RANCANG BANGUN SISTEM PARKIR DENGAN FINGERPRINT MENGGUNAKAN RASPBERRY PI

DESIGN AND CONSTRUCTION OF A PARKING SYSTEM WITH FINGERPRINT USING RASPBERRY PI

Edwin Chandra[1)], Mariska Putri Pratiwi[2)]
[1]Program Studi Teknik Informatika, Fakultas Teknik dan Komputer, Universitas Putra Batam
alamat, Kota Batam, Kepulauan Riau
Telp : (29461) 081363851313
E-mail : pb220210036@upbatam.ac.id[1)] mrskaptrw@gmail.com[2)]

Abstract

This research presents the design and implementation of a Smart Parking System that integrates biometric authentication, license plate recognition, and dynamic tariff computation to enhance security and operational efficiency in urban parking environments. The system is developed using modular Python architecture and deployed on a Raspberry Pi platform, incorporating hardware components such as fingerprint sensors, cameras, infrared detectors, and servo motors. A graphical user interface (GUI), built with Tkinter, provides real-time updates on hardware status, captured images, and fee calculations based on a tiered pricing model (Rp 4000 base fee plus Rp 2000 per hour, capped at Rp 30,000). Dual authentication at entry and exit points ensures secure access control, while automated gate operation and ultrasonic clearance detection streamline traffic flow. The system supports structured CSV logging with weekly data rollovers to facilitate traceability and long-term monitoring. Experimental results demonstrate the system's reliability, scalability, and maintainability, indicating its potential for deployment in smart city infrastructure and automated parking management.

Kata Kunci: Raspberry PI, IOT, Smart Parking, License Plate Recognition, Fingerprint

Abstrak

Penelitian ini menyajikan perancangan dan implementasi Sistem Parkir Pintar yang mengintegrasikan autentikasi biometrik, pengenalan plat nomor kendaraan, dan perhitungan tarif dinamis untuk meningkatkan keamanan serta efisiensi operasional di lingkungan perkotaan. Sistem ini dikembangkan menggunakan arsitektur Python modular dan dijalankan pada platform Raspberry Pi, dengan komponen perangkat keras seperti sensor sidik jari, kamera, detektor inframerah, dan motor servo. Antarmuka pengguna grafis (GUI) yang dibangun menggunakan Tkinter menyediakan pembaruan waktu nyata mengenai status perangkat keras, gambar yang ditangkap, serta perhitungan biaya berdasarkan model tarif bertingkat (Rp 4000 biaya dasar ditambah Rp 2000 per jam, dengan batas maksimum Rp 30.000). Autentikasi ganda pada titik masuk dan keluar memastikan kontrol akses yang aman, sementara operasi gerbang otomatis dan deteksi jarak ultrasonik memperlancar alur lalu lintas. Sistem ini mendukung pencatatan terstruktur dalam format CSV dengan rotasi data mingguan untuk memfasilitasi pelacakan dan pemantauan jangka panjang. Hasil eksperimen menunjukkan bahwa sistem ini memiliki keandalan, skalabilitas, dan kemudahan pemeliharaan yang tinggi, serta berpotensi untuk diterapkan dalam infrastruktur kota pintar dan manajemen parkir otomatis.

Kata Kunci: Raspberry PI, IOT, Smart Parking, License Plate Recognition, Fingerprint

1. Pendahuluan

Di era urbanisasi cepat, manajemen parkir menjadi krusial untuk efisiensi ruang, pengurangan kemacetan, dan pelayanan publik. Sistem parkir konvensional berbasis karcis kertas rentan terhadap antrean panjang, kesalahan pencatatan, pemalsuan, dan biaya operasional serta dampak lingkungan. Studi sebelumnya menunjukkan kebutuhan akan solusi digital yang mendukung pembayaran elektronik dan pemantauan real-time [1]. Pertumbuhan jumlah kendaraan di Kota Batam yang tinggi, ditambah sistem parkir konvensional berbasis karcis yang rawan pemalsuan, antrean panjang, dan pencatatan manual yang tidak akurat, menimbulkan masalah serius dalam efisiensi dan keamanan pengelolaan parkir. Selain itu, penggunaan kertas sebagai media karcis meningkatkan biaya operasional dan berdampak negatif terhadap lingkungan. Kondisi ini menegaskan urgensi perlunya sistem parkir pintar berbasis Internet of Things (IoT) yang mengintegrasikan

biometrik sidik jari dan pengenalan plat nomor kendaraan untuk menciptakan solusi yang lebih aman, efisien, dan ramah lingkungan.

Penelitian mengusulkan prototipe sistem parkir berbasis Raspberry Pi yang mengintegrasikan autentikasi biometrik sidik jari, deteksi kendaraan, pembacaan plat nomor (YOLO + OCR), dan kontrol gerbang otomatis. Pendekatan rekayasa sistem berbasis IoT dipilih untuk memungkinkan pengumpulan data real-time, otomatisasi alur masuk/keluar, serta pencatatan sesi dalam format CSV untuk traceability. Raspberry Pi dipilih karena fleksibilitasnya, kemampuan menjalankan aplikasi Python, dan kemudahan integrasi dengan sensor serta kamera.[2]

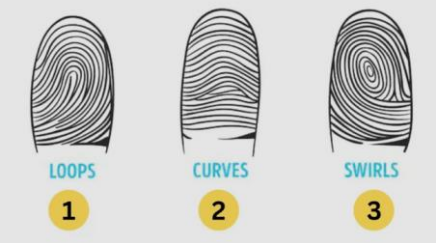
Kontribusi penelitian meliputi desain dan implementasi prototipe end-to-end yang menggabungkan biometrik dan ANPR, evaluasi performa di laboratorium dan lapangan, serta rekomendasi optimasi model dan operasi untuk deployment pada perangkat edge[3]. Diharapkan solusi ini mengurangi penggunaan kertas, mempercepat proses parkir, meningkatkan keamanan transaksi, dan mendukung adopsi sistem parkir digital di lingkungan perkotaan seperti Batam.

2. Tinjauan Pustaka

mengulas teori, penelitian terdahulu, dan teknologi inti yang menjadi dasar pengembangan sistem parkir pintar berbasis sidik jari dan Raspberry Pi. Fokus ulasan meliputi autentikasi biometrik (sidik jari), arsitektur embedded dan IoT untuk sistem parkir, deteksi objek dengan YOLO untuk ANPR, teknik OCR untuk pembacaan karakter plat, sensor dan aktuator yang dipakai (ultrasonic dan servo), serta praktik penyimpanan dan logging data ringan menggunakan CSV.

2.1 Autentikasi Biometrik Sidik Jari

Sidik jari merupakan metode identifikasi yang aman dan sulit dipalsukan. Penelitian menunjukkan bahwa penggunaan sensor R305 efektif untuk sistem keamanan dan parkir [7], [10]. Namun, penelitian tersebut masih terbatas pada autentikasi tunggal. Gap yang muncul adalah belum adanya integrasi **dual authentication** (fingerprint + ANPR) untuk meningkatkan keamanan sistem parkir.



2.2 Arsitektur Embedded dan IoT

Raspberry Pi banyak digunakan sebagai pusat kendali sistem IoT karena fleksibilitas, konsumsi daya rendah, dan dukungan Python [4], [5]. Kurniawan [6] menekankan keunggulan IoT dalam mengurangi kemacetan dan meningkatkan efisiensi parkir. Akan tetapi, penelitian sebelumnya lebih fokus pada **informasi slot parkir** tanpa menggabungkan autentikasi biometrik. Penelitian ini menutup gap dengan mengintegrasikan IoT, biometrik, dan ANPR dalam satu sistem end-to-end.

2.3 Deteksi Objek dengan YOLO untuk ANPR

Santanu & Jonatan [13] menggunakan YOLO dan EasyOCR untuk membaca plat nomor kendaraan dengan hasil cukup baik. Namun, penelitian tersebut hanya fokus pada ANPR tanpa mempertimbangkan keterbatasan perangkat **edge computing** seperti Raspberry Pi. Penelitian ini menambahkan kontribusi dengan **optimasi model YOLO (quantization, ONNX/TFLite)** agar dapat berjalan efisien di perangkat dengan sumber daya terbatas.

2.4 Optical Character Recognition (OCR)

OCR digunakan untuk membaca karakter plat setelah deteksi area dengan YOLO. Penelitian sebelumnya menekankan preprocessing (grayscale, thresholding, resize) untuk meningkatkan akurasi [14]. Gap yang masih ada adalah **akurasi rendah pada kondisi pencahayaan buruk dan plat kotor**, sehingga penelitian ini mengusulkan **multipass preprocessing dan fallback manual** untuk menjaga kontinuitas layanan.

2.5 Sensor dan Aktuator (Ultrasonik & Servo)

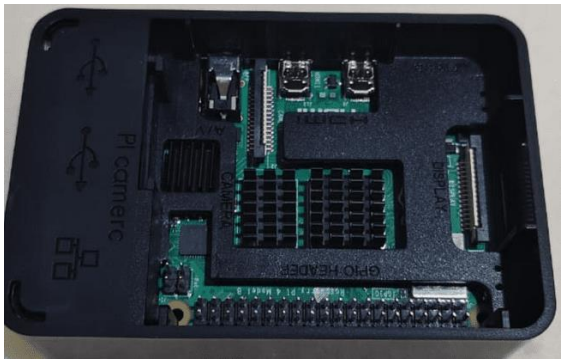
Sensor HC-SR04 digunakan untuk mendeteksi kendaraan, sementara servo MG90S mengontrol palang otomatis. Penelitian terdahulu menunjukkan efektivitas komponen ini [5], [11], tetapi belum banyak membahas **failsafe design** (misalnya lock mode atau manual override saat servo gagal). Penelitian ini menambahkan aspek tersebut agar sistem lebih andal di lapangan.

2.6 Penyimpanan dan Logging Data (CSV)

CSV dipilih karena sederhana dan portabel. Penelitian Irsan [15] menekankan keunggulan CSV untuk pencatatan ringan. Gap penelitian sebelumnya adalah **kurangnya strategi rotasi dan atomic write** untuk mencegah data corrupt saat beban tinggi. Penelitian ini menutup gap dengan **rotasi mingguan dan atomic write** untuk meningkatkan keandalan.

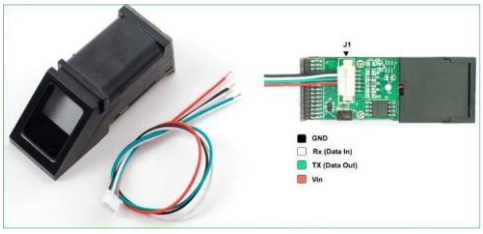
2.7 Raspberry PI

Raspberry Pi adalah komputer singleboard berukuran kecil yang banyak dipakai untuk prototipe sistem embedded dan aplikasi IoT karena biaya rendah, konsumsi daya hemat, dan dukungan GPIO untuk integrasi sensor/aktuator [7]. Pada sistem parkir, Raspberry Pi bertindak sebagai pusat kendali: mengumpulkan sinyal dari sensor ultrasonik [6] dan modul fingerprint, menangkap gambar kamera, menjalankan logika orkestrasi (entry/exit, perhitungan tarif), serta menyimpan log dalam format ringan seperti CSV. Kelebihan praktisnya meliputi kompatibilitas dengan Python dan pustaka computer vision (OpenCV), kemampuan menjalankan model inference terkonversi (ONNX/TFLite), dan opsi konektivitas (WiFi/USB) untuk backup atau monitoring. Keterbatasan utama adalah sumber daya terbatas untuk beban komputasi berat, sehingga perlu strategi pembagian tugas (worker threads), optimasi model untuk edge (quantization/pruning), atau pemindahan pelatihan ke cloud[8].



2.8 Fingerprint Module R305

Modul R305 digunakan untuk autentikasi biometrik dengan kemampuan menyimpan template lokal dan melakukan pencocokan cepat melalui antarmuka serial. Penelitian menunjukkan sensor ini cocok untuk aplikasi embedded dengan kebutuhan keamanan menengah hingga tinggi [7].



2.9 Sensor Ultrasonik HC-SR04

Sensor HC-SR04 digunakan untuk mendeteksi keberadaan kendaraan di gerbang masuk/keluar. Penelitian menunjukkan sensor ini memiliki akurasi tinggi dengan biaya rendah, meskipun perlu kalibrasi threshold untuk mengurangi false positive akibat pantulan [5].



2.10 Servo MG90S

Servo MG90S digunakan sebagai aktuator pembuka palang otomatis. Penelitian sebelumnya menekankan pentingnya kontrol PWM dan mekanisme failsafe agar servo tidak rusak akibat beban berlebih [11] .



2.11 Geany

Geany digunakan sebagai IDE untuk pengembangan perangkat lunak berbasis Python. Keunggulannya adalah ringan, mendukung berbagai bahasa pemrograman, serta kompatibel dengan Raspberry Pi, sehingga memudahkan proses coding dan debugging [10].

3. Metodologi Penelitian

Penelitian ini menggunakan pendekatan rekayasa sistem dengan metode prototyping iteratif. Tahapan dimulai dari studi literatur dan observasi lapangan, dilanjutkan dengan perancangan hardware dan software modular berbasis Raspberry Pi. Pengembangan dilakukan per modul (fingerprint, sensor, kamera, ANPR, logging), kemudian diintegrasikan secara endtoend. Model YOLO dilatih menggunakan dataset ANPR dan dioptimasi untuk perangkat edge. Pengujian dilakukan secara fungsional dan lapangan untuk mengevaluasi akurasi fingerprint, keberhasilan OCR, latency sistem, dan keandalan operasional. Hasil dicatat dalam format CSV dan dianalisis untuk perbaikan sistem.

3.1 Studi Literatur

Tahap studi literatur dilakukan untuk mengkaji teori dan penelitian terdahulu yang relevan dengan pengembangan sistem parkir pintar. Kajian ini mencakup teori parkir dan manajemen transportasi, konsep sistem berbasis Internet of Things, autentikasi biometrik sidik jari, serta arsitektur sistem embedded menggunakan Raspberry Pi. Selain itu, literatur juga membahas teknologi deteksi plat nomor kendaraan dengan algoritma YOLO dan pembacaan karakter menggunakan OCR, penggunaan sensor ultrasonik HC-SR04 untuk deteksi kendaraan, servo MG90S sebagai aktuator palang otomatis, serta praktik pencatatan data menggunakan format CSV. Studi literatur ini memberikan landasan teoritis sekaligus mengidentifikasi gap penelitian, yaitu belum adanya integrasi biometrik dengan ANPR dan belum optimalnya penerapan model deteksi pada perangkat dengan sumber daya terbatas.

3.2 Observasi Lapangan

Observasi dilakukan di lokasi parkir di Kota Batam yang memiliki mobilitas kendaraan tinggi. Hasil observasi menunjukkan bahwa sistem parkir konvensional masih menghadapi berbagai permasalahan, seperti antrean panjang pada pintu masuk dan keluar, pencatatan manual yang tidak akurat, potensi pemalsuan tiket parkir, serta dampak lingkungan akibat penggunaan karcis berbasis kertas. Temuan ini memperkuat urgensi penelitian dan menjadi dasar perancangan sistem parkir pintar yang lebih efisien, aman, dan ramah lingkungan

3.3 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk menentukan komponen utama yang diperlukan dalam sistem. Dari sisi perangkat keras, sistem menggunakan Raspberry Pi sebagai pusat kendali, sensor sidik jari R305 untuk autentikasi pengguna, kamera USB untuk pengambilan gambar plat nomor, sensor ultrasonik HC-SR04 untuk mendeteksi kendaraan, serta servo MG90S sebagai aktuator pembuka palang otomatis. Dari sisi perangkat lunak, sistem dikembangkan menggunakan bahasa pemrograman Python dengan arsitektur modular, antarmuka grafis berbasis Tkinter, pipeline YOLO dan OCR untuk ANPR, serta pencatatan data menggunakan format CSV.

3.4 Perancangan Sistem

Perancangan sistem dilakukan dengan memodelkan alur kerja menggunakan diagram UML yang meliputi use case, activity, sequence, dan class diagram. Perancangan perangkat keras dilakukan dengan mengintegrasikan sensor sidik jari, kamera, sensor ultrasonik, dan servo melalui GPIO Raspberry Pi dengan konfigurasi wiring yang efisien dan mudah dipelihara. Perancangan perangkat lunak dilakukan dengan membangun modul fingerprint, plate recognizer, sessions manager, event logger, serta antarmuka Tkinter untuk operator. Sistem dirancang agar modular sehingga setiap komponen dapat diuji secara terpisah sebelum diintegrasikan menjadi sistem end-to-end.

3.5 Perancangan Software

Perancangan perangkat lunak dilakukan dengan pendekatan modular menggunakan bahasa pemrograman Python. Setiap fungsi utama sistem dipisahkan ke dalam modul agar mudah diuji, dipelihara, dan dikembangkan. Modul fingerprint dirancang untuk melakukan autentikasi biometrik menggunakan sensor R305. Proses yang dilakukan meliputi pendaftaran sidik jari, pencocokan template, serta mekanisme retry apabila pembacaan gagal. Modul plate recognizer mengintegrasikan pipeline YOLO untuk mendeteksi area plat nomor dan OCR untuk membaca karakter. Hasil deteksi disimpan bersama tingkat kepercayaan agar dapat dianalisis lebih lanjut.

Modul sessions manager berfungsi mengelola data sesi parkir, termasuk waktu masuk, waktu keluar, identitas pengguna, hasil ANPR, durasi parkir, dan biaya. Data dicatat dalam format CSV dengan rotasi mingguan untuk menjaga keteraturan dan keandalan sistem. Modul event logger mencatat setiap aktivitas sistem, seperti hasil autentikasi, status palang, dan error yang terjadi. Pencatatan dilakukan dalam format CSV untuk memudahkan audit dan debugging.

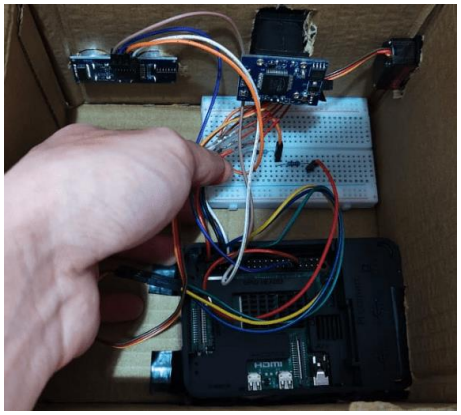
Selain itu, antarmuka grafis dibangun menggunakan Tkinter. Antarmuka ini menampilkan status perangkat, hasil pemindaian sidik jari, preview kamera, hasil deteksi plat, serta biaya parkir. Operator juga dapat melakukan kontrol manual seperti membuka palang jika terjadi kegagalan sistem.



Perancangan perangkat lunak ini memastikan bahwa setiap komponen dapat diuji secara terpisah sebelum diintegrasikan ke dalam sistem end-to-end. Pendekatan modular juga memudahkan pengembangan lebih lanjut, misalnya penambahan fitur pembayaran digital atau integrasi dengan server cloud.

3.6 Perancangan Hardware

Perancangan hardware sistem parkir pintar ini menggunakan Raspberry Pi sebagai pusat kendali yang terhubung dengan beberapa komponen utama: sensor sidik jari R305 untuk autentikasi pengguna, kamera USB untuk pengambilan gambar plat nomor, sensor ultrasonik HC-SR04 untuk mendeteksi keberadaan kendaraan, dan servo MG90S sebagai aktuator pembuka palang otomatis. Semua komponen dihubungkan melalui GPIO Raspberry Pi dengan konfigurasi wiring yang disesuaikan untuk efisiensi ruang dan kemudahan pemeliharaan. Sistem ini dirancang agar modular dan dapat diuji secara terpisah sebelum integrasi penuh.



3.7 Autentikasi biometrik sidik jari

Autentikasi biometrik sidik jari merupakan metode identifikasi yang andal karena pola alur dan minutiae pada ujung jari bersifat unik untuk setiap individu[3]. Sensor seperti modul R305 banyak digunakan pada aplikasi embedded karena kemampuannya menyimpan template lokal dan melakukan pencocokan secara mandiri melalui antarmuka serial[13]. Dalam literatur ditegaskan keunggulan sidik jari untuk akses kontrol pada sistem dengan kebutuhan keamanan menengah hingga tinggi: waktu pencocokan relatif singkat, integrasi mudah dengan mikrokontroler atau singleboard computer, serta toleransi error yang dapat ditangani dengan mekanisme retry dan fallback. Namun banyak kajian juga menyoroti keterbatasan praktis—mis. pembacaan gagap bila kondisi jari berminyak, basah, atau terluka—sehingga desain sistem perlu memasukkan strategi penanganan kondisi buruk dan opsi otentikasi cadangan.

3.8 Raspberry Pi dan arsitektur sistem embedded untuk aplikasi IoT

Raspberry Pi dipilih luas sebagai pusat kendali sistem embedded karena fleksibilitas, dukungan GPIO, kemampuan menjalankan Python, serta kompatibilitas dengan kamera dan modul komunikasi[5]. Dalam aplikasi IoT, Raspberry Pi berperan sebagai hub pengumpulan data sensor, pengolah logika kontrol, dan penginterface ke layanan cloud bila diperlukan. Studistudi penerapan menekankan desain modular pada level perangkat lunak (mis. pemisahan modul sensor, aktor, ANPR, dan logging) dan penggunaan service/daemon agar sistem tahan gangguan. Untuk deployment lapangan, aspek seperti manajemen daya, proteksi I/O, mekanisme watchdog, dan backup data lokal menjadi kriteria penting yang sering direkomendasikan.

3.9 Deteksi objek untuk ANPR: YOLO

YOLO (You Only Look Once) adalah keluarga model deteksi objek yang memproses gambar dalam satu langkah inferensi sehingga cocok untuk aplikasi realtime. Versi versi terbaru (mis. YOLOv5, YOLOv8) menawarkan varian backbone ringan sampai menengah yang dapat disesuaikan antara akurasi dan latency untuk perangkat[11]. Literatur teknik ANPR menggarisbawahi perlunya dataset yang representatif (sudut, jarak, kondisi pencahayaan) dan anotasi bounding box yang konsisten untuk melatih YOLO agar dapat mendeteksi area plat secara andal[9]. Praktik terbaik meliputi augmentasi konservatif, validasi split (train/val/test) yang reproducible, serta konversi/optimasi model (quantization, TFLite/ONNX) sebelum deployment pada Raspberry Pi untuk memenuhi batasan performa dan memori[10].

3.10 Optical Character Recognition (OCR) pada area plat nomor

Setelah YOLO mendeteksi dan mengekstrak crop area plat, tahap OCR membaca karakter pada crop tersebut. Pilihan populer pada penelitian dan prototipe adalah EasyOCR dan Tesseract. Teknik prapemrosesan (grayscale, adaptive thresholding, denoising, contrast enhancement, dan resize) memainkan peran kunci meningkatkan akurasi OCR—terutama pada kondisi pencahayaan buruk atau plat yang kotor[14].

Penelitian menunjukkan bahwa pipeline deteksi→crop→pra-pemrosesan→OCR harus diuji end-to-end; apabila confidence OCR rendah, sistem harus menerapkan retry dengan preprocessing berbeda atau menandai hasil sebagai unknown untuk input manual operator.

3.11 Sensor ultrasonik dan motor servo pada pengendalian gerbang

Sensor HCSR04 dan varian ultrasonic umum dipakai untuk mendeteksi kehadiran kendaraan di ambang gerbang karena biaya rendah dan implementasi sederhana pada GPIO Raspberry Pi. Teknik debounce (mengharuskan beberapa pembacaan konsisten dalam waktu) dan threshold kalibrasi diperlukan untuk mengurangi false positive/negative akibat pantulan/lingkungan. Untuk motor servo, servo MG90S atau serupa biasa dipakai pada mekanisme palang;

kontrol PWM harus disertai safety timeout dan pengawasan posisi untuk mencegah kerusakan mekanik atau kondisi tak terduga. Banyak studi menekankan desain failsafe: jika servo gagal, sistem harus masuk mode aman (lock atau manual override).

3.12 Manajemen data ringan menggunakan CSV dan logging audit

Dalam konteks prototipe dan deployment ringan, format CSV sering dipilih untuk menyimpan session log karena kesederhanaan, portabilitas, dan kemudahan analisis. Praktik terbaik meliputi skema kolom yang konsisten (id_session, entry_iso, exit_iso, user_id, plate, entry_image, exit_image, duration_min, fee, ocr_conf, notes), penulisan atomik (write tmp + rename), rotasi berkala (mis. mingguan), dan pencadangan otomatis bila tersedia media eksternal atau jaringan. Selain session log, event logger berformat CSV memberikan jejak audit untuk debugging dan analisis insiden[12].

4. Hasil dan Pembahasan

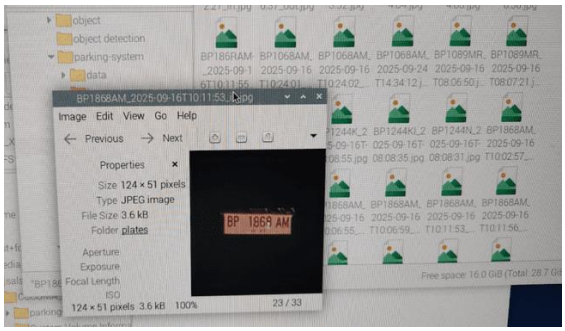
Prototipe sistem parkir berbasis Raspberry Pi berhasil dibangun dan diuji end-to-end. Komponen inti (sensor sidik jari R305, sensor ultrasonik HC-SR04, kamera USB, servo MG90S), modul perangkat lunak (fingerprint, ultrasonic, camera, plate_recognizer, sessions_manager, event_logger), antarmuka operator (Tkinter), serta pipeline ANPR (YOLO → preprocessing → OCR) telah diintegrasikan. Pengujian fungsional dan lapang menghasilkan metrik operasional yang divisualisasikan dalam sesi log CSV dan laporan pengujian.

4.1 Hasil Pengujian

Fingerprint (R305) Pengujian skenario variasi kondisi jari (basah, kering, berminyak, berdebu, posisi miring) menunjukkan tingkat keberhasilan rata-rata 85.8% pada sampel uji yang dilakukan. Kegagalan terutama terjadi pada kondisi jari sangat basah atau sangat kotor; mekanisme retry dan prompt pengguna mengurangi false negative pada beberapa kasus.

Nomor Test	Status Sidik jari	Respon fingerprint 1	Respon fingerprint 2	Waktu respond
1	Basah	Tidak Valid	Valid	0.2
2	Basah	Tidak Valid	Valid	0.4
3	Kering	Valid	Valid	0.1
4	Kering	Valid	Valid	0.2
5	Berminyak	Valid	Valid	0.2
6	Berminyak	Valid	Valid	0.1
7	Berdebu	Tidak Valid	Valid	0.2
8	Berdebu	Valid	Valid	0.1
9	Posisi miring	Valid	Valid	0.2
10	Posisi miring	Valid	Valid	0.1

ANPR (YOLO + OCR) Dari 65 gambar uji, 22 gambar menghasilkan pembacaan OCR yang sempurna. Ini memberikan success rate OCR sebesar 33.85%. Analisis error menunjukkan penyebab utama kegagalan: blur akibat gerak, pencahayaan rendah/kontras buruk, sudut pengambilan gambar yang ekstrem, refleksi pada plat, dan variasi format/warna plat yang tidak cukup direpresentasikan di dataset.



Sistem terintegrasi (end-to-end) Alur entry/exit yang menggabungkan verifikasi sidik jari → capture → deteksi plat → OCR → logging → buka palang berjalan dengan baik pada skenario ideal. Fallback yang diterapkan (plate="unknown", flag untuk review manual) menjaga kontinuitas layanan ketika ANPR gagal. CSV sessions menyimpan entry_iso, exit_iso, user_id, plate, entry/exit image path, duration, fee, yolo_conf, ocr_conf.

Kinerja dan latensi Rata-rata latency untuk pipeline YOLO+OCR pada perangkat pengujian (Raspberry Pi tanpa akselerator) berada di kisaran yang memerlukan optimasi untuk pengalaman real-time; fingerprint match <1 detik pada rata-rata kondisi normal. Penulisan CSV dan penyimpanan gambar terukur cepat namun perlu lock/atomic write untuk keandalan saat beban tinggi.

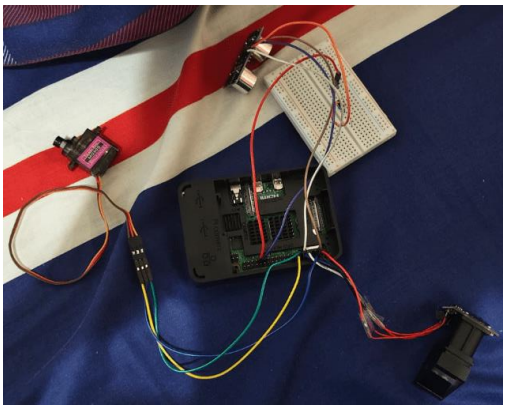
Antarmuka dan operabilitas UI Tkinter menampilkan status perangkat, preview kamera dan crop plat, tombol

operasi (Scan Entry/Exit, Manual Open), log singkat dan last fee. UI tetap responsif karena pekerjaan berat (YOLO/OCR, disk I/O) dijalankan di worker threads.

4.2 Pembahasan

Interpretasi success rate OCR 33.85% menunjukkan bahwa deteksi bbox dan/atau pra-pemrosesan citra masih perlu ditingkatkan. Meskipun YOLO mampu menemukan area plat pada banyak frame, kualitas crop dan kondisi citra menentukan hasil OCR. Dataset latihan yang berasal dari Kaggle besar namun belum sepenuhnya mencakup variasi kondisi lokal (refleksi, desain plat lokal, angle) menjadi faktor pembatas utama.

Kekuatan dari integrasi sistem ini adalah kombinasi fingerprint + ANPR memberikan lapisan verifikasi ganda saat fingerprint valid tetapi OCR gagal, sistem tetap dapat mengidentifikasi pengguna dan mencatat sesi. Desain modular memudahkan penggantian atau perbaikan modul tanpa mengganggu alur sistem lain.



Kelemahan dari sistem itu adalah Raspberry Pi tanpa akselerator GPU membatasi throughput YOLO; beberapa frame mengalami latency tinggi. Fingerprint rentan terhadap kondisi jari buruk; OCR sensitif terhadap noise. Logging pada CSV efektif tetapi memerlukan kontrol konkurensi dan strategi rotasi untuk skala produksi.

Validitas eksperimen pengujian dilakukan di lapangan dan depan rumah penulis. metrik mencerminkan kondisi nyata namun perlu dataset lapang yang lebih representatif serta pengujian jangka panjang untuk menyimpulkan stabilitas operasional.

5. Kesimpulan

Bagian terdiri atas simpulan dan saran atas penelitian hasil penelitian.

5.1 Simpulan

Prototipe Sistem Parkir Pintar berbasis Raspberry Pi berhasil diimplementasikan dengan integrasi fingerprint (R305), sensor ultrasonik, kontrol servo, dan pipeline ANPR (YOLO + OCR). Sistem bekerja end-to-end (verifikasi masuk/keluar, deteksi plat, OCR, pencatatan CSV, buka palang) dengan mekanisme fallback (plate="unknown"). Hasil uji menunjukkan fingerprint success ≈85.8% dan OCR perfect 22/65 (≈33.85%). Kendala utama: kualitas citra (pencahayaan, blur, sudut) dan keterbatasan sumber daya Raspberry Pi yang memengaruhi latency dan akurasi ANPR.

5.2 Saran

Prototipe Sistem Parkir Pintar berbasis Raspberry Pi yang menggabungkan fingerprint R305, sensor ultrasonik, servo, dan pipeline ANPR (YOLO+OCR) berhasil diimplementasikan dan bekerja end-to-end (verifikasi masuk/keluar, deteksi plat, OCR, pencatatan CSV, buka palang) dengan fallback plate="unknown" saat ANPR gagal; pengujian menunjukkan fingerprint ≈85,8% dan OCR sempurna 22/65 (≈33,85%). Kendala utama adalah kualitas citra (pencahayaan, blur, sudut) dan keterbatasan komputasi Pi yang memengaruhi latency dan akurasi ANPR. Rekomendasi prioritas: perluas dataset lokal dan augmentasi, optimasi model untuk edge (quantization/ONNX/TFLite) atau gunakan akselerator, terapkan preprocessing OCR adaptif dan multi-pass, serta perkuat operasional (retry, atomic CSV write, watchdog).

Ucapan Terima Kasih

terima kasih kepada pembimbing, dosen, staf, dan pihak pendukung yang membantu pengadaan komponen dan uji lapang.

Daftar Rujukan

[1] P-ISSN E-ISSN : 2456-2343 berjudul “International Journal of Transportation Engineering and Traffic System IoT Based Smart Car Parking System (2024)" https://doi.org/10.37628/IJTETS

2] D. Priyantika, A. Marom, and H. Nurcahyanto, "mAnajemen Perparkiran Dan Kualitas Pelayanan Juru Parkir Dalam Kebijakan Retribusi Zona Parkir Tepi Jalan Umum," *Journal of Public Policy and Management Review*, vol. 13, no. 1, pp. 1 - 20, Dec. 2023. https://doi.org/10.14710/jppmr.v13i1.42440

[3] Muhaimin, Abdi E - ISSN : 2579-373X, Berjudul ” Implementasi Fingerprint Android Pada Sistem Keamanan Laci Meja (2022)” l : http://ojsamik.amikmitragama.ac.id

[4] Dan Pemantauan implementasi Raspberry Pi Pada Sistem Issn : 2355-9365 berjudul, “Implementasi Raspberry Pi Pada Sistem (2024)” Https://Doi.Org/10.30651/Cl.V7i01.20459

[5] Salsabilah B Issn : 3062-8849. Berjudul “Pengujian Akurasi Sensor Ultrasonik Hc-sr04 Dalam Mengukur (2024)”. Https://Doi.Org/10.25047/Nacia.V2i1.236

6] kurniawan, Helmi Issn : 2598-8158. Berjudul "Implementasi Teknologi Digital File Csv Sebagai Data Dukung Aplikasi Data Stm Dan Pemakaman Berbasis Cloud (2024)". Https://Doi.Org/10.31764/Jmm.V8i6.27560

[7] nur Ikhsan, M. P-issn: 2527 – 9661 | E-issn: 2549 - 2837. Berjudul "String (Satuan Tulisan Riset Dan Inovasi Teknologi) Sistem Keamanan Sepeda Motor Dengan Teknologi Biometrik Sidik Jari Menggunakan Sensor Fingerprint R305 (2022). Http://Dx.Doi.Org/10.30998/String.V7i2.13797

[8] arunda Engga Reswara e-issn : 2623-1700 | P-issn : 1693-3672. Berjudul "Rancang Bangun Sistem Smart Parking Berbasis Internet Of Things (Iot) (2024)". Https://Doi.Org/10.52643/Jti.V11i1.6422

[9] Gabriel, Adilson ISSN : 2442-5826 Berjudul "P Perancangan Dan Realisasi Biometric Fingerprint Attendance Dan Web Monitoring System (2020)"

[10]Hendrawan, Beni P-ISSN: 2549-6131|E-ISSN: 2549-614X . Berjudul "Rancang Bangun Sistem Parkir Dengan Menggunakan Fingerprint (2020)." https://doi.org 10.33857/patj.v4i2.378

[11]Agustiawan P-ISSN: 2549-6131|E-ISSN: 2549-614X. Berjudul "Rancang Bangun Sistem Parkir Dengan Menggunakan Fingerprint (2020)" https://doi.org/10.33857/patj.v4i2.378

[12] Raghu Kumar, Lingamallu. Berjudul "Advanced Parking System (2024)" https://doi.org/10.1051/matecconf/202439201085

[13]Santanu, Jonatan Rajif E-issn: 2986-7592, Berjudul "Deteksi Dan Pembacaan Plat Nomor Kendaraan Menggunakan Yolo Dan Easyocr (2025)" https://Doi.Org/10.54082/Jiki.33

[14]Juju Jumadi, P-ISSN : 2303-3142 E-ISSN : 2548-8570 Berjudul "Pengolahan Citra Digital Untuk Identifikasi Objek Menggunakan Metode Hierarchical Agglomerative Clustering (2021)" https://doi.org/10.23887/jst-undiksha.v10i2.33636

[15]Muhamad Irsan E – ISSN : 2987-0135 berjudul "Implementasi Aplikasi Pandas (Phyton) Dalam Mengelola Data Excel Sebagai Media Persiapan Pelaporan Nilai Raport Siswa (2023)" https://doi.org/10.59837/jpmba.v2i4.977