

BAB I

PENDAHULUAN

1.1. Latar Belakang

Bagi sebagian besar individu, kemampuan untuk mengenali objek dan memahami lingkungan melalui penglihatan merupakan aspek fundamental yang menunjang efektivitas aktivitas sehari-hari [1]. Namun, bagi individu tunanetra, keterbatasan fungsi penglihatan menuntut indra lain, seperti indra peraba dan indra pendengar untuk mengambil alih peran sebagai kanal informasi primer [2]. Spektrum tunanetra itu sendiri mencakup kondisi yang beragam, mulai dari *low vision*, di mana sisa penglihatan masih dapat difungsikan, hingga kebutaan total (*total blind*), di mana fungsi visual tidak lagi dapat diandalkan sama sekali [3].

Skala permasalahan ini di Indonesia cukup signifikan. Data dari Kementerian Kesehatan mengindikasikan bahwa prevalensi tunanetra mencapai 1,5% dari total populasi [4], sementara itu, studi lain dari Australia-Indonesia Disability Research and Advocacy Network (AIDRAN) memberikan estimasi yang lebih tinggi, yakni sekitar 4 juta jiwa [5]. Angka-angka ini menggarisbawahi urgensi pengembangan solusi teknologi yang tidak hanya inklusif dan adaptif, tetapi juga dirancang secara presisi untuk menjawab tantangan spesifik yang dihadapi oleh komunitas tunanetra. Dalam navigasi dan interaksi sehari-hari, penyandang tunanetra mengandalkan berbagai strategi dan alat bantu, mulai dari tongkat hingga kaca mata pintar. Meskipun krusial untuk menunjang mobilitas, efektivitas alat bantu konvensional ini sering kali terbatas saat berhadapan dengan lingkungan yang dinamis dan kompleks [6]. Keterbatasan ini melahirkan berbagai hambatan fungsional yang

berdampak langsung pada penurunan kualitas hidup [7]. Aktivitas yang dianggap sederhana bagi individu awas, seperti mengenali objek di sekitar, membaca teks pada kemasan produk, mengidentifikasi nominal uang kertas saat bertransaksi, atau menemukan barang pribadi dapat menjadi tantangan besar yang mengurangi tingkat kemandirian tanpa adanya dukungan teknologi asistif yang memadai [8].

Hambatan fungsional tersebut kemudian sering kali meluas menjadi tantangan psikososial yang lebih dalam. Ketergantungan pada orang lain dapat membatasi partisipasi sosial dan memicu dinamika yang tidak seimbang, seperti perlakuan protektif berlebihan atau munculnya stereotip negatif terhadap kapabilitas individu tunanetra [9]. Fenomena ini teramati bahkan pada usia dini, di mana anak-anak tunanetra dilaporkan mengalami kesulitan dalam membangun interaksi dengan teman sebaya non-disabilitas, yang berimplikasi pada rendahnya keterlibatan sosial mereka [10]. Akumulasi dari hambatan fungsional dan sosial ini berpotensi menimbulkan perasaan terisolasi dan keterasingan dalam kehidupan bermasyarakat. Oleh karena itu, pengembangan teknologi yang dapat mengatasi hambatan fungsional utama tidak hanya bertujuan untuk meningkatkan kemandirian, tetapi juga secara tidak langsung berpotensi memitigasi dampak sosial negatif tersebut dengan memberdayakan individu untuk berpartisipasi lebih penuh dalam masyarakat.

Di tengah tantangan tersebut, kemajuan pesat dalam teknologi digital, khususnya dalam satu dekade terakhir, telah membuka paradigma baru dalam pengembangan solusi asistif. Perkembangan pesat di bidang kecerdasan buatan (*Artificial Intelligence*), *Internet of Things* (IoT), serta perangkat lunak berbasis

mobile atau *website* telah menghadirkan peluang untuk menciptakan solusi yang lebih efektif, adaptif, dan fungsional. Di antara berbagai platform teknologi, *smartphone* muncul sebagai perangkat yang paling strategis dan potensial [11]. Sifatnya yang portabel, daya komputasinya yang terus meningkat, serta tingkat adopsinya yang tinggi di kalangan komunitas tunanetra menjadikannya landasan teknologi yang ideal bagi pengembangan inovasi asistif [12].

Potensi *smartphone* ini diperkuat oleh fakta bahwa perangkat tersebut telah terintegrasi secara mendalam dalam kehidupan pengguna tunanetra melalui fitur aksesibilitas bawaan. Teknologi pembaca layar (*screen reader*), seperti TalkBack pada Android dan VoiceOver pada iOS, memungkinkan pengguna untuk bernavigasi menggunakan umpan balik suara (*auditory feedback*) dan gestur sentuh yang intuitif [13][14]. Efektivitas adopsi ini divalidasi oleh studi, salah satunya di Universitas Airlangga, yang menunjukkan bagaimana mahasiswa tunanetra secara mahir memanfaatkan *smartphone* untuk berbagai kebutuhan akademik dan sosial, mulai dari mengakses literatur digital hingga berkomunikasi secara daring [15].

Seiring dengan evolusi *smartphone* menjadi perangkat komputasi personal yang ada di mana-mana, potensinya sebagai prostesis sensorik untuk penyandang tunanetra semakin nyata [12]. Meski telah dilengkapi perangkat keras mumpuni seperti kamera, prosesor, dan aktuator umpan balik haptik, pemanfaatan maksimalnya masih terhambat oleh keterbatasan perangkat lunak. Fitur aksesibilitas bawaan, sekalipun esensial, umumnya hanya berfokus pada antarmuka digital, bukan dunia fisik di sekitar pengguna. Kesenjangan inilah yang menjadi justifikasi fundamental bagi pengembangan aplikasi asistif berbasis *computer*

vision, yang bertujuan mentransformasi perangkat yang sudah ada menjadi mata artifisial yang fungsional.

Kemajuan pesat dalam *computer vision*, khususnya pada algoritma *deep learning*, mendorong terciptanya aplikasi asistif generasi baru. Di antara berbagai pendekatan, arsitektur *You Only Look Once* (YOLO) tampil sebagai salah satu yang paling potensial untuk deteksi objek *real-time* karena efisiensi dan kecepatannya [16]. Evolusi arsitekturnya, dari versi awal hingga versi yang lebih modern seperti YOLOv5, YOLOv7, dan YOLOv8, terus menunjukkan peningkatan performa, menjadikannya pilihan yang divalidasi secara luas untuk aplikasi asistif [17]. Relevansinya telah ditegaskan melalui berbagai studi yang mengonfirmasi keseimbangan optimal antara akurasi dan kecepatan YOLO untuk navigasi dalam ruangan (*indoor*) [18], serta kemampuannya dalam mendeteksi rintangan di luar ruangan (*outdoor*) dengan presisi tinggi, yang krusial untuk mobilitas yang aman dan mandiri [19][20].

Namun, penerapan model canggih seperti YOLO pada perangkat *mobile* yang memiliki keterbatasan sumber daya (CPU, RAM, baterai) merupakan sebuah tantangan rekayasa yang signifikan [21][22]. Untuk mengatasi kendala ini, teknik optimisasi model seperti kuantisasi menjadi sangat krusial. Proses kuantisasi, khususnya ke format INT8, mampu mengurangi ukuran model secara drastis sekaligus mempercepat waktu inferensi pada CPU *mobile* dengan degradasi akurasi yang minimal. Kesesuaian pendekatan ini dengan riset terkini pada arsitektur TFLite mengukuhkan kuantisasi INT8 sebagai langkah rekayasa yang strategis [23].

Aplikasi asistif modern dibangun di atas serangkaian teknologi pendukung yang fundamental. Salah satu yang paling esensial adalah *Optical Character Recognition* (OCR) yang dipadukan dengan *Text-to-Speech* (TTS), sebuah kombinasi yang memungkinkan pengguna untuk mengubah teks cetak dari berbagai sumber menjadi informasi audio yang dapat diakses [24]. Meskipun demikian, analisis terhadap solusi komersial terkemuka seperti Seeing AI dan Envision AI menunjukkan bahwa integrasi berbagai fitur ini sering kali belum optimal. Walaupun kaya fitur, performanya kerap tidak konsisten di berbagai tugas [25]. Keterbatasan utama yang dilaporkan pengguna mencakup latensi akibat ketergantungan pada pemrosesan awan (*cloud*), isu privasi data, dan konsumsi daya baterai yang tinggi [26]. Di sisi lain, riset akademis cenderung menghasilkan prototipe yang unggul pada satu fungsi spesifik namun gagal terintegrasi dalam sebuah ekosistem yang komprehensif [25]. Dari sini, teridentifikasi sebuah kesenjangan riset berupa kebutuhan akan sebuah solusi asistif dengan integrasi fitur-fitur pendukung tunanetra yang tetap mampu beroperasi tanpa konektivitas internet.

Sebagai respons terhadap kesenjangan riset yang telah teridentifikasi, penelitian ini berfokus pada pengembangan See4Me, sebuah aplikasi asistif berbasis Android yang bertujuan untuk mengurangi beberapa tantangan fungsional yang kerap dihadapi oleh individu tunanetra. Prinsip kerja aplikasi ini adalah mengonversi data visual esensial, meliputi pengenalan objek, teks, dan mata uang ke dalam format umpan balik audio yang informatif. Untuk mengoptimalkan responsivitas, seluruh proses komputasi dieksekusi secara lokal pada perangkat

pengguna (*on-device*) sebagai sebuah pendekatan yang meminimalkan latensi pemrosesan. Dengan demikian, See4Me dikembangkan sebagai alat bantu asistif terintegrasi, yang diproyeksikan dapat berkontribusi pada peningkatan kemandirian pengguna saat berinteraksi dengan lingkungan sekitar mereka.

1.2. Pembatasan Masalah

Untuk memastikan penelitian ini tetap fokus, terarah, dan dapat diselesaikan dalam lingkup yang realistis, ditetapkan batasan-batasan sebagai berikut:

1. Pengembangan perangkat lunak difokuskan khusus untuk sistem operasi Android. Desain antarmuka dan fungsionalitas difokuskan untuk mencapai kompatibilitas fungsional dengan *screen reader* bawaan Android (TalkBack). Kompatibilitas dengan *screen reader* atau platform aksesibilitas lainnya tidak menjadi ruang lingkup penelitian.
2. Penelitian menggunakan model *pre-trained* `yolo11n_int8.tflite` yang disediakan oleh Ultralytics. Tidak dilakukan proses pelatihan ulang (*fine-tuning*) atau pengembangan model baru.
3. Deteksi objek terbatas pada 80 kelas standar yang tersedia pada dataset COCO, sesuai cakupan data pelatihan model. Objek di luar kategori tersebut tidak dapat dikenali.
4. Fokus penelitian adalah pada pengembangan dan integrasi empat fitur utama, yaitu Cari Benda, Deskripsi Cerdas, Deteksi Nominal Uang, dan Scan Teks. Fitur asistif lain seperti navigasi berbasis GPS, maupun pengukuran jarak antara kamera dengan objek tidak termasuk dalam ruang lingkup penelitian ini.

5. Fitur Deteksi Nominal Uang hanya mendukung pengenalan uang kertas Rupiah (IDR). Tidak dirancang untuk mengenali uang logam atau mata uang negara lain.
6. Fitur Scan Teks hanya menerima inputan berupa gambar dari kamera atau galeri. Deteksi bahasa terbatas pada Bahasa Indonesia dan Bahasa Inggris berdasarkan identifikasi otomatis. Bahasa bacaan ditentukan secara otomatis melalui deteksi bahasa, pengguna tidak dapat memilih bahasa secara manual. Fungsi seperti terjemahan, ringkasan, atau analisis lanjutan tidak termasuk ruang lingkup penelitian ini.

1.3. Tujuan dan Manfaat

1.3.1 Tujuan

Penelitian ini bertujuan untuk:

1. Merancang dan membangun aplikasi asistif berbasis Android dengan pemrosesan *on-device* untuk memastikan operasionalitas luring dan mengoptimalkan responsivitas aplikasi.
2. Mengimplementasikan deteksi objek *real-time* menggunakan model *pre-trained* `yolo11n_int8.tflite` yang dilengkapi analisis warna berbasis CIE L^*a^*b untuk mendeskripsikan lingkungan visual.
3. Mengintegrasikan teknologi *Optical Character Recognition* (OCR) dan *Text-to-Speech* (TTS) sebagai basis untuk dua fungsionalitas utama berupa pengenalan nominal mata uang Rupiah dengan validasi heuristik, dan konversi teks dari gambar.

1.3.2 Manfaat

Pengembangan aplikasi ini diharapkan memberikan manfaat sebagai berikut:

1. Menyediakan sistem panduan non-visual berbasis audio dan getaran untuk membantu pengguna dalam melokalisasi objek yang dicari, serta menawarkan fungsi konversi teks ke suara untuk mengakses informasi dari media cetak.
2. Menyajikan sebuah mekanisme validasi nominal uang Rupiah berbasis OCR dan logika heuristik, yang dirancang untuk mengurangi potensi kesalahan identifikasi selama transaksi.
3. Menyajikan informasi deskriptif mengenai lingkungan sekitar secara *real-time*, yang mencakup identifikasi objek, estimasi posisi relatif, dan atribut warna.

1.4. Tinjauan Pustaka

Perkembangan dalam bidang kecerdasan buatan (*Artificial Intelligence*), khususnya pada ranah *computer vision*, menawarkan potensi untuk pengembangan solusi teknologi yang mendukung aktivitas penyandang tunanetra. Salah satu pendekatan adalah melalui integrasi teknologi pendukung seperti *Optical Character Recognition* (OCR) dan *Text-to-Speech* (TTS) yang berfungsi untuk mentransformasikan informasi visual menjadi format audio yang aksesibel [27]. Sejumlah penelitian terdahulu telah mengeksplorasi penerapan teknologi ini, namun sering kali dengan fokus yang spesifik dan batasan tertentu.

Penelitian pertama mengembangkan sebuah aplikasi pembaca buku bagi tunanetra dan *low vision*. Dengan memanfaatkan teknologi *Optical Character Recognition* (OCR) untuk mengubah teks dari media cetak menjadi format digital yang kemudian dibacakan melalui *Text-to-Speech* (TTS), aplikasi ini menjawab

kebutuhan esensial terkait aksesibilitas literatur. Keunggulan utama dari pendekatan ini adalah fokusnya yang tajam pada satu kebutuhan spesifik dan kemudahan penggunaannya yang didukung oleh antarmuka sederhana serta panduan suara. Namun, sifatnya sebagai aplikasi fungsi tunggal (*single-function*) menjadi keterbatasan utama. Pengguna yang memerlukan bantuan untuk tugas lain, seperti mengenali objek di sekitarnya atau mengidentifikasi nominal uang, harus beralih ke aplikasi lain. Hal ini menciptakan pengalaman pengguna yang terfragmentasi dan kurang efisien untuk mendukung berbagai aktivitas harian secara holistik [28].

Studi lain mengembangkan aplikasi Recoptic, sebuah platform asistif yang dirancang khusus untuk membantu tunanetra dalam mengidentifikasi obat-obatan. Aplikasi ini mengintegrasikan beberapa teknologi seperti OCR dari ML Kit dan model deteksi objek berbasis TensorFlow Lite untuk mengenali teks pada kemasan obat serta objek-objek umum lainnya. Nilai kebaruan utamanya terletak pada spesialisasi pada domain medis, yang memberikan informasi krusial seperti indikasi dan dosis obat, sebuah fitur yang tidak umum ditemukan pada aplikasi asistif lainnya. Meskipun memiliki fitur deteksi objek dan teks, fokus utamanya yang sangat terspesialisasi pada identifikasi obat membuat fungsionalitas lainnya bersifat sekunder. Keterbatasan ini menunjukkan adanya celah untuk sebuah aplikasi yang dirancang dengan rangkaian fitur yang lebih seimbang dan berorientasi pada kebutuhan umum sehari-hari, seperti menemukan barang personal, memahami lingkungan secara umum, dan melakukan transaksi keuangan [29].

Studi berikutnya mengembangkan aplikasi deteksi objek *real-time* di Android yang beroperasi sepenuhnya secara *on-device*. Dengan memanfaatkan model YOLOv4-Tiny yang dioptimalkan untuk TensorFlow Lite, aplikasi ini mampu mendeteksi berbagai objek dari dataset COCO dan memberikan umpan balik melalui suara dan getaran tanpa memerlukan koneksi internet. Keunggulan signifikan dari penelitian ini adalah validasi konsep bahwa deteksi objek yang kompleks dapat dijalankan secara efisien di perangkat *mobile* dengan latensi rendah, yang menjawab masalah ketergantungan pada *cloud*. Akan tetapi, aplikasi ini masih terbatas pada satu fungsionalitas, yaitu deteksi objek. Belum ada integrasi dengan fitur-fitur esensial lain seperti pemindaian teks (OCR) atau pengenalan mata uang Rupiah, sehingga potensi platform *on-device* tersebut belum dimanfaatkan secara maksimal untuk menciptakan solusi asistif yang komprehensif [30].

Penelitian berbasis perangkat lunak mengembangkan *Assistive Android Application for Visually Impaired Individuals*. Aplikasi ini mengintegrasikan berbagai fitur seperti deteksi objek, konversi gambar ke teks dengan OCR, laporan cuaca, serta deteksi lokasi, yang memberikan informasi kepada pengguna melalui umpan balik audio secara *real-time*. Kelebihan pendekatan ini adalah berbasis *software only*, sehingga dapat dijalankan di *smartphone* tanpa perangkat tambahan. Namun demikian, potensi ketergantungan pada API eksternal untuk fitur seperti cuaca dapat memengaruhi ketersediaan layanan, dan detail performa teknis model deteksi objek yang digunakan tidak dijelaskan secara mendalam [31].

Selain itu, penelitian berbasis aplikasi *mobile* dengan fungsi tunggal yang dikembangkan menggunakan model VGG-19 berfokus pada mendeteksi keaslian

dan nominal uang Rupiah, khususnya pecahan Rp 50.000 dan Rp 100.000, dengan akurasi sebesar 88%. Aplikasi ini cukup relevan dengan konteks lokal dan membantu pengguna tunanetra saat transaksi keuangan. Namun, solusi ini dirancang sebagai aplikasi fungsi tunggal, sehingga pengguna tetap harus berganti aplikasi lain untuk kebutuhan berbeda, seperti membaca dokumen atau mendeteksi objek di sekitarnya [32].

Tabel 1.1 Gap Penelitian

No	Penelitian Terkait	Gap Penelitian	
		Penelitian Terdahulu	Penelitian Sekarang
1.	Aplikasi Baca Buku untuk Tunanetra <i>Low Vision</i> Menggunakan an <i>Optical</i> <i>Character</i> <i>Recognition</i> dan <i>Text-to-</i> <i>Speech</i> . (2023)	<p>1. Teknologi: OCR, <i>Text-to-Speech</i> (TTS), Android.</p> <p>2. Fitur: Fungsi tunggal untuk pembacaan teks.</p> <p>3. Detail Perbandingan: Menyediakan solusi untuk satu kebutuhan spesifik, namun menyebabkan pengalaman pengguna terfragmentasi.</p>	<p>1. Teknologi: YOLOv11n dan Tensorflow Lite, OCR (ML Kit), TTS, Android.</p> <p>2. Fitur: Empat fitur terintegrasi (Cari Benda, Deskripsi Cerdas, Deteksi Uang, Scan Teks).</p> <p>3. Detail Perbandingan: Menyediakan platform <i>all-in-one</i> yang holistik, mengatasi masalah fragmentasi aplikasi.</p>

No	Penelitian Terkait	Gap Penelitian	
		Penelitian Terdahulu	Penelitian Sekarang
2.	<i>Recoptic Application Development with Artificial Intelligence on Drug Detection Feature for Visually Impaired People.</i> (2023)	<p>1. Teknologi: TensorFlow Lite & ML Kit pada platform Android.</p> <p>2. Fitur: Multifungsi (deteksi obat, objek dan teks).</p> <p>3. Detail Perbandingan: Fungsionalitas dirancang secara spesifik untuk domain medis (identifikasi obat).</p>	<p>1. Teknologi: YOLOv11n dan Tensorflow Lite, OCR (ML Kit), TTS, Android.</p> <p>2. Fitur: Multifungsi (deteksi objek, deskripsi lingkungan, deteksi uang dan teks).</p> <p>3. Detail Perbandingan: Fungsionalitas dirancang untuk kebutuhan umum sehari-hari yang lebih luas (mencari barang, transaksi, dll.).</p>
3.	<i>Real-Time Object Detection App for Android with Voice</i>	<p>1. Teknologi: YOLOv4-Tiny dan Tensorflow Lite, Android.</p> <p>2. Fitur: Fungsi tunggal untuk deteksi objek.</p>	<p>1. Teknologi: YOLOv11n dan Tensorflow Lite, OCR (ML Kit), TTS, Android.</p>

No	Penelitian Terkait	Gap Penelitian	
		Penelitian Terdahulu	Penelitian Sekarang
	<i>Feedback.</i> (2025)	<p>3. Detail Perbandingan:</p> <p>Arsitektur <i>on-device</i> yang terbatas pada satu fungsi (deteksi objek) dan menggunakan model generasi sebelumnya.</p>	<p>2. Fitur:</p> <p>Mengintegrasikan deteksi objek dengan tiga fitur lain.</p> <p>3. Detail Perbandingan:</p> <p>Arsitektur <i>on-device</i> yang mengintegrasikan beberapa fungsi dan menggunakan model generasi terbaru.</p>
4.	<i>Assistive Android Application for Visually Impaired Individuals.</i> (2024)	<p>1. Teknologi:</p> <p>Java, ML, OCR, & API Eksternal, Android.</p> <p>2. Fitur:</p> <p>Multifungsi (objek, teks, cuaca, lokasi).</p> <p>3. Detail Perbandingan:</p> <p>Arsitektur <i>hybrid</i> yang memerlukan koneksi internet untuk beberapa fitur utama, sehingga</p>	<p>1. Teknologi:</p> <p>YOLOv11n dan Tensorflow Lite, OCR (ML Kit), TTS, Android.</p> <p>2. Fitur:</p> <p>Empat modul inti (Cari Benda, Deskripsi Cerdas, Deteksi Uang, Scan Teks).</p> <p>3. Detail Perbandingan:</p>

No	Penelitian Terkait	Gap Penelitian	
		Penelitian Terdahulu	Penelitian Sekarang
		membatasi penggunaan di area tanpa sinyal.	Arsitektur sepenuhnya <i>on-device</i> , memastikan fungsionalitas penuh tanpa koneksi internet.
5.	<i>Indonesia Rupiah Currency Detection for Visually Impaired Using Transfer Learning VGG-19. (2025)</i>	<p>1. Teknologi: Model klasifikasi gambar (VGG-19), Android.</p> <p>2. Fitur: Deteksi mata uang Rupiah.</p> <p>3. Detail Perbandingan: Cakupan fungsional terbatas, hanya mendukung 2 dari 7 nominal uang kertas, sehingga kurang praktis untuk semua skenario transaksi.</p>	<p>1. Teknologi: YOLOv11n dan Tensorflow Lite, OCR (ML Kit), TTS, Android.</p> <p>2. Fitur: Deteksi nominal mata uang kertas Rupiah.</p> <p>3. Detail Perbandingan: Cakupan fungsional komprehensif, mendukung semua 7 nominal uang kertas, sehingga lebih praktis untuk penggunaan sehari-hari.</p>

Tinjauan terhadap penelitian terdahulu menunjukkan bahwa teknologi *computer vision* telah diterapkan pada platform *mobile* untuk berbagai fungsi

asistif, seperti deteksi objek *real-time*, pengenalan teks (OCR), dan identifikasi mata uang. Meskipun demikian, ditemukan batasan dari pendekatan yang ada, bahwa sebagian besar solusi dirancang sebagai aplikasi fungsi tunggal yang membatasi utilitasnya dalam skenario harian yang beragam, dan beberapa pendekatan justru bergantung pada jaringan internet.

Menjawab batasan-batasan tersebut, penelitian ini berfokus mengusulkan pengembangan aplikasi *mobile* Android yang mengintegrasikan empat fitur, Cari Benda, Deskripsi Cerdas, Deteksi Nominal Uang, dan Scan Teks. Aplikasi ini dirancang untuk beroperasi sepenuhnya secara *on-device*, memungkinkan berfungsi tanpa koneksi internet dan dengan demikian meminimalkan potensi keterlambatan respons (*latency*). Pendekatan terintegrasi dalam satu aplikasi ini diharapkan dapat menawarkan sebuah solusi yang lebih efisien untuk membantu pengguna tunanetra dalam menjalankan aktivitas tanpa perlu beralih antar aplikasi yang berbeda.

1.5. Data Penelitian

Penelitian ini menggunakan beberapa jenis data untuk mendukung pengembangan dan fungsionalitas dari setiap fitur pada aplikasi See4Me. Data tersebut mencakup dataset model *pre-trained* untuk deteksi objek, serta data input yang diproses secara *real-time* untuk pengenalan mata uang, teks, dan warna.

1.5.1 Dataset Objek (COCO Dataset)

Untuk fitur Cari Benda dan Deskripsi Cerdas, penelitian ini memanfaatkan model deteksi objek *pre-trained* `yolo11n_int8.tflite`. Model ini telah dilatih sebelumnya oleh Ultralytics menggunakan dataset COCO (*Common Objects in*

Context), yang merupakan dataset berskala besar dan menjadi standar dalam penelitian *computer vision*.

Dataset COCO mencakup lebih dari 200.000 gambar yang telah dilabeli dengan 80 kategori objek umum, seperti orang, kendaraan, perabotan rumah tangga, dan hewan. Penggunaan model yang dilatih pada dataset ini memberikan aplikasi kemampuan dasar untuk mengenali beragam objek di lingkungan sekitar pengguna, yang berkontribusi pada fungsi pemahaman konteks lingkungan. Beberapa contoh kelas objek yang didukung dapat dilihat pada Tabel 1.2.

Tabel 1.2 Contoh Kelas Objek dari Dataset COCO

Kategori	Contoh Kelas Objek
Perabotan dan Interior Rumah	kursi, sofa, tanaman pot, tempat tidur, meja makan, toilet, televisi, kulkas, buku, jam, vas
Peralatan Dapur dan Makanan	botol, cangkir, garpu, pisau, sendok, mangkuk, pisang, apel, roti lapis, jeruk, pizza, donat
Elektronik dan Aksesoris	laptop, mouse, remote, keyboard, ponsel, microwave, oven, ransel, tas tangan, koper
Hewan	burung, kucing, anjing, kuda, domba, sapi, gajah, beruang, zebra, jerapah
Olahraga dan Luar Ruang	bangku, payung, frisbi, ski, bola, layang-layang, tongkat bisbol, papan seluncur, raket tenis

1.5.2 Data Input Uang Kertas Rupiah

Berbeda dengan deteksi objek, fitur Deteksi Uang tidak menggunakan dataset gambar yang dilatih secara khusus. Sebaliknya, fitur ini bekerja dengan memproses data gambar yang ditangkap secara langsung dari kamera pengguna. Metode yang digunakan adalah *pipeline* pemrosesan dua tahap yang menggabungkan *Optical Character Recognition* (OCR) dengan logika heuristik.

Alur kerjanya adalah sebagai berikut:

1. Pengambilan Gambar, sistem menangkap gambar uang kertas dari kamera perangkat.
2. Ekstraksi Teks dengan OCR, gambar yang ditangkap kemudian diproses menggunakan *engine* OCR dari Google ML Kit (*TextRecognizer*) untuk mengekstraksi semua teks yang terlihat pada uang kertas.
3. Analisis Heuristik, teks hasil OCR dianalisis menggunakan logika *parsing heuristik*. Sistem akan mencari string numerik yang sesuai dengan nominal yang ada (misalnya “100000”, “50000”) dan memvalidasinya dengan memeriksa keberadaan kata kunci kontekstual seperti “rupiah” atau “bank Indonesia” pada teks yang sama.

Strategi ini memungkinkan pengenalan nominal tanpa memerlukan pelatihan model klasifikasi mata uang yang terpisah.

1.5.3 Data Input Tekstual dan Warna

Aplikasi ini juga memproses dua jenis data input lainnya secara *real-time*:

1. Data Teks (untuk Fitur Scan Teks)

Fitur ini memproses data berupa gambar yang ditangkap melalui kamera atau dipilih dari galeri perangkat. Teknologi OCR dari Google ML Kit digunakan untuk mengekstraksi teks dari gambar tersebut. Untuk mendukung fungsionalitas multibahasa, sistem juga mengintegrasikan LanguageIdentifier untuk secara otomatis mengenali bahasa teks (Indonesia atau Inggris) dan menyesuaikan lafal *Text-to-Speech* (TTS) agar sesuai dengan bahasa yang terdeteksi.

2. Data Warna (untuk Fitur Deskripsi Cerdas)

Fitur ini mengambil data warna dari area objek yang terdeteksi pada gambar. Untuk memberikan deskripsi warna, data piksel dari gambar (yang umumnya dalam format RGB) dikonversi ke ruang warna CIE $L^*a^*b^*$. Ruang warna ini memisahkan komponen kecerahan/luminans (L^*) dari dua komponen kromatisitas (a^* untuk spektrum hijau-merah dan b^* untuk spektrum biru-kuning). Pemisahan ini membuat identifikasi komponen warna (a^* dan b^*) menjadi lebih konsisten terhadap variasi intensitas cahaya, karena tidak terikat langsung dengan tingkat kecerahan gambar.

1.5.4 Alat Penelitian

Pengembangan dan pengujian aplikasi See4Me didukung oleh serangkaian perangkat keras dan perangkat lunak.

Tabel 1.3 Alat Penelitian

Jenis Alat	Nama Alat	Fungsi Utama
<i>Hardware</i>	Laptop (HP Envy x360, AMD Ryzen 7 5800U, RAM 16GB)	Komponen utama untuk pengembangan, kompilasi kode, dan manajemen proyek
	<i>Smartphone</i> Android (Minimal OS 8.0 Oreo, RAM 4GB)	Perangkat target untuk instalasi, pengujian, dan <i>debugging</i> aplikasi secara <i>real-time</i>
	Kabel USB	Menghubungkan <i>smartphone</i> ke laptop untuk proses <i>debugging</i> dan transfer data
<i>Software</i>	Windows 11 (64-bit)	Sistem operasi yang digunakan pada komputer pengembang
	Visual Studio Code / Android Studio	<i>Integrated Development Environment</i> (IDE) untuk pengembangan aplikasi
	Flutter SDK (v3.32.4) & Dart SDK (v3.8.1)	Kerangka kerja dan bahasa pemrograman utama yang

Jenis Alat	Nama Alat	Fungsi Utama
		digunakan untuk membangun aplikasi
	TensorFlow Lite	Kerangka kerja <i>machine learning</i> untuk menjalankan model deteksi objek secara <i>on-device</i>
	Google ML Kit	Menyediakan model OCR dan <i>Language Identification</i> untuk pemrosesan teks
	Git	Sistem kontrol versi untuk manajemen kode