.89%. Nilai tersebut menunjukkan bahwa model mampu melakukan klasifikasi nominal uang kertas rupiah dengan tingkat kesalahan yang sangat rendah. Dari total 900 data testing, sebanyak 890 data berhasil diprediksi dengan benar, sedangkan hanya 10 data yang mengalami kesalahan klasifikasi.

Tingginya nilai akurasi menunjukkan bahwa proses pelatihan model menggunakan arsitektur MobileNetV2 dan teknik augmentasi data mampu meningkatkan kemampuan model dalam mengenali pola, warna, tekstur, dan karakteristik visual pada setiap nominal uang kertas rupiah. Selain itu, keberadaan kelas bukan_uang juga berhasil dikenali dengan sangat baik sehingga sistem mampu membedakan antara objek uang dan objek selain uang.

Secara keseluruhan, hasil pengujian membuktikan bahwa metode Convolutional Neural Network (CNN) yang diterapkan pada penelitian ini telah mampu memberikan performa klasifikasi yang sangat baik dan memenuhi tujuan penelitian sebagai sistem deteksi nominal uang kertas rupiah berbasis android untuk membantu penyandang tunanetra dalam mengenali nominal uang secara mandiri.

d. Hasil Pengujian Efektivitas Sistem

Pengujian efektivitas sistem dilakukan untuk mengetahui kemampuan aplikasi dalam menjalankan fungsi utamanya sebagai alat bantu pendeteksi nominal uang kertas rupiah secara real-time. Selain mengevaluasi tingkat keberhasilan deteksi, pengujian ini juga bertujuan untuk mengukur kecepatan aplikasi dalam memberikan hasil deteksi kepada pengguna. Oleh karena itu, efektivitas sistem pada penelitian ini diukur menggunakan dua indikator, yaitu tingkat keberhasilan deteksi dan response time aplikasi.

Pengukuran response time dilakukan mulai saat proses deteksi dimulai hingga aplikasi berhasil menampilkan hasil klasifikasi nominal uang beserta keluaran suara menggunakan fitur Text-to-Speech (TTS). Pengujian dilakukan pada seluruh nominal uang kertas rupiah yang digunakan dalam penelitian, yaitu Rp 1.000, Rp 2.000, Rp 5.000, Rp 10.000, Rp 20.000, Rp 50.000, dan Rp 100.000. Masing – masing nominal diuji sebanyak 7 kali sehingga total pengujian yang dilakukan sebanyak 49 kali.

Pengukuran waktu respon dilakukan pada aplikasi menggunakan fungsi `DateTime.now()` pada Flutter. Selisih waktu antara awal dan akhir proses deteksi dihitung menggunakan fungsi `difference().inMicroseconds`, kemudian dikonversi ke dalam satuan milidetik (ms) dengan membagi hasil pengukuran sebesar 1000,0. Dengan demikian, nilai response time menunjukkan waktu yang dibutuhkan aplikasi sejak proses deteksi dimulai hingga hasil berhasil ditampilkan kepada pengguna.

Tabel 17 Hasil Pengujian Efektivitas

No	Nominal	1	2	3	4	5	6	7	Rata – rata (ms)	Berhasil
1	Rp 1.000	528	541	536	532	538	530	535	534,3	7/7
2	Rp 2.000	547	553	549	551	545	550	548	549,0	7/7
3	Rp 5.000	514	520	517	516	519	515	518	517,0	7/7
4	Rp 10.000	501	506	504	503	507	505	502	504,0	7/7
5	Rp 20.000	522	527	524	525	523	526	524	524,4	7/7
6	Rp 50.000	612	619	615	621	617	614	618	616,6	7/7
7	Rp 100.000	648	652	650	655	649	651	653	651,1	7/7
Rata – rata keseluruhan									556,6 ms	49/49 (100%)

Perhitungan Response Time :

Nilai rata – rata response time dihitung menggunakan persamaan berikut

$$\bar{t} = \frac{\sum_{i=1}^n t_i}{n}$$

Keterangan :

- $\bar{t}$: rata – rata response time (ms)
- t_i : response time pada pengujian ke-i
- n : jumlah pengujian

Perhitungan :

$$\bar{t} = \frac{534,3 + 549,0 + 517,0 + 504,0 + 524,4 + 616,6 + 651,1}{7}$$

$$\bar{t} = 556,6 \text{ ms}$$

Berdasarkan hasil pengujian pada tabel diatas, aplikasi berhasil melakukan deteksi terhadap seluruh nominal uang kertas rupiah pada setiap pengujian yang dilakukan. Dari total 49 kali pengujian, seluruh proses deteksi berhasil dilakukan tanpa mengalami kegagalan, sehingga tingkat keberhasilan mencapai 100% pada pengujian efektivitas. Hasil ini menunjukkan bahwa integrasi antara kamera, model Convolutional Neural Network (CNN), TensorFlow Lite, dan fitur Text-to-Speech (TTS) telah berjalan dengan baik pada perangkat android.

Pengukuran response time menunjukkan bahwa aplikasi memiliki rata – rata waktu respon sebesar 556,6 milidetik atau sekitar 0,56 detik. Waktu tersebut menunjukkan bahwa aplikasi mampu melakukan proses pengambilan citra, klasifikasi menggunakan model CNN, hingga menampilkan hasil deteksi dan

menghasilkan keluaran suara dalam waktu kurang dari satu detik. Dengan waktu respon tersebut, aplikasi telah memenuhi karakteristik sistem real-time, yaitu mampu memberikan hasil deteksi secara cepat setelah objek diarahkan ke kamera.

Perbedaan response time antar nominal relatif kecil, yaitu berada pada rentang 504,0 ms hingga 651,1 ms. Nominal Rp 10.000 memiliki rata – rata waktu respon tercepat sebesar 504,0 ms, sedangkan nominal Rp 100.000 memiliki rata – rata waktu respon paling lama sebesar 651,1 ms. Perbedaan tersebut masih berada dalam batas yang wajar dan tidak memberikan pengaruh yang signifikan terhadap pengalaman pengguna ketika menggunakan aplikasi.

Secara keseluruhan, hasil pengujian efektivitas menunjukkan bahwa aplikasi yang dikembangkan mampu menjalankan fungsi utamanya secara optimal. Seluruh nominal uang berhasil dikenali dengan tingkat keberhasilan 100% pada pengujian efektivitas, sementara rata – rata waktu respon sebesar 556,6 ms menunjukkan bahwa aplikasi mampu memberikan hasil deteksi secara cepat. Hasil ini juga sejalan dengan pengujian akurasi model menggunakan Confusion Matrix yang memperoleh tingkat akurasi sebesar 98,89%. Dengan demikian, aplikasi deteksi nominal uang kertas rupiah berbasis CNN yang dikembangkan dapat dinyatakan efektif untuk digunakan sebagai alat bantu penyandang tunanetra dalam mengenali nominal uang secara real-time melalui perangkat android.

B. Pembahasan

a. Pengujian Fungsional Sistem

Pengujian fungsional sistem dilakukan untuk mengetahui kemampuan aplikasi dalam mendeteksi nominal uang kertas rupiah pada berbagai kondisi penggunaan. Pengujian dilakukan berdasarkan dua skenario utama, yaitu pengujian berdasarkan jarak kamera terhadap objek dan pengujian berdasarkan kondisi pencahayaan. Pengujian ini bertujuan untuk mengetahui tingkat performa sistem dalam kondisi nyata saat aplikasi digunakan oleh pengguna.

1. Pengujian Berdasarkan Jarak

Pengujian berdasarkan jarak dilakukan untuk mengetahui pengaruh jarak kamera terhadap kemampuan sistem dalam mendeteksi nominal uang kertas rupiah. Pada pengujian ini digunakan beberapa variasi jarak, yaitu jarak dekat (± 10 cm), jarak sedang (± 20 cm), jarak jauh (± 30 cm), dan jarak lebih dari 40 cm.

Berdasarkan hasil pengujian, sistem menunjukkan performa terbaik pada jarak dekat sekitar 10 cm hingga 15 cm. Pada jarak tersebut, sebagian besar nominal uang berhasil di deteksi dengan tingkat akurasi yang sangat tinggi, bahkan beberapa nominal mencapai akurasi hingga 100%. Hal tersebut menunjukkan bahwa model CNN mampu mengenali fitur visual uang secara optimal ketika objek berada pada jarak yang cukup dekat dengan kamera.

Pada jarak sedang sekitar 20 cm hingga 25 cm, sistem masih mampu melakukan deteksi nominal uang dengan baik, meskipun terjadi penurunan

tingkat akurasi pada beberapa kelas nominal. Penurunan tersebut disebabkan karena detail visual pada uang mulai berkurang sehingga proses ekstraksi fitur oleh model CNN menjadi kurang optimal.

Pada pengujian jarak sekitar 30 cm, performa sistem mengalami penurunan yang lebih signifikan. Beberapa nominal uang menunjukkan nilai akurasi yang lebih rendah dibandingkan pengujian jarak dekat dan sedang. Hal ini terjadi karena objek uang terlihat lebih kecil pada kamera sehingga detail pola, warna, dan tekstur uang menjadi lebih sulit dikenali oleh model CNN.

Sementara itu, pada jarak lebih dari 40 cm sistem mengalami kesulitan dalam mendeteksi nominal uang secara akurat. Berdasarkan hasil pengujian, sebagian besar nominal uang tidak dapat terdeteksi dengan baik. Kondisi tersebut menunjukkan bahwa jarak kamera sangat mempengaruhi kualitas citra yang diterima sistem. Semakin jauh jarak kamera terhadap objek, maka semakin rendah detail citra yang diperoleh sehingga kemampuan klasifikasi model CNN ikut menurun.

Berdasarkan hasil pengujian tersebut dapat diketahui bahwa jarak optimal untuk melakukan proses deteksi nominal uang berada pada kisaran 10 cm hingga 20 cm. Pada rentang jarak tersebut sistem mampu memberikan hasil deteksi yang lebih stabil dan akurat.

2. Pengujian Berdasarkan Cahaya

Selain pengujian jarak, dilakukan pula pengujian berdasarkan kondisi pencahayaan untuk mengetahui pengaruh intensitas cahaya terhadap performa sistem dalam mendeteksi nominal uang kertas rupiah. Pengujian dilakukan pada empat kondisi pencahayaan, yaitu cahaya terang, cahaya normal, cahaya redup, dan cahaya sangat rendah.

Berdasarkan hasil pengujian, sistem menunjukkan performa terbaik pada kondisi pencahayaan terang dengan intensitas cahaya di atas 500 lux. Pada kondisi tersebut, sebagian besar nominal uang berhasil dikenali dengan tingkat akurasi yang tinggi. Hal ini disebabkan karena pencahayaan yang cukup mampu menghasilkan citra yang lebih jelas sehingga fitur visual uang dapat dikenali dengan baik oleh model CNN.

Pada kondisi pencahayaan normal dengan rentang 100 – 500 lux, sistem masih mampu melakukan deteksi nominal uang dengan cukup baik. Meskipun terjadi sedikit penurunan akurasi pada beberapa nominal, secara umum sistem masih dapat mengenali objek uang dengan stabil.

Pada kondisi pencahayaan redup dengan rentang 10 – 100 lux, performa sistem mulai mengalami penurunan. Beberapa nominal uang menunjukkan hasil deteksi yang kurang stabil dan tingkat akurasi yang lebih rendah. Hal tersebut terjadi karena kondisi cahaya yang rendah menyebabkan citra menjadi kurang jelas, sehingga proses ekstraksi fitur oleh model CNN menjadi tidak optimal.

Pada kondisi pencahayaan sangat rendah di bawah 10 lux, sistem mengalami kesulitan dalam mendeteksi nominal uang. Pada kondisi ini, kamera tidak mampu menangkap detail visual objek secara jelas sehingga

model CNN kesulitan mengenali pola dan karakteristik uang kertas rupiah. Akibatnya, proses klasifikasi menjadi tidak akurat atau bahkan gagal mendeteksi objek.

Hasil pengujian tersebut menunjukkan bahwa pencahayaan merupakan salah satu faktor penting yang mempengaruhi performa sistem deteksi berbasis citra. Semakin baik kualitas pencahayaan, maka semakin tinggi pula kemampuan model CNN dalam mengenali objek uang kertas rupiah.

3. Analisis Kinerja Sistem

Berdasarkan hasil pengujian jarak dan pencahayaan, dapat diketahui bahwa sistem deteksi nominal uang kertas rupiah berbasis CNN telah mampu bekerja dengan baik pada kondisi penggunaan normal. Sistem berhasil melakukan proses klasifikasi secara real-time dan menghasilkan output berupa teks dan suara sesuai dengan hasil deteksi.

Implementasi model CNN menggunakan TensorFlow Lite juga berhasil berjalan dengan baik pada perangkat android. Sistem mampu melakukan proses inferensi dengan waktu respon yang relatif cepat sehingga aplikasi dapat digunakan secara langsung oleh pengguna tanpa memerlukan koneksi internet.

Meskipun demikian, hasil pengujian menunjukkan bahwa performa sistem masih dipengaruhi oleh faktor eksternal, seperti jarak kamera, kondisi pencahayaan, kualitas kamera smartphone, serta posisi objek saat proses scanning dilakukan. Oleh karena itu, pengguna perlu memperhatikan kondisi penggunaan agar sistem dapat menghasilkan deteksi yang lebih optimal.

Secara keseluruhan, hasil pengujian fungsional sistem menunjukkan bahwa aplikasi yang dikembangkan telah berhasil memenuhi tujuan penelitian, yaitu mampu mendeteksi nominal uang kertas rupiah secara otomatis dan membantu pengguna, khususnya penyandang tunanetra, dalam mengenali nominal uang melalui teknologi deteksi citra dan output suara.

b. Analisis Performa Model CNN

Evaluasi model dilakukan untuk mengetahui tingkat performa metode Convolutional Neural Network (CNN) dalam mengklasifikasikan nominal uang kertas rupiah berdasarkan citra yang digunakan pada data testing. Proses evaluasi dilakukan menggunakan beberapa parameter, yaitu confusion matrix, accuracy, precision, recall, dan F1-score. Pengujian dilakukan menggunakan 900 data testing yang terdiri dari beberapa kelas nominal uang kertas rupiah dan satu kelas tambahan, yaitu bukan uang.

Berdasarkan hasil evaluasi model, metode Convolutional Neural Network (CNN) menunjukkan performa yang sangat baik dalam melakukan klasifikasi nominal uang kertas rupiah. Hal tersebut ditunjukkan oleh nilai accuracy model sebesar 98,89%, yang menunjukkan bahwa sebagian besar data testing berhasil diklasifikasikan dengan benar oleh sistem. Dari total 900 data testing, sebanyak 890 data berhasil diprediksi dengan benar oleh model CNN.

Pada hasil confusion matrix, sebagian besar kelas nominal uang berhasil dikenali dengan baik oleh model CNN. Nilai precision, recall, dan F1-score yang tinggi menunjukkan kemampuan model dalam membedakan setiap kelas nominal uang dengan sangat baik. Kelas Rp 10.000, Rp 20.000, dan Rp 50.000 menunjukkan hasil klasifikasi yang sangat baik dengan nilai precision dan recall mencapai 100%. Hal tersebut menunjukkan bahwa model mampu mengenali karakteristik visual pada nominal tersebut secara optimal.

Namun, masih terdapat beberapa kesalahan klasifikasi pada kelas Rp 1.000, Rp 2.000, Rp 100.000, dan kelas bukan uang. Kesalahan tersebut disebabkan oleh adanya kemiripan pola visual antar kelas, kualitas citra yang kurang optimal, serta kondisi pengambilan gambar yang mempengaruhi proses ekstraksi fitur oleh model CNN. Selain itu, faktor pencahayaan dan posisi objek saat proses pengambilan gambar juga mempengaruhi hasil klasifikasi sistem.

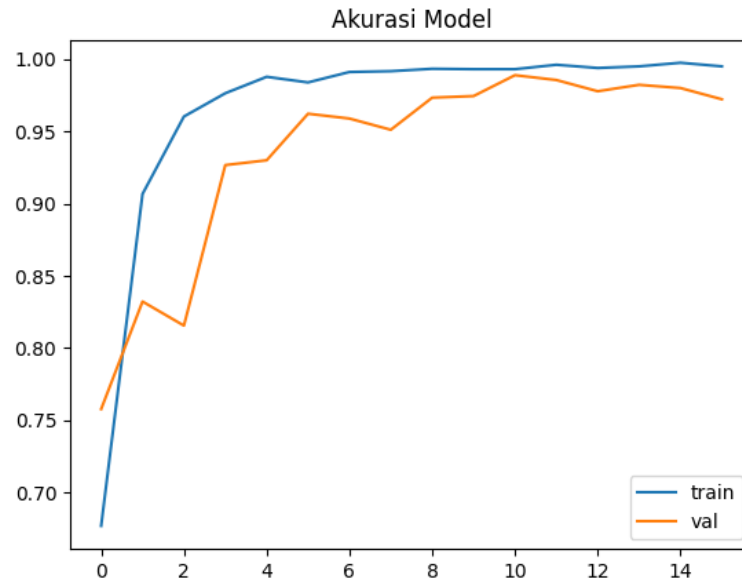
Nilai accuracy sebesar 98,89% menunjukkan bahwa model CNN memiliki tingkat ketepatan yang sangat baik dalam melakukan klasifikasi nominal uang kertas rupiah secara keseluruhan. Tingginya nilai accuracy dipengaruhi oleh kemampuan CNN dalam melakukan ekstraksi fitur citra secara otomatis, seperti warna dominan, pola gambar, tekstur, dan tata letak desain pada masing – masing nominal uang. Selain itu, penggunaan dataset yang cukup beragam membantu model dalam mempelajari berbagai variasi citra uang kertas sehingga kemampuan generalisasi model terhadap data baru menjadi lebih baik.

Hasil pengujian precision menunjukkan bahwa sebagian besar kelas memiliki nilai precision di atas 97%. Kelas Rp 10.000, Rp 20.000, Rp 50.000, dan Rp 100.000 memperoleh nilai precision sebesar 100%, sedangkan kelas Rp 1.000, Rp 2.000, dan bukan uang juga menunjukkan nilai precision yang sangat baik, masing – masing sebesar 98%, 97%, dan 99%. Tingginya nilai precision menunjukkan bahwa model CNN mampu memprediksi nominal uang secara tepat dengan tingkat kesalahan prediksi yang rendah. Hal tersebut menunjukkan bahwa fitur visual pada uang kertas berhasil dipelajari dengan baik selama proses training model.

Hasil pengujian recall juga menunjukkan performa model yang sangat baik. Sebagian besar kelas memperoleh nilai recall yang tinggi, bahkan beberapa kelas mencapai nilai 100%, seperti Rp 10.000, Rp 20.000, Rp 50.000, dan kelas bukan uang. Hal tersebut menunjukkan bahwa model mampu mengenali sebagian besar data aktual pada masing – masing kelas nominal uang dengan baik. Namun, kelas Rp 2.000 memperoleh nilai recall sebesar 93%, yang menunjukkan bahwa masih terdapat beberapa citra uang Rp 2.000 yang salah dikenali sebagai nominal uang tertentu. Kondisi tersebut dapat terjadi karena adanya kemiripan pola visual antara objek bukan uang dengan karakteristik tertentu pada uang kertas rupiah.

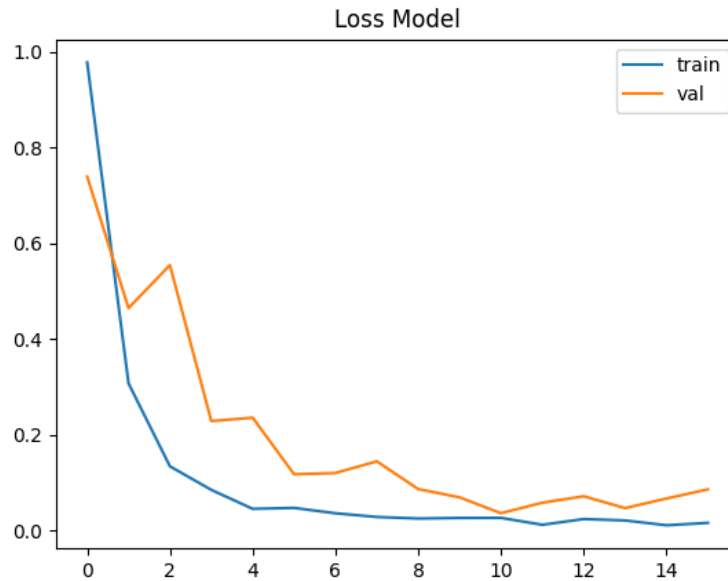
Hasil pengujian F1-score juga menunjukkan performa model yang sangat baik. Sebagian besar kelas memperoleh nilai F1-score di atas 95%, bahkan kelas Rp

10.000, Rp 20.000, dan Rp 50.000 memperoleh nilai F1-score sebesar 100%. Nilai F1-score yang tinggi menunjukkan bahwa model memiliki keseimbangan yang baik antara precision dan recall sehingga model mampu melakukan klasifikasi secara stabil dan konsisten.



Gambar 26 Grafik Akurasi Model

Berdasarkan grafik accuracy model, terlihat bahwa nilai training accuracy dan validation accuracy mengalami peningkatan pada setiap epoch. Hal tersebut menunjukkan bahwa model CNN mampu mempelajari pola data dengan baik selama proses pelatihan. Nilai accuracy yang meningkat dari sekitar 67,8% pada epoch awal menjadi 99,5% pada epoch terakhir, sedangkan validation accuracy meningkat dari sekitar 75,8% menjadi 97,2%. Nilai accuracy yang meningkat menunjukkan bahwa kemampuan model dalam melakukan klasifikasi citra uang semakin baik seiring bertambahnya proses training.



Gambar 27 Grafik Loss Model

Pada grafik loss model terlihat bahwa nilai training loss dan validation loss mengalami penurunan secara bertahap pada setiap epoch. Nilai training loss menurun dari sekitar 0,98 menjadi 0,02, sedangkan validation loss menurun dari sekitar 0,74 menjadi 0,09. Penurunan nilai loss menunjukkan bahwa tingkat kesalahan model dalam melakukan prediksi semakin kecil selama proses training berlangsung. Selain itu, perbedaan antara training accuracy dan validation accuracy tidak terlalu jauh, begitu pula pada training loss dan validation loss. Kondisi tersebut menunjukkan bahwa model tidak mengalami overfitting secara berlebihan dan masih memiliki kemampuan generalisasi yang baik terhadap data baru.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa metode Convolutional Neural Network (CNN) memiliki performa yang sangat baik dalam mengklasifikasikan nominal uang kertas rupiah. Model mampu melakukan klasifikasi dengan tingkat accuracy sebesar 98,89% serta nilai precision, recall, dan F1-score yang tinggi pada sebagian besar kelas nominal uang. Hasil tersebut menunjukkan bahwa model CNN berhasil mempelajari fitur – fitur visual pada uang kertas rupiah dengan baik sehingga mampu mengenali nominal uang secara akurat pada data testing. Dengan performa yang diperoleh, model CNN dinilai layak untuk diimplementasikan pada aplikasi deteksi nominal uang kertas rupiah berbasis android.

BAB V KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian, perancangan, implementasi, dan pengujian aplikasi deteksi nominal uang kertas rupiah berbasis Convolutional Neural Network (CNN), maka dapat diperoleh kesimpulan sebagai berikut :

1. Aplikasi deteksi nominal uang kertas rupiah berbasis CNN berhasil dirancang dan diimplementasikan pada perangkat android. Aplikasi mampu melakukan deteksi nominal uang secara real – time menggunakan kamera smartphone serta menghasilkan output suara melalui fitur Text-to-Speech (TTS) sehingga dapat membantu pengguna, khususnya penyandang tunanetra dalam mengenali nominal uang kertas secara mandiri.
2. Berdasarkan hasil pengujian sistem, aplikasi mampu mengenali nominal uang kertas rupiah dengan tingkat keberhasilan sebesar 98,89%. Hasil tersebut menunjukkan bahwa model memiliki kemampuan klasifikasi yang sangat baik dalam membedakan setiap nominal uang kertas rupiah dengan tingkat kesalahan yang rendah.
3. Hasil pengujian menunjukkan bahwa sistem dapat bekerja dengan baik pada jarak pengambilan gambar 10 cm, 20 cm, dan 30 cm. Selain itu, sistem mampu mendeteksi nominal uang secara optimal pada kondisi pencahayaan lebih dari 100 lux. Namun, pada kondisi pencahayaan rendah di bawah 100 lux, sistem mengalami beberapa kesalahan deteksi karena kualitas citra yang ditangkap kamera menjadi kurang jelas sehingga memengaruhi proses klasifikasi oleh model CNN.

B. Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat dijadikan bahan pengembangan pada penelitian selanjutnya, yaitu sebagai berikut :

1. Berdasarkan hasil penelitian yang telah dilakukan, sistem dapat dikembangkan lebih lanjut agar mampu mendeteksi berbagai tahun emisi uang kertas rupiah yang beredar di Indonesia. Dengan menambahkan dataset dari berbagai tahun emisi, model diharapkan dapat mengenali lebih banyak variasi desain uang kertas sehingga hasil deteksi menjadi lebih akurat dan aplikasi dapat digunakan secara lebih luas oleh pengguna.

DAFTAR PUSTAKA

- [1] A. Darmawan, I. G. N. G. A. Widyadhana, and E. H. Binugroho, "Implementasi Metode Deep Learning Pada Prototipe Validator Uang Rupiah," *Sebatik*, vol. 26, no. 2, pp. 535–542, Dec. 2022, DOI: 10.46984/sebatik.v26i2.2101.
- [2] D. Darmawan, "Rancang Bangun Alat Pendeteksi Uang Untuk Penyandang Tuna Netra," 2022.
- [3] A. Bahri, J. Fisika, and F. Matematika dan Ilmu Pengetahuan Alam, "Sistem Pendeteksi Keaslian dan Nominal Uang untuk Penyandang Tunanetra Menggunakan Sensor UV GYML 8511 dan TCS3200," vol. 12, no. 2, pp. 317–322, 2023, DOI: 10.25077/jfu.12.2.317-322.2023.
- [4] A. Zuhair, K. T. Pambudi, M. Syukron, A. Prasetyaningtyas, and T. Baggas Pratama, "Sistem Pendeteksi Keaslian Nominal Uang Kertas Untuk Tunanetra Dengan Sensor Warna Dan Sistem Suara," 2025.
- [5] Fadlioni and Kusnoto, "Rancang Bangun dan Implementasi Alat Pendeteksi Nilai Uang untuk Tuna Netra Menggunakan Mikrokontroler Arduino," vol. 6, no. 1.
- [6] N. Fatmawati, dan Yuli Amalia, and U. Bina Bangsa Getsempena, "Pengaruh Template Matching Correlation Terhadap Tunanetra Dalam Mengenai Uang Kertas Rupiah Di Kota Banda Aceh," *Jurnal Ilmiah Mahasiswa*, vol. 3, no. 2, 2022.
- [7] M. Adias Sabara, B. Niam, R. Darpono, P. Studi DIII Teknik Elektronika Politeknik Harpan Bersama Tegal Jl Mataram No, and K. Tegal, "Alat Deteksi Nominal Uang Kertas Untuk Penyandang Tuna Netra Dengan Metode Deteksi Warna Banknote Nominal Detection Tool For The Blind Person Using Color Detection Method," 2023. [Online]. Available: <http://jurnalnasional.ump.ac.id/index.php/JRRE>
- [8] A. Muhammad Nur Hidayat and I. M. Zakiyah, "Identifikasi Nominal Mata Uang Rupiah Bagi Penyandang Tunanetra Dengan Algoritma Convolutional Neural Network Berbasis Android," *JOURNAL SHIFT VOL*, vol. 3, 2023.
- [9] M. Malik Ibrahim, R. Rahmadewi, and L. Nurpulaela, "Pendeteksian Nominal Uang Pada Gambar Menggunakan Convolutional Neural Network: Integrasi Metode Pra-Pemrosesan Citra dan Klasifikasi Berbasis CNN," 2023.
- [10] F. Industri Kreatif and I. Teknologi dan Bisnis Kalbis Jalan Pulomas Selatan Kav, "Pengembangan Model Pembelajaran Mesin untuk Mengidentifikasi Nominal Uang Kertas Rupiah Menggunakan Algoritma CNN Dengan Library Keras I Made Gede Indra Mahendra," 2022.
- [11] P. Ayu, L. Zakaria, and M. Zainal, "Pengenalan Nilai Mata Uang Kertas Untuk Tunanetra Berbasis Android," 2023. [Online]. Available: <https://jurnal.umpar.ac.id/index.php/sylog#40>
- [12] D. Khairu, F. Wantasen, A. Yusupa, and V. Taringan, "Klasifikasi Jenis Uang Kertas Menggunakan Convolutional Neural Network A Convolutional Neural Network-Based

Approach For Classifying Types Of Paper Currency,” *VARIABLE RESEARCH JOURNAL*, vol. 02, 2025.

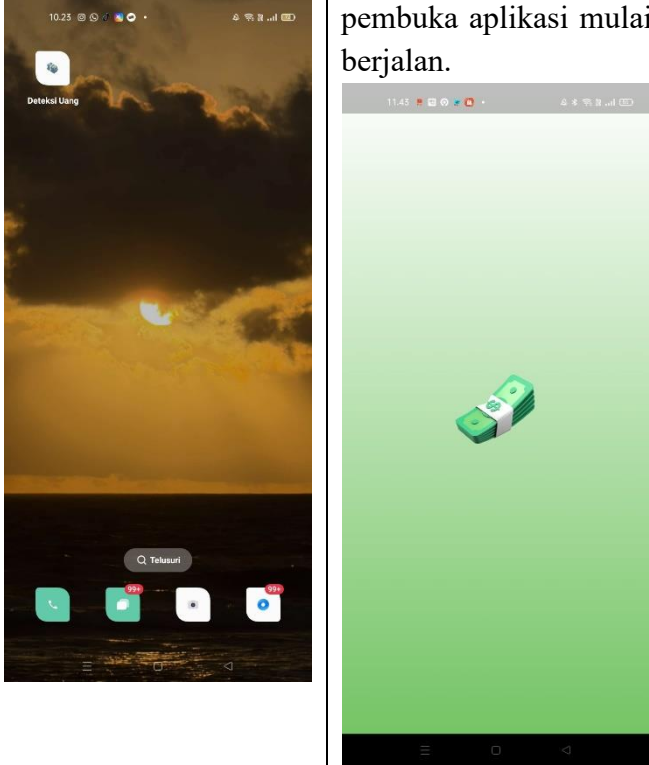
- [13] K. Maulana Azhar, I. Santoso, D. Yosua, and A. A. Soetrisno, “Implementasi Deep Learning Menggunakan Metode Convolutional Neural Network Dan Algoritma Yolo Dalam Sistem Pendeteksi Uang Kertas Rupiah Bagi Penyandang Low Vision,” 2021. [Online]. Available: <https://ejournal3.undip.ac.id/index.php/transient>
- [14] A. P. Nandika, M. Imanullah, A. Wijaya, and D. Sunardi, “Deteksi Kondisi Uang Bagus Dan Rusak Dengan Pengolahan Citra Digital Berbasis Convolutional Neural Network (CNN),” *Jurnal Media Infotama*, vol. 21, no. 1, 2025.
- [15] R. Daniel Yc Raban *et al.*, “Model Pendeteksi Nominal Uang Kertas Rupiah Menggunakan Teknologi Optical Character Recognition,” *Jurnal TICOM: Technology of Information and Communication*, vol. 12, no. 1, 2023.
- [16] D. Rafikil, M. Basri, U. Suwardoyo, A. Wafiah, and M. Yunus, “Pendeteksi Nominal Mata Uang Kertas menggunakan Kamera berbasis Android,” *Jurnal Manajemen Informatika, Sistem Informasi dan Teknologi Komputer (JUMISTIK)*, vol. 4, no. 2, pp. 495–504, Dec. 2025, doi: 10.70247/jumistik.v4i2.170.
- [17] M. R. Roshan and H. Irsyad, “Klasifikasi Kerusakan Uang Rupiah Menggunakan CNN Dengan Arsitektur VGG16,” *Arcitech: Journal of Computer Science and Artificial Intelligence*, vol. 5, no. 2, pp. 221–242, Dec. 2025, doi: 10.29240/arcitech.v5i2.15125.
- [18] N. Rinaldi, M. Syahroni, J. Teknik Elektro, and P. Negeri Lhokseumawe, “Analisis Perbedaan Metode Canny Edge dan Region Segmentation Active Contour Pada Aplikasi Deteksi Watermark Pada Uang Kertas,” 2023.
- [19] V. Putra Alegria *et al.*, “86 Implementasi Digital Watermaking Dengan Metode Discrete Cosine Transform (DCT) Pada Citra Digital,” 2022.
- [20] A. Hermawan, L. Lianata, Junaedi, and A. R. K. Maranto, “Implementasi Machine Learning Sebagai Pengenal Nominal Uang Rupiah dengan Metode YOLOv3,” *SATIN - Sains dan Teknologi Informasi*, vol. 8, no. 1, pp. 12–22, Jun. 2022, doi: 10.33372/stn.v8i1.816.
- [21] G. F. Hanin and T. Dewayanto, “Peran Machine Learning dan Deep Learning Dalam Pendeteksian Pencucian Uang-A Systematic Literature Review,” *DIPONEGORO JOURNAL OF ACCOUNTING*, vol. 13, no. 3, pp. 1–11, 2024, [Online]. Available: <http://ejournal-s1.undip.ac.id/index.php/accounting>
- [22] B. N. Susanto, M. A. C. Nasrulah, Y. G. Endriawan, and A. Aliyu, “Efektivitas Teachable Machine dan Raspberry Pi dalam Mendeteksi Nominal Uang Kertas pada Alat Pendeteksi Model Virtual Reality,” *Jurnal Penelitian Inovatif*, vol. 4, no. 3, pp. 1731–1740, Aug. 2024, doi: 10.54082/jupin.634.
- [23] M. A. Karim and A. R. Adriansyah, “Analisis dan Perancangan Aplikasi Mobile Untuk Donasi Menggunakan Metode Hybrid Berbasis React Native,” *Jurnal Informatika Terpadu*, vol. 8, no. 1, pp. 26–34, 2022, [Online]. Available: <https://journal.nurulfikri.ac.id/index.php/JIT>

- [24] D. P. Nugrohos and W. S. Huda, “Rancang Bangun Aplikasi Mobile Money Changer Menggunakan Metode Waterfall Berbasis Android,” 2024.
- [25] Y. Apridiansyah, “Aplikasi Antrian Pembayaran Uang Kuliah Berbasis Android Menggunakan Algoritma Fifo Di Universitas Muhammadiyah Bengkulu,” 2021.
- [26] N. ' Arrizqi, I. Santoso, D. Yosua, and A. A. Soetrisno, “Implementasi Google Text To Speech Pada Aplikasi Pendeteksi Uang Berbasis Android,” 2021. [Online]. Available: <https://ejournal3.undip.ac.id/index.php/transient>

LAMPIRAN

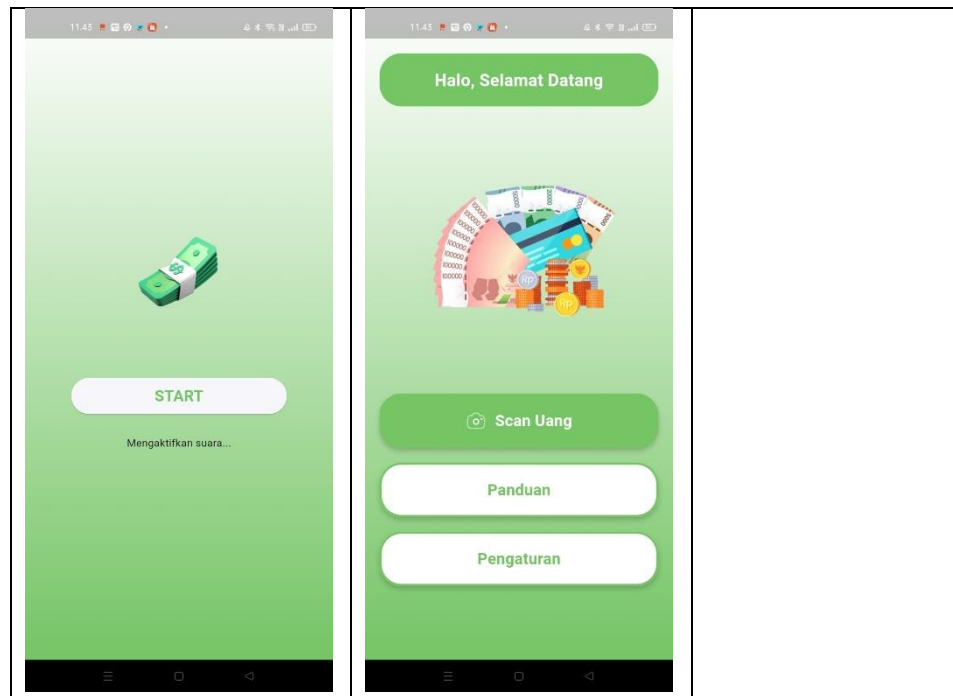
1. Hasil Pengujian Fungsional Aplikasi

a. Pengujian pada menu *splash*

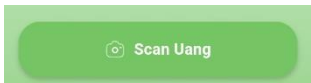
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Tampilan splash screnn 	Menampilkan logo aplikasi sebagai pembuka aplikasi mulai berjalan.	Berhasil

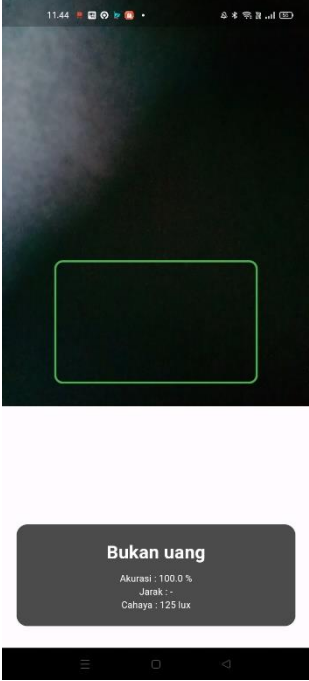
b. Pengujian pada menu masuk

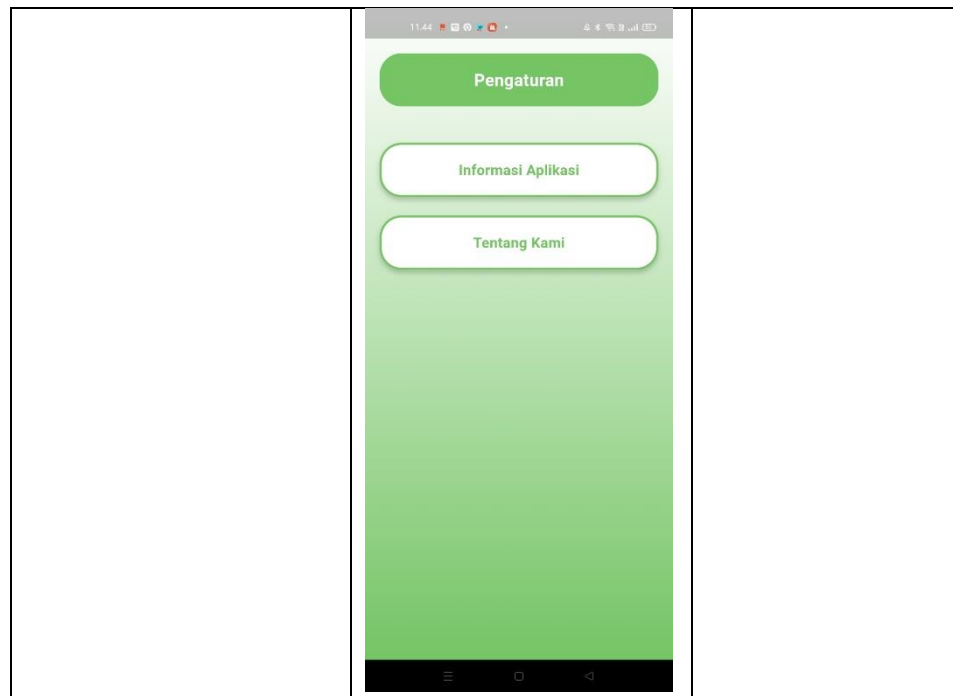
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses button masuk	Masuk berhasil dan mengakses dashboard	Berhasil



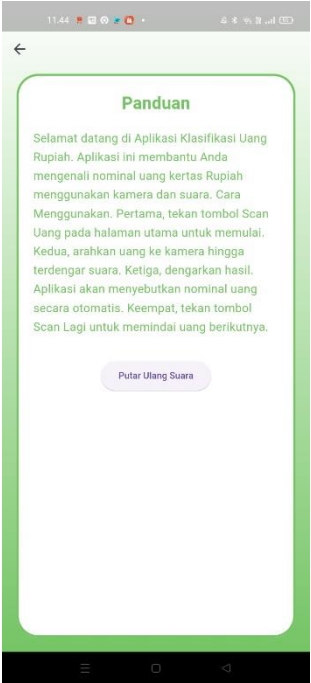
c. Pengujian pada menu dashboard

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses tampilan dashboard 	Menampilkan menu utama 	Berhasil
Akses scan uang 	Menampilkan kamera, mengeluarkan hasil teks dan suara	Berhasil

		
Akses panduan 	Menampilkan teks dan suara panduan 	Berhasil
Akses pengaturan 	Menampilkan button 'Informasi Aplikasi' dan 'Tentang Kami'	Berhasil



d. Pengujian pada menu panduan

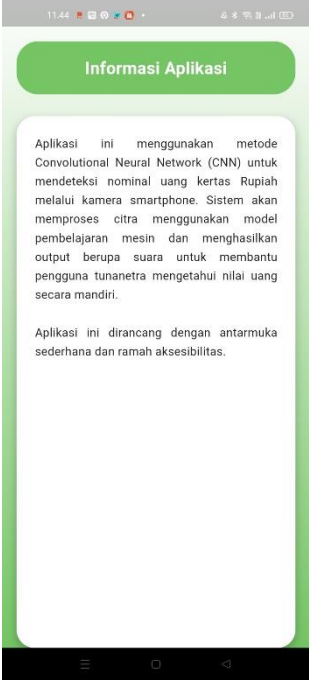
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses panduan 	Menampilkan teks dan suara panduan 	Berhasil

e. Pengujian pada menu pengaturan

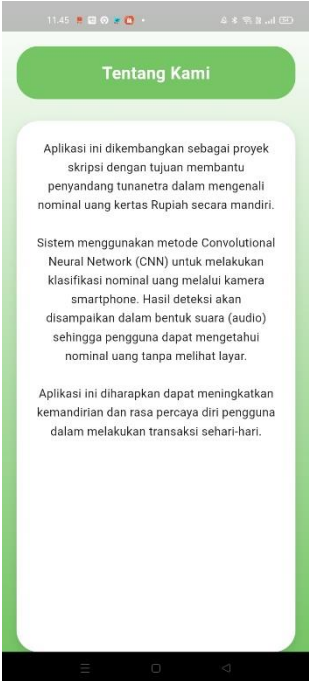
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses pengaturan 	Menampilkan button 'Informasi Aplikasi' dan 'Tentang Kami' 	Berhasil

f. Pengujian pada menu informasi aplikasi

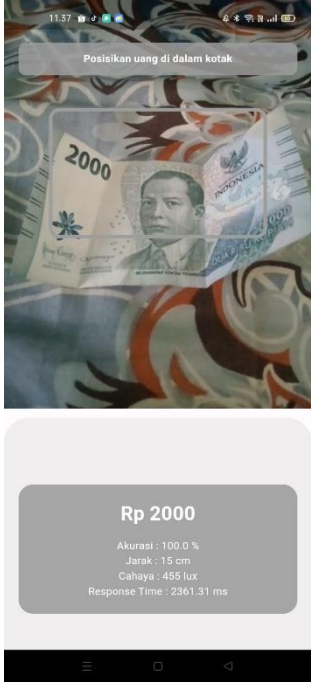
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses informasi aplikasi 	Menampilkan teks dan suara informasi aplikasi.	Berhasil

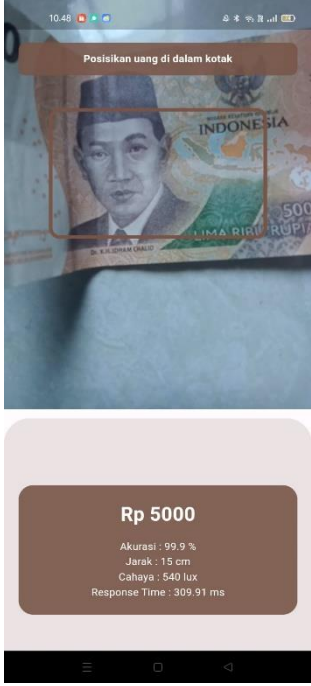
		
--	--	--

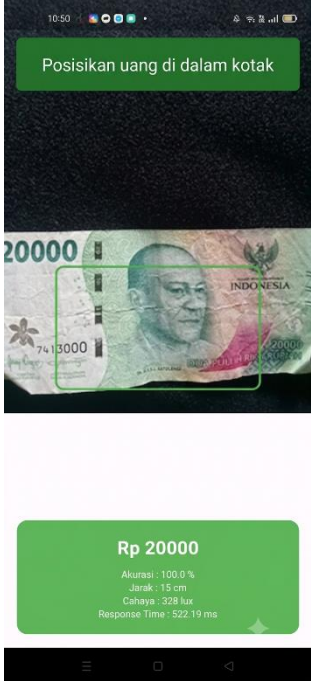
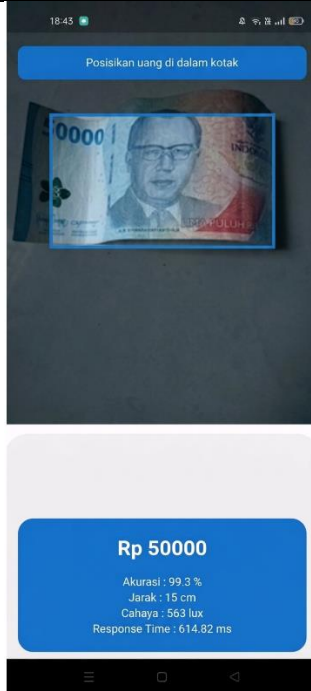
g. Pengujian pada menu tentang kami

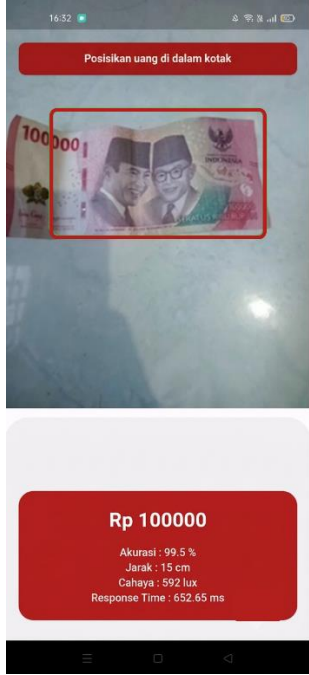
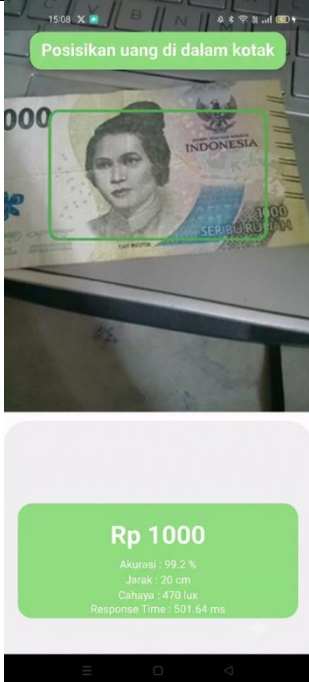
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses tentang kami 	Menampilkan teks dan suara tentang kami 	Berhasil

2. Pengujian Sistem Berdasarkan Jarak

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
± 10 cm	15 cm	Rp 1.000		99.5%	Berhasil
	15 cm	Rp 2.000		100.0%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	15 cm	Rp 5.000	 The screenshot shows a mobile app interface for detecting Indonesian Rupiah. At the top, a status bar shows the time 10:48. Below it, a purple banner reads 'Posisikan uang di dalam kotak' (Position the money inside the box). The main image is a 5000 Rupiah banknote with a red bounding box around the portrait. Below the image, a white box displays the results: 'Rp 5000', 'Akurasi : 99.9 %', 'Jarak : 15 cm', 'Cahaya : 540 lux', and 'Response Time : 309.91 ms'.	99.9%	Berhasil
	15 cm	Rp 10.000	 The screenshot shows the same mobile app interface. The status bar shows the time 18:41. The purple banner reads 'Posisikan uang di dalam kotak'. The main image is a 10000 Rupiah banknote with a red bounding box around the portrait. Below the image, a white box displays the results: 'Rp 10000', 'Akurasi : 99.2 %', 'Jarak : 15 cm', 'Cahaya : 542 lux', and 'Response Time : 506.84 ms'.	99.2%	Berhasil

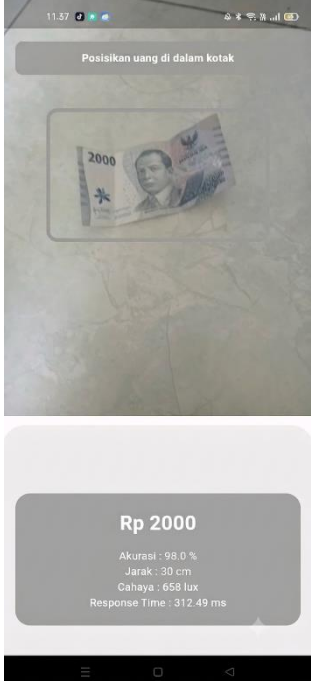
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	15 cm	Rp 20.000		100.0%	Berhasil
	15 cm	Rp 50.000		99.3%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	15 cm	Rp 100.000		99.5%	Berhasil
± 20 cm	20 cm	Rp 1.000		99.2%	Berhasil

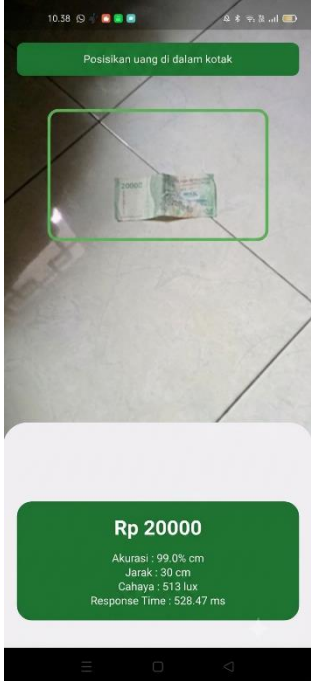
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	20 cm	Rp 2.000		99.5%	Berhasil
	20 cm	Rp 5.000		99.2%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	25 cm	Rp 10.000	 	99.5%	Berhasil
	23 cm	Rp 20.000	 	98.8%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	25 cm	Rp 50.000		99.0%	Berhasil
	17 cm	Rp 100.000		99.1%	Berhasil

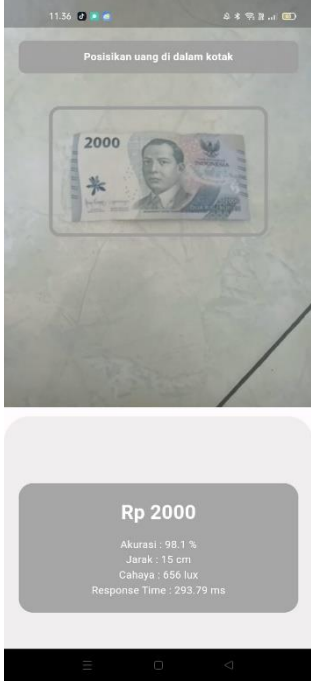
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
± 30 cm	30 cm	Rp 1.000		98.2%	Berhasil
	30 cm	Rp 2.000		98.0%	Berhasil

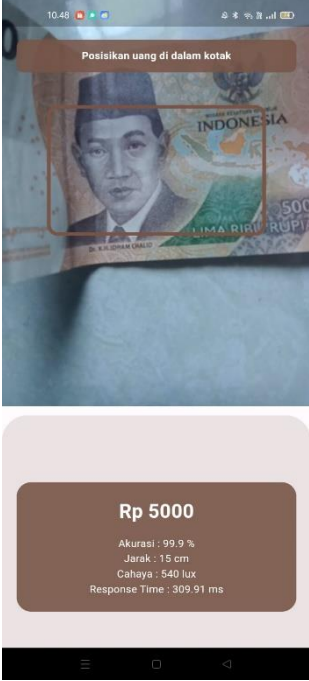
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	30 cm	Rp 5.000		97.5%	Berhasil
	31 cm	Rp 10.000		99.5%	Berhasil

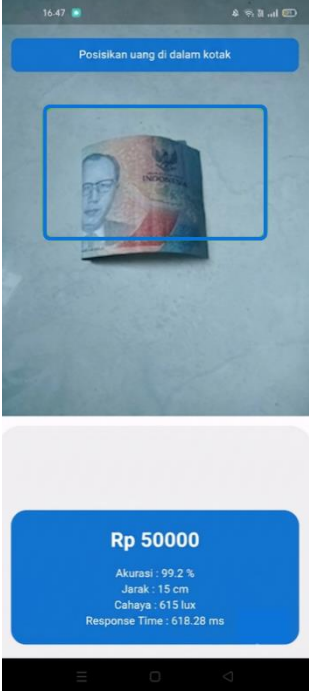
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	30 cm	Rp 20.000		99.0%	Berhasil
	30 cm	Rp 50.000		99.2%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	30 cm	Rp 100.000		99.0%	Berhasil
> 40 cm	Rp 1.000	Tidak Terdeteksi			Gagal
	Rp 2.000				
	Rp 5.000				
	Rp 10.000				
	Rp 20.000				
	Rp 50.000				
	Rp 100.000				

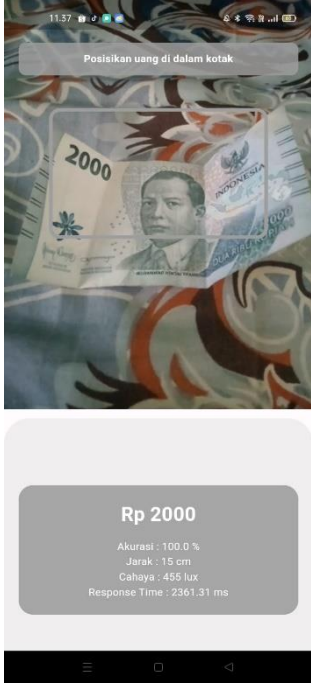
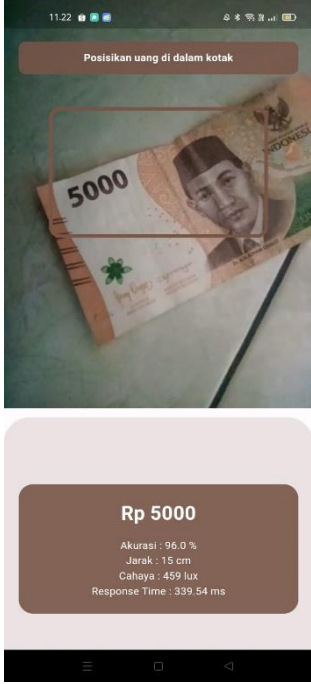
3. Pengujian Sistem Berdasarkan Cahaya

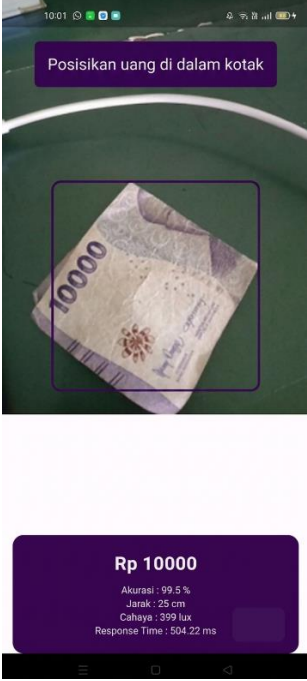
Cahaya		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
> 500 lux	612 lux	Rp 1.000		99.5%	Berhasil
	656 lux	Rp 2.000		98.1%	Berhasil

	540 lux	Rp 5.000		99.9%	Berhasil
	590 lux	Rp 10.000		99.5%	Berhasil

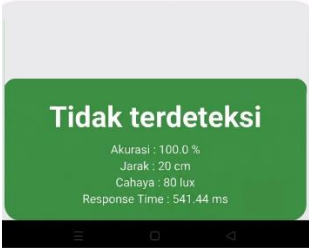
	513 lux	Rp 20.000		99.0%	Berhasil
	615 lux	Rp 50.000		99.2%	Berhasil

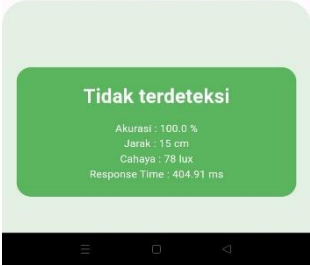
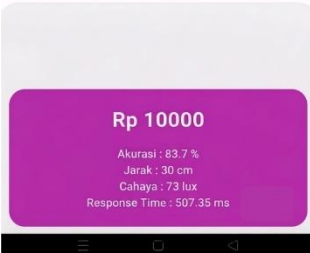
	551 lux	Rp 100.000	 Posisikan uang di dalam kotak Rp 100000 Akurasi : 99.1% Jarak : 17 cm Cahaya : 551 lux Response Time : 617.92 ms	99.1%	Berhasil
100 – 500 lux	442 lux	Rp 1.000	 Posisikan uang di dalam kotak Rp 1000 Akurasi : 100.0 % Jarak : 12 cm Cahaya : 442 lux Response Time : 535.78 ms	100.0%	Berhasil

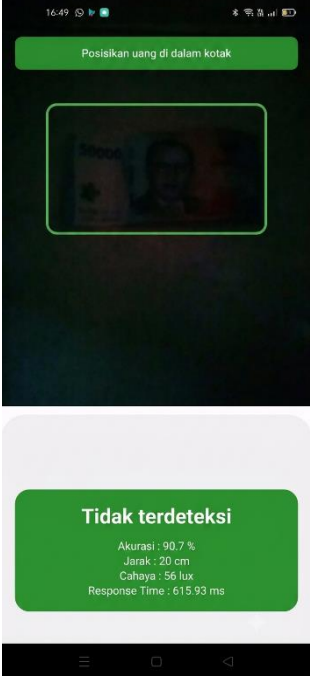
	455 lux	Rp 2.000		100.0%	Berhasil
	459 lux	Rp 5.000		96.0%	Berhasil

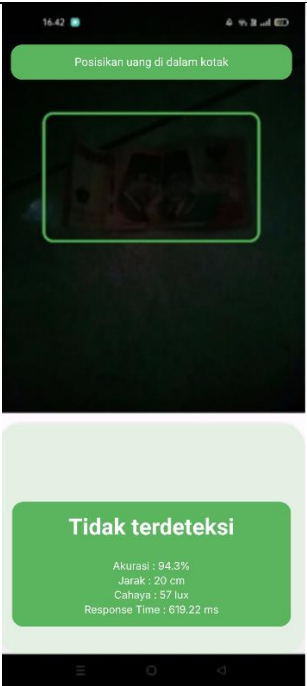
	399 lux	Rp 10.000		99.5%	Berhasil
	405 lux	Rp 20.000		98.8%	Berhasil

	365 lux	Rp 50.000		99.0%	Berhasil
	401 lux	Rp 100.000		99.7%	Berhasil

10 — 100 lux	80 lux	Rp 1.000	 	100.0%	Gagal
	79 lux	Rp 2.000	 	100.0%	Gagal

	78 lux	Rp 5.000	 	100.0%	Gagal
	73 lux	Rp 10.000	 	83.7%	Berhasil

	75 lux	Rp 20.000		84.0%	Berhasil
	56 lux	Rp 50.000		90.7%	Gagal

				94.3%	Gagal
	57 lux	Rp 100.000			
	< 10 lux	Rp 1.000	Tidak Terdeteksi		Gagal
		Rp 2.000			
		Rp 5.000			
		Rp 10.000			
		Rp 20.000			
		Rp 50.000			
		Rp 100.000			

4. Kode program

a. main.dart

```
import 'package:flutter/material.dart';
import 'splash.dart';

Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      debugShowCheckedModeBanner: false,
      home: SplashPage(),
    );
  }
}
```

b. splash.dart

```
import 'dart:async';
import 'package:flutter/material.dart';
import 'mulai.dart';

class SplashPage extends StatefulWidget {
  const SplashPage({super.key});

  @override
  State<SplashPage> createState() => _SplashPageState();
}

class _SplashPageState extends State<SplashPage> {
  @override
  void initState() {
    super.initState();

    // Splash tampil 5 detik
    Timer(const Duration(seconds: 3), () {
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(
          builder: (context) => const MulaiPage(),
        ),
      );
    });
  }

  @override
  Widget build(BuildContext context) {
    final size = MediaQuery.of(context).size;

    return Scaffold(
      body: Container(
        width: size.width,
        height: size.height,
        decoration: const BoxDecoration(
          gradient: LinearGradient(
            begin: Alignment.topCenter,
            end: Alignment.bottomCenter,
            colors: [
              Colors.white,
              Color(0xFF75C565),
            ],
          ),
        ),
      child: Center(
        child: Image.asset(
          'assets/img/money.png',
          width: size.width * 0.35,
          fit: BoxFit.contain,
        ),
      ),
    );
  }
}
```

```

        ),
      ),
    );
  }
}

```

c. mulai.dart

```

import 'package:flutter/material.dart';
import 'package:speech_to_text/speech_to_text.dart' as stt;
import 'dashboard.dart';

class MulaiPage extends StatefulWidget {
  const MulaiPage({super.key});

  @override
  State<MulaiPage> createState() => _MulaiPageState();
}

class _MulaiPageState extends State<MulaiPage> {
  late stt.SpeechToText _speech;
  bool _isListening = false;

  @override
  void initState() {
    super.initState();
    _speech = stt.SpeechToText();
    _initSpeech();
  }

  Future<void> _initSpeech() async {
    bool available = await _speech.initialize();
    if (available && mounted) {
      _startListening();
    }
  }

  void _startListening() {
    if (!_isListening) {
      setState(() => _isListening = true);

      _speech.listen(
        localeId: "id_ID",
        onResult: (result) async {
          String command =
result.recognizedWords.toLowerCase();

          if (command.contains("mulai") ||
              command.contains("masuk") ||
              command.contains("start")) {

```

```

        await _speech.stop();
        await _speech.cancel();

        Future.delayed(const Duration(milliseconds: 300),
() {
            if (mounted) {
                _goToDashboard();
            }
        });
    },
);
}

void _goToDashboard() {
    Navigator.pushReplacement(
        context,
        MaterialPageRoute(
            builder: (context) => const DashboardPage(),
        ),
    );
}

@override
void dispose() {
    _speech.stop();
    _speech.cancel();
    super.dispose();
}

@override
Widget build(BuildContext context) {
    final size = MediaQuery.of(context).size;

    return Scaffold(
        body: Container(
            width: size.width,
            height: size.height,
            decoration: const BoxDecoration(
                gradient: LinearGradient(
                    begin: Alignment.topCenter,
                    end: Alignment.bottomCenter,
                    colors: [
                        Colors.white,
                        Color(0xFF75C565),
                    ],
                ),
            ),
        child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [

```

```

        Image.asset(
          'assets/img/money.png',
          width: size.width * 0.35,
        ),

        SizedBox(height: size.height * 0.08),

        SizedBox(
          width: size.width * 0.7,
          height: 50,
          child: ElevatedButton(
            onPressed: () async {
              await _speech.stop();
              await _speech.cancel();
              _goToDashboard();
            },
            style: ElevatedButton.styleFrom(
              backgroundColor: Colors.white,
              shape: RoundedRectangleBorder(
                borderRadius: BorderRadius.circular(35),
              ),
            ),
            child: const Text(
              'START',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
                color: Color(0xFF75C565),
              ),
            ),
          ),
        ),
      ),
    ),
  ),
  const SizedBox(height: 30),

  Text(
    _isListening
      ? "Mendengarkan... Ucapkan 'Mulai'"
      : "Mengaktifkan suara...",
    style: const TextStyle(fontSize: 16),
  ),
),
],
),
),
);
}
}

```

d. dashboard.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:flutter_tts/flutter_tts.dart';
import 'package:speech_to_text/speech_to_text.dart' as stt;

import 'hasil.dart';
import 'panduan.dart';
import 'pengaturan.dart';

class DashboardPage extends StatefulWidget {
  const DashboardPage({super.key});

  @override
  State<DashboardPage> createState() => _DashboardPageState();
}

class _DashboardPageState extends State<DashboardPage> {
  final FlutterTts tts = FlutterTts();
  late stt.SpeechToText _speech;

  bool isListening = false;
  bool isSpeaking = false;
  bool isActive = false;

  String statusText = "Menunggu perintah...";
  String lastSpoken = "";

  @override
  void initState() {
    super.initState();
    initAll();
  }

  Future<void> initAll() async {
    _speech = stt.SpeechToText();
    await _speech.initialize();

    await tts.setLanguage("id-ID");
    await tts.setSpeechRate(0.5);
    await tts.awaitSpeakCompletion(true);

    await speak("Aplikasi siap. Ucapkan halo aplikasi untuk
mulai");

    startListening();
  }

  Future<void> speak(String text) async {
    if (text == lastSpoken) return; // 🔥 anti spam
    lastSpoken = text;

    setState(() {
      isSpeaking = true;
      statusText = text;
    });
  }
}
```

```

    });

    await tts.stop();
    await tts.speak(text);

    setState(() => isSpeaking = false);
  }

  void startListening() {
    if (isListening) return;

    setState(() {
      isListening = true;
      statusText = "Mendengarkan...";
    });

    _speech.listen(
      localeId: "id_ID",
      listenMode: stt.ListenMode.dictation,
      listenFor: const Duration(seconds: 30),
      pauseFor: const Duration(seconds: 5),
      cancelOnError: true,
      onResult: (result) async {
        if (!result.finalResult) return;

        String command = result.recognizedWords.toLowerCase();
        await processCommand(command);
      },
    );
  }

  Future<void> processCommand(String command) async {
    await _speech.stop();
    setState(() => isListening = false);

    HapticFeedback.mediumImpact();

    // WAKE WORD
    if (!isActive) {
      if (command.contains("halo")) {
        isActive = true;
        await speak(
          "Silakan pilih menu. Scan uang, panduan, atau pengaturan");
      } else {
        startListening();
      }
      return;
    }

    if (command.contains("scan") || command.contains("kamera"))
    {
      await navigate("Membuka kamera", const HasilPage());
    }
  }

```

```

    } else if (command.contains("panduan")) {
        await navigate("Membuka panduan", const PanduanPage());
    } else if (command.contains("pengaturan")) {
        await navigate("Membuka pengaturan", const
PengaturanPage());
    } else if (command.contains("kembali")) {
        if (Navigator.canPop(context)) {
            Navigator.pop(context);
            await speak("Kembali");
        } else {
            await speak("Sudah di dashboard");
        }
    } else {
        await speak("Perintah tidak dikenali");
    }

    startListening();
}

Future<void> navigate(String message, Widget page) async {
    await speak(message);

    if (!mounted) return;

    await Navigator.push(
        context,
        MaterialPageRoute(builder: (_) => page),
    );

    await speak("Kembali ke dashboard");
    startListening();
}

@override
void dispose() {
    _speech.stop();
    tts.stop();
    super.dispose();
}

// 🔥 BUTTON FULL AKSESIBILITAS
Widget buildButton(String text, VoidCallback onTap) {
    return Padding(
        padding: const EdgeInsets.symmetric(horizontal: 20,
vertical: 10),
        child: GestureDetector(
            onTapDown: (_) async {
                // 🗣️ 1x sentuh → baca
                await tts.stop();
                await tts.speak(text);
            },
            onDoubleTap: () async {

```

```

        // 📢 2x → eksekusi
        HapticFeedback.heavyImpact();

        await tts.stop();
        await tts.speak("Membuka $text");

        await Future.delayed(const Duration(milliseconds:
300));

        onTap();
      },
      child: Container(
        height: 70,
        decoration: BoxDecoration(
          color: Colors.white,
          borderRadius: BorderRadius.circular(25),
          border: Border.all(color: const Color(0xFF4CAF50),
width: 2),
          boxShadow: const [
            BoxShadow(
              blurRadius: 5,
              color: Colors.black12,
              offset: Offset(0, 3),
            )
          ],
        ),
      child: Center(
        child: Text(
          text,
          style: const TextStyle(
            fontSize: 20,
            fontWeight: FontWeight.bold,
            color: Color(0xFF4CAF50),
          ),
        ),
      ),
    ),
  ),
);
}

@override
Widget build(BuildContext context) {
  final size = MediaQuery.of(context).size;

  return Scaffold(
    body: Container(
      width: size.width,
      height: size.height,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          colors: [Colors.white, Color(0xFF4CAF50)],

```

```

        begin: Alignment.topCenter,
        end: Alignment.bottomCenter,
    ),
),
child: SafeArea(
  child: Column(
    children: [
      const SizedBox(height: 20),

      Container(
        padding: const EdgeInsets.all(20),
        margin: const EdgeInsets.symmetric(horizontal:
20),

        decoration: BoxDecoration(
          color: const Color(0xFF4CAF50),
          borderRadius: BorderRadius.circular(30),
        ),
        child: const Center(
          child: Text(
            "Dashboard Aksesibilitas",
            style: TextStyle(
              fontSize: 22,
              color: Colors.white,
              fontWeight: FontWeight.bold,
            ),
          ),
        ),
      ),

      const SizedBox(height: 30),

      Image.asset(
        'assets/img/uang.png',
        width: size.width * 0.6,
      ),

      const SizedBox(height: 30),

      buildButton("Scan Uang", () {
        navigate("Membuka kamera", const HasilPage());
      }),
      buildButton("Panduan", () {
        navigate("Membuka panduan", const
PanduanPage());
      }),
      buildButton("Pengaturan", () {
        navigate("Membuka pengaturan", const
PengaturanPage());
      }),

      const Spacer(),

```

```

        Text(
          isSpeaking
            ? "🔊 $statusText"
            : isListening
              ? "🔊 Mendengarkan..."
              : statusText,
          style: const TextStyle(fontSize: 16),
        ),
        const SizedBox(height: 20),
      ],
    ),
  ),
);
}
}

```

e. hasil.dart

```

import 'package:flutter/material.dart';
import 'package:camera/camera.dart';
import 'package:tflite_v2/tflite_v2.dart';
import 'package:flutter_tts/flutter_tts.dart';

class HasilPage extends StatefulWidget {
  const HasilPage({Key? key}) : super(key: key);

  @override
  State<HasilPage> createState() => _HasilPageState();
}

class _HasilPageState extends State<HasilPage> {
  CameraController? cameraController;
  FlutterTts tts = FlutterTts();

  bool detecting = false;

  String nominal = "Scanning...";
  String akurasi = "-";
  String jarak = "-";
  String cahaya = "-";

  String lastSpoken = "";

  Color beezColor = Colors.green;

  @override
  void initState() {
    super.initState();
    initAll();
  }
}

```

```

    }

    Color getMoneyColor(String nominal) {
        switch (nominal) {
            case "1000":
                return const Color(0xFF4CAF50);

            case "2000":
                return const Color(0xFF9E9E9E);

            case "5000":
                return const Color(0xFF795548);

            case "10000":
                return const Color(0xFF9C27B0);

            case "20000":
                return const Color(0xFF2E7D32);

            case "50000":
                return const Color(0xFF1976D2);

            case "100000":
                return const Color(0xFFD32F2F);

            case "bukan_uang":
                return Colors.grey;

            default:
                return Colors.teal;
        }
    }

    String getNominal2022(String label) {
        if (label.contains("_2022")) {
            return label.replaceAll("_2022", "");
        }
        return "";
    }

    Future initAll() async {
        final cameras = await availableCameras();

        cameraController = CameraController(
            cameras[0],
            ResolutionPreset.medium,
            enableAudio: false,
        );

        await cameraController!.initialize();

        await cameraController!.setFocusMode(

```

```

        FocusMode.auto,
    );

    await Tflite.loadModel(
      model: "assets/model/model_uang.tflite",
      labels: "assets/model/labels.txt",
    );

    await tts.setLanguage("id-ID");
    await tts.setSpeechRate(0.5);

    startStream();

    setState(() {});
  }

  void startStream() {
    cameraController!.startImageStream((image) {
      if (!detecting) {
        detecting = true;
        runModel(image);
      }
    });
  }

  Future runModel(CameraImage image) async {
    try {
      var result = await Tflite.runModelOnFrame(
        bytesList: image.planes.map((e) => e.bytes).toList(),
        imageHeight: image.height,
        imageWidth: image.width,
        imageMean: 127.5,
        imageStd: 127.5,
        rotation: 90,
        numResults: 3,
        threshold: 0.95,
      );

      double brightness = image.planes[0].bytes.reduce((a, b)
=> a + b) /
        image.planes[0].bytes.length;

      double lux = brightness * 4;

      if (result != null && result.isNotEmpty) {
        result.sort(
          (a, b) =>
            (b['confidence'] as
double).compareTo(a['confidence'] as double),
        );

        var res = result[0];

```

```

print(result);

String label = res["label"].toString().trim();

double conf = (res["confidence"] as double) * 100;

print("LABEL   : $label");
print("CONF    : $conf");

if (conf < 95) {
  setState(() {
    nominal = "Tidak terdeteksi";
    akurasi = "${conf.toStringAsFixed(1)} %";
    jarak = "-";
    cahaya = "${lux.toStringAsFixed(0)}";
    beezColor = Colors.green;
  });

  detecting = false;
  return;
}

double jarakCm = 15;

if (conf >= 95) {
  jarakCm = 15;
} else if (conf >= 90) {
  jarakCm = 20;
} else if (conf >= 85) {
  jarakCm = 25;
} else if (conf >= 80) {
  jarakCm = 30;
} else {
  jarakCm = 40;
}

if (label == "bukan_uang") {
  setState(() {
    nominal = "Tidak terdeteksi";
    akurasi = "${conf.toStringAsFixed(1)} %";
    jarak = "${jarakCm.toStringAsFixed(0)} cm";
    cahaya = "${lux.toStringAsFixed(0)} lux";
    beezColor = Colors.green;
  });

  if (lastSpoken != "bukan_uang") {
    lastSpoken = "bukan_uang";

    await tts.stop();

    await tts.speak("Bukan uang");
  }
}

```

```

    } else {
        setState(() {
            nominal = "Rp $label";
            akurasi = "${conf.toStringAsFixed(1)} %";
            jarak = "${jarakCm.toStringAsFixed(0)} cm";
            cahaya = "${lux.toStringAsFixed(0)} lux";
            beezColor = getMoneyColor(label);
        });

        if (lastSpoken != "label") {
            lastSpoken = "label";

            await tts.stop();

            String spokenText = label.replaceAll("_2022", "");

            await tts.speak("Uang ${spokenText} rupiah");

        }
    }
} catch (e) {
    print("ERROR : $e");
}

await Future.delayed(
    const Duration(milliseconds: 30),
);

detecting = false;
}

@override
void dispose() {
    cameraController?.dispose();
    Tflite.close();
    tts.stop();
    super.dispose();
}

@override
Widget build(BuildContext context) {
    if (cameraController == null ||
!cameraController!.value.isInitialized) {
        return const Scaffold(
            body: Center(
                child: CircularProgressIndicator(),
            ),
        );
    }

    return Scaffold(
        body: Stack(

```

```

children: [
  CameraPreview(cameraController!),
  Positioned(
    top: 150,
    left: 0,
    right: 0,
    child: Center(
      child: AnimatedContainer(
        duration: const Duration(milliseconds: 400),
        width: 300,
        height: 180,
        decoration: BoxDecoration(
          border: Border.all(
            color: beezColor,
            width: 5,
          ),
          borderRadius: BorderRadius.circular(15),
        ),
      ),
    ),
  ),
  Positioned(
    top: 60,
    left: 20,
    right: 20,
    child: AnimatedContainer(
      duration: const Duration(milliseconds: 400),
      padding: const EdgeInsets.all(12),
      decoration: BoxDecoration(
        color: beezColor.withOpacity(0.85),
        borderRadius: BorderRadius.circular(10),
      ),
      child: const Text(
        "Posisikan uang di dalam kotak",
        textAlign: TextAlign.center,
        style: TextStyle(
          color: Colors.white,
          fontSize: 16,
          fontWeight: FontWeight.bold,
        ),
      ),
    ),
  ),
  Align(
    alignment: Alignment.bottomCenter,
    child: AnimatedContainer(
      duration: const Duration(milliseconds: 400),
      height: 320,
      width: double.infinity,
      decoration: BoxDecoration(
        color: beezColor.withOpacity(0.15),
        borderRadius: const BorderRadius.only(
          topLeft: Radius.circular(35),

```

```

        topRight: Radius.circular(35),
      ),
    ),
  ),
),
Positioned(
  bottom: 50,
  left: 20,
  right: 20,
  child: AnimatedContainer(
    duration: const Duration(milliseconds: 400),
    padding: const EdgeInsets.all(20),
    decoration: BoxDecoration(
      color: beezColor.withOpacity(0.9),
      borderRadius: BorderRadius.circular(20),
    ),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        Text(
          nominal,
          style: const TextStyle(
            color: Colors.white,
            fontSize: 30,
            fontWeight: FontWeight.bold,
          ),
        ),
        const SizedBox(height: 15),
        Text(
          "Akurasi : $akurasi",
          style: const TextStyle(
            color: Colors.white,
            fontSize: 16,
          ),
        ),
        Text(
          "Jarak : $jarak",
          style: const TextStyle(
            color: Colors.white,
            fontSize: 16,
          ),
        ),
        Text(
          "Cahaya : $cahaya",
          style: const TextStyle(
            color: Colors.white,
            fontSize: 16,
          ),
        ),
      ],
    ),
  ),
),
),

```

```

        ),
      ],
    ),
  );
}
}

```

f. panduan.dart

```

import 'package:flutter/material.dart';
import 'package:flutter_tts/flutter_tts.dart';

class PanduanPage extends StatefulWidget {
  const PanduanPage({super.key});

  @override
  State<PanduanPage> createState() => _PanduanPageState();
}

class _PanduanPageState extends State<PanduanPage> {
  final FlutterTts flutterTts = FlutterTts();

  final String teksPanduan =
    "Selamat datang di Aplikasi Klasifikasi Uang Rupiah. "
    "Aplikasi ini membantu Anda mengenali nominal uang kertas "
    "Rupiah menggunakan kamera dan suara. "
    "Cara Menggunakan. "
    "Pertama, tekan tombol Scan Uang pada halaman utama untuk "
    "memulai. "
    "Kedua, arahkan uang ke kamera hingga terdengar suara. "
    "Ketiga, dengarkan hasil. Aplikasi akan menyebutkan "
    "nominal uang secara otomatis. "
    "Keempat, tekan tombol Scan Lagi untuk memindai uang "
    "berikutnya.";

  @override
  void initState() {
    super.initState();
    speakPanduan();
  }

  Future<void> speakPanduan() async {
    await flutterTts.setLanguage("id-ID");
    await flutterTts.setSpeechRate(0.5);
    await flutterTts.setVolume(1.0);
    await flutterTts.speak(teksPanduan);
  }

  @override
  void dispose() {
    flutterTts.stop();
    super.dispose();
  }
}

```

```

}

@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Container(
      width: double.infinity,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [
            Colors.white,
            Color(0xFF75C565),
          ],
        ),
      ),
    child: SafeArea(
      child: Column(
        children: [
          Align(
            alignment: Alignment.centerLeft,
            child: IconButton(
              icon: const Icon(Icons.arrow_back),
              onPressed: () {
                flutterTts.stop();
                Navigator.pop(context);
              },
            ),
          ),
          const SizedBox(height: 10),
          Expanded(
            child: Container(
              margin: const
EdgeInsets.symmetric(horizontal: 20),
              padding: const EdgeInsets.all(20),
              decoration: BoxDecoration(
                color: Colors.white,
                borderRadius: BorderRadius.circular(30),
                border: Border.all(
                  color: const Color(0xFF75C565),
                  width: 3,
                ),
              ),
            child: SingleChildScrollView(
              child: Column(
                children: [
                  const Text(
                    "Panduan",
                    style: TextStyle(
                      fontSize: 26,
                      fontWeight: FontWeight.bold,
                      color: Color(0xFF75C565),

```

```

        ),
      ),
      const SizedBox(height: 20),
      Text(
        teksPanduan,
        style: const TextStyle(
          fontSize: 18,
          height: 1.6,
          color: Color(0xFF75C565),
        ),
      ),
    ),
    const SizedBox(height: 30),

    // Tombol ulangi suara
    ElevatedButton(
      onPressed: speakPanduan,
      child: const Text("Putar Ulang
Suara"),
    ),
  ],
),
),
),
),
const SizedBox(height: 20),
],
),
),
),
);
}
}

```

g. pengaturan.dart

```

import 'package:flutter/material.dart';
import 'package:flutter_tts/flutter_tts.dart';
import 'informasiaplikasi.dart';
import 'tentangkami.dart';

class PengaturanPage extends StatefulWidget {
  const PengaturanPage({super.key});

  @override
  State<PengaturanPage> createState() =>
    _PengaturanPageState();
}

class _PengaturanPageState extends State<PengaturanPage> {
  final FlutterTts flutterTts = FlutterTts();

  @override

```

```

void initState() {
  super.initState();
  speak("Halaman pengaturan. Silakan pilih menu.");
}

Future<void> speak(String text) async {
  await flutterTts.setLanguage("id-ID");
  await flutterTts.setSpeechRate(0.5);
  await flutterTts.speak(text);
}

@override
void dispose() {
  flutterTts.stop();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  final size = MediaQuery.of(context).size;

  return Scaffold(
    body: Container(
      width: size.width,
      height: size.height,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [Colors.white, Color(0xFF75C565)],
        ),
      ),
    child: SafeArea(
      child: Padding(
        padding: const EdgeInsets.symmetric(horizontal:
20),
        child: Column(
          children: [
            const SizedBox(height: 20),

            Container(
              width: double.infinity,
              padding: const EdgeInsets.symmetric(vertical:
20),
              decoration: BoxDecoration(
                color: const Color(0xFF75C565),
                borderRadius: BorderRadius.circular(30),
              ),
              child: const Center(
                child: Text(
                  "Pengaturan",
                  style: TextStyle(
                    fontSize: 26,

```



```

        decoration: BoxDecoration(
          color: Colors.white,
          borderRadius: BorderRadius.circular(30),
          border: Border.all(color: const Color(0xFF75C565),
width: 3),
          boxShadow: [
            BoxShadow(
              color: Colors.black.withOpacity(0.2),
              blurRadius: 5,
              offset: const Offset(0, 4),
            ),
          ],
        ),
        child: Center(
          child: Text(
            text,
            style: const TextStyle(
              fontSize: 22,
              fontWeight: FontWeight.bold,
              color: Color(0xFF75C565),
            ),
          ),
        ),
      ),
    ),
  );
}
}

```

h. informasiaplikasi.dart

```

import 'package:flutter/material.dart';
import 'package:flutter_tts/flutter_tts.dart';

class InformasiAplikasiPage extends StatefulWidget {
  const InformasiAplikasiPage({super.key});

  @override
  State<InformasiAplikasiPage> createState() =>
    _InformasiAplikasiPageState();
}

class _InformasiAplikasiPageState
  extends State<InformasiAplikasiPage> {

  final FlutterTts flutterTts = FlutterTts();

  @override
  void initState() {
    super.initState();
    speak();
  }
}

```

```

Future<void> speak() async {
  await flutterTts.setLanguage("id-ID");
  await flutterTts.setSpeechRate(0.5);
  await flutterTts.speak(
    "Halaman informasi aplikasi. "
    "Aplikasi ini menggunakan metode Convolutional Neural
Network "
    "untuk mendeteksi nominal uang kertas Rupiah dan
memberikan output suara."
  );
}

@override
void dispose() {
  flutterTts.stop();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  final size = MediaQuery.of(context).size;

  return Scaffold(
    body: Container(
      width: size.width,
      height: size.height,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [
            Colors.white,
            Color(0xFF75C565),
          ],
        ),
      ),
    child: SafeArea(
      child: Padding(
        padding: const EdgeInsets.symmetric(horizontal:
20),

        child: Column(
          children: [
            const SizedBox(height: 20),

            // HEADER
            Container(
              width: double.infinity,
              padding: const EdgeInsets.symmetric(vertical:
20),

              decoration: BoxDecoration(
                color: const Color(0xFF75C565),
                borderRadius: BorderRadius.circular(30),

```

```

    ),
    child: const Center(
      child: Text(
        "Informasi Aplikasi",
        style: TextStyle(
          fontSize: 26,
          fontWeight: FontWeight.bold,
          color: Colors.white,
        ),
      ),
    ),
  ),
),

const SizedBox(height: 30),

// HALAMAN PUTIH (CARD)
Expanded(
  child: Container(
    width: double.infinity,
    padding: const EdgeInsets.all(25),
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius: BorderRadius.circular(25),
      boxShadow: [
        BoxShadow(
          color: Colors.black26,
          blurRadius: 10,
          offset: Offset(0, 5),
        ),
      ],
    ),
  ),
  child: const SingleChildScrollView(
    child: Text(
      "Aplikasi ini menggunakan metode  

      Convolutional Neural Network (CNN) "  

      "untuk mendeteksi nominal uang kertas  

      Rupiah melalui kamera smartphone. "  

      "Sistem akan memproses citra  

      menggunakan model pembelajaran mesin "  

      "dan menghasilkan output berupa suara  

      untuk membantu pengguna tunanetra "  

      "mengetahui nilai uang secara  

      mandiri.\n\n"  

      "Aplikasi ini dirancang dengan  

      antarmuka sederhana dan ramah aksesibilitas.",
      style: TextStyle(
        fontSize: 18,
        height: 1.6,
      ),
      textAlign: TextAlign.justify,
    ),
  ),
),
),

```

```

        ),
      ],
    ),
  ),
),
);
}
}

```

i. tentangkami.dart

```

import 'package:flutter/material.dart';
import 'package:flutter_tts/flutter_tts.dart';

class TentangKamiPage extends StatefulWidget {
  const TentangKamiPage({super.key});

  @override
  State<TentangKamiPage> createState() =>
    _TentangKamiPageState();
}

class _TentangKamiPageState extends State<TentangKamiPage> {

  final FlutterTts flutterTts = FlutterTts();

  @override
  void initState() {
    super.initState();
    speak();
  }

  Future<void> speak() async {
    await flutterTts.setLanguage("id-ID");
    await flutterTts.setSpeechRate(0.5);
    await flutterTts.speak(
      "Halaman tentang kami. "
      "Aplikasi ini dikembangkan sebagai proyek skripsi "
      "untuk membantu penyandang tunanetra mengenali nominal "
      "uang secara mandiri."
    );
  }

  @override
  void dispose() {
    flutterTts.stop();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {

```

```

final size = MediaQuery.of(context).size;

return Scaffold(
  body: Container(
    width: size.width,
    height: size.height,
    decoration: const BoxDecoration(
      gradient: LinearGradient(
        begin: Alignment.topCenter,
        end: Alignment.bottomCenter,
        colors: [
          Colors.white,
          Color(0xFF75C565),
        ],
      ),
    ),
  child: SafeArea(
    child: Padding(
      padding: const EdgeInsets.symmetric(horizontal:
20),

      child: Column(
        children: [
          const SizedBox(height: 20),

          // Header
          Container(
            width: double.infinity,
            padding: const EdgeInsets.symmetric(vertical:
20),

            decoration: BoxDecoration(
              color: const Color(0xFF75C565),
              borderRadius: BorderRadius.circular(30),
            ),
            child: const Center(
              child: Text(
                "Tentang Kami",
                style: TextStyle(
                  fontSize: 26,
                  fontWeight: FontWeight.bold,
                  color: Colors.white,
                ),
              ),
            ),
          ),

          const SizedBox(height: 30),

          // Halaman Putih seperti Panduan
          Expanded(
            child: Container(
              width: double.infinity,
              padding: const EdgeInsets.all(25),
              decoration: BoxDecoration(

```

