

SISTEM MONITORING AREA PARKIR BERBASIS KAMERA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK UNTUK KETERSEDIAAN SLOT PARKIR

Delia Ramadani^{1,*}, Annisa Rahmawati², dan Nisrina Fitriyani³

^{1,2,3} Program Studi Sistem Informasi
Fakultas Ilmu Komputer, Universitas Pamulang,
Jl. Raya Puspittek, Buaran, Kec. Pamulang, Kota Tangerang Selatan, Banten 15310

E-mail: deliaramadhanilia@gmail.com¹, annisarahmawati339@gmail.com²,
nisrinafitriyani1511@gmail.com³

ABSTRAK

SISTEM MONITORING AREA PARKIR BERBASIS KAMERA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK UNTUK KETERSEDIAAN SLOT PARKIR. Permasalahan keterbatasan informasi mengenai ketersediaan lahan parkir di area publik sering menyebabkan kemacetan, antrean kendaraan, serta pemborosan waktu dan bahan bakar. Penelitian ini bertujuan mengembangkan sistem monitoring area parkir berbasis kamera menggunakan Convolutional Neural Network untuk mendeteksi ketersediaan slot parkir secara real-time. Metode penelitian dilakukan dengan pengumpulan dataset citra slot parkir menggunakan kamera statis yang kemudian dilabeli menjadi dua kelas, yaitu kosong dan terisi. Data diproses menggunakan arsitektur CNN yang terdiri dari layer Conv2D, MaxPooling2D, dan Dense dengan framework TensorFlow dan bahasa pemrograman Python. Sistem diimplementasikan ke dalam antarmuka web berbasis Flask untuk menampilkan status slot parkir secara visual. Hasil pengujian menunjukkan akurasi pelatihan mencapai 99% dan akurasi pengujian berada pada rentang 94% hingga 96%, dengan akurasi keseluruhan sebesar 65% dan nilai F1-score sebesar 0,64. Pengujian fungsional menggunakan metode black box menunjukkan seluruh fitur sistem berjalan sesuai dengan kebutuhan. Berdasarkan hasil tersebut, sistem ini dapat digunakan sebagai solusi awal pemantauan ketersediaan lahan parkir secara efisien tanpa kehadiran langsung petugas.

Kata kunci: Sistem, Monitoring, Parkir, Berbasis, Kamera

ABSTRACT

CAMERA-BASED PARKING AREA MONITORING SYSTEM USING CONVOLUTIONAL NEURAL NETWORK FOR PARKING SLOT AVAILABILITY. The limited availability of information regarding parking spaces in public areas often leads to traffic congestion, vehicle queues, and inefficient use of time and fuel. This study aims to develop a camera-based parking area monitoring system using a Convolutional Neural Network to detect parking slot availability in real time. The research method involves collecting a dataset of parking slot images using a static camera, which are then labeled into two classes, namely vacant and occupied. The data are processed using a CNN architecture consisting of Conv2D, MaxPooling2D, and Dense layers, implemented with the TensorFlow framework and the Python programming language. The system is integrated into a Flask-based web interface to visually display parking slot status. The experimental results show that the training accuracy reaches 99%, while the testing accuracy ranges from 94% to 96%, with an overall accuracy of 65% and an F1-score of 0.64. Functional testing using the black-box method indicates that all system features operate as expected. Based on these results, the proposed system can serve as an initial solution for efficient parking space availability monitoring without the need for direct human supervision.

Keywords: Camera, Based, Parking, Monitoring, System

1. PENDAHULUAN

Seiring dengan meningkatnya jumlah kendaraan bermotor akibat pertumbuhan penduduk serta meningkatnya taraf hidup masyarakat, kebutuhan akan sistem parkir yang lebih efisien menjadi semakin mendesak. Salah satu pendekatan yang dapat diterapkan untuk mengatasi persoalan ini adalah sistem parkir cerdas (smart parking), yaitu sistem yang mengintegrasikan teknologi dalam proses pengelolaan parkir, khususnya dalam mendeteksi slot parkir yang tersedia. Teknologi berbasis kamera dengan dukungan deep learning menjadi pilihan yang menjanjikan karena mampu mendeteksi keberadaan kendaraan melalui citra visual. Solusi ini dinilai lebih hemat biaya dibandingkan penggunaan sensor individual, sebab satu kamera mampu mengawasi beberapa slot parkir sekaligus [Rahman and Dafitri, 2022].

2. TINJAUAN PUSTAKA

2.1 Visi Komputer (Computer Vision)

Computer vision merupakan cabang dari kecerdasan buatan yang fokus pada bagaimana sebuah sistem atau mesin mampu "melihat" dan memahami objek dalam bentuk gambar atau video.

2.2 Sistem Monitoring Parkir

Pada dasarnya, sistem ini dirancang untuk memantau kondisi parkir secara otomatis dan real-time, baik untuk mendeteksi kendaraan yang masuk dan keluar maupun menginformasikan ketersediaan slot parkir.

2.3 Deteksi Objek dan Slot Parkir

Deteksi objek dalam konteks sistem parkir digunakan untuk mengenali posisi dan kondisi slot parkir berdasarkan tampilan visual atau citra yang ditangkap oleh kamera. Dalam penerapannya, sistem akan membaca gambar area parkir, kemudian mengidentifikasi apakah suatu slot parkir masih kosong atau sudah terisi dengan menampilkan penanda tertentu, seperti bounding box [Sani and Ayyasy, 2022].

2.4 Python dalam Pengolahan Citra

Python dikenal sebagai salah satu bahasa pemrograman yang sangat cocok digunakan dalam bidang pengolahan citra digital. Kelebihan utama Python terletak pada struktur sintaks yang sederhana dan mudah dibaca, sehingga sangat membantu baik untuk pemula maupun profesional dalam merancang alur kerja pemrosesan gambar.

3. METODE

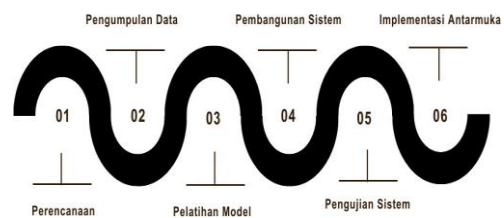
3.1 METODE PENELITIAN

Penelitian ini menggunakan metode kuantitatif yang berfokus pada pengumpulan data numerik, pengolahan data citra digital, serta analisis performa sistem untuk mengukur efektivitas model klasifikasi status slot parkir berbasis Convolutional Neural Network (CNN).

3.2 TAHAPAN PENELITIAN

Penelitian ini dilaksanakan melalui tahapan yang dirancang secara sistematis untuk memastikan bahwa proses pengembangan sistem monitoring parkir dapat berjalan secara optimal.

Pada Gambar 1 terdapat tahapan untuk mengembangkan sistem monitoring parkir, berikut penjelasan lebih rinci mengenai setiap tahapan yang dilakukan dalam penelitian ini:



Gambar 1. Tahapan Penelitian

a. Perencanaan

Tahap perencanaan merupakan fondasi awal dalam proses pengembangan sistem monitoring area parkir. Pada tahap ini, peneliti merumuskan tujuan utama yaitu menciptakan sistem yang mampu mendeteksi kondisi slot parkir secara otomatis melalui media kamera, tanpa perlu kehadiran langsung dari petugas di lapangan. Tujuan tersebut menjadi landasan untuk menentukan metode dan teknologi yang akan digunakan sepanjang proses penelitian [Budi et al., 2024].

b. Pengumpulan Data

Pengumpulan data dilakukan secara manual dengan menggunakan kamera statis untuk merekam kondisi slot parkir di area tertentu. Setiap citra menggambarkan kondisi riil dari slot yang dalam keadaan kosong maupun terisi kendaraan. Proses ini bertujuan untuk membangun dataset dasar yang akan digunakan dalam pelatihan model klasifikasi setelah gambar dikumpulkan, setiap citra dikategorikan ke dalam dua

kelas utama: empty dan occupied [Sani and Ayyasy, 2022].

c. Pelatihan Model

Pelatihan model merupakan tahap penting dalam membangun sistem klasifikasi yang mampu membedakan antara slot parkir yang kosong dan yang terisi berdasarkan citra. Proses inti pada CNN adalah operasi konvolusi, di mana filter (kernel) bergerak melintasi gambar dan menghitung fitur dengan rumus:

$$S(i, j) = \sum_m \sum_n X(i + m, j + n) \cdot K(m, n)$$

Gambar 2. Rumus Convolutional Neural Network (CNN)

di mana X adalah input gambar, K adalah filter, dan S(i,j) adalah output feature map di posisi (i,j). Hasil konvolusi ini kemudian melewati fungsi aktivasi ReLU:

$$f(x) = \max(0, x)$$

Gambar 3. Rumus *pooling*

untuk memperkenalkan *non-linearitas* dan menyingkirkan nilai negatif. Setelah itu, dilakukan proses *pooling* untuk mereduksi dimensi data, biasanya menggunakan *max pooling*:

$$Y = \max\{x_1, x_2, x_3, x_4\}$$

Gambar 4. Rumus *fully connected*

Setelah fitur penting diekstrak dan dipadatkan, data diflatten dan dimasukkan ke dalam lapisan dense (*fully connected*), lalu diproses dengan fungsi aktivasi softmax pada *output layer* untuk menghasilkan dua kemungkinan kelas prediksi: *occupied* (terisi) dan *empty* (kosong).

d. Pembangunan Sistem

Pembangunan sistem dilakukan dengan mengintegrasikan model klasifikasi hasil pelatihan ke dalam lingkungan pengembangan berbasis *Python* dan *OpenCV*. Hasil klasifikasi kemudian diolah dalam bentuk teks atau visual sebagai penanda status slot.

e. Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh fungsi berjalan sesuai dengan yang diharapkan, khususnya dalam mendeteksi dan menampilkan status slot parkir secara otomatis.

Metode yang digunakan adalah *black box testing*, yaitu pengujian berdasarkan *output* sistem tanpa melihat struktur kode internal. Pengujian dilakukan dengan memberikan input berupa video parkir dan mencocokkan hasil klasifikasi sistem dengan kondisi nyata.

3.3 METODE PENGEMBANGAN SISTEM

Dalam penelitian ini, peneliti menggunakan metode *Waterfall* sebagai pendekatan dalam pengembangan sistem *monitoring* area parkir. Model ini dipilih karena memberikan alur kerja yang terstruktur dan mudah dikendalikan, sehingga sangat cocok untuk proyek dengan ruang lingkup yang sudah ditentukan sejak awal, seperti sistem *monitoring* parkir ini.

Gambar 5 . Metode Waterfall



Berikut penjelasan mengenai tahapan-tahapan dalam metode *Waterfall* yang diterapkan dalam penelitian ini.

a. Analysis

Pada tahap analisis, peneliti melakukan proses identifikasi dan pengumpulan kebutuhan sistem, baik dari sisi teknis maupun fungsional.

b. Design

Tahap desain bertujuan untuk merancang sistem secara menyeluruh sebelum dilakukan pengembangan.

c. Implementation

Tahap implementasi merupakan realisasi dari hasil desain ke dalam bentuk program yang dapat dijalankan. Pada tahap ini, peneliti mulai mengembangkan sistem menggunakan bahasa pemrograman *Python*. Proses ini mencakup pembuatan modul deteksi slot parkir menggunakan library *OpenCV* dan pengolahan citra, serta pembangunan antarmuka pengguna menggunakan Tkinter di *Visual Studio Code*.

d. Testing

Uji coba dilakukan dengan berbagai kondisi slot parkir (terisi sebagian, penuh, dan kosong) untuk memastikan sistem dapat secara akurat mengenali status tiap slot. Tujuan dari tahap ini adalah untuk menemukan bug dan memastikan keandalan sistem sebelum digunakan oleh pengguna sebenarnya.

e. Maintenance

Tahap terakhir adalah pemeliharaan, di mana sistem yang telah dikembangkan akan terus dipantau performanya dan dilakukan perbaikan jika ditemukan kesalahan atau kekurangan selama penggunaan.

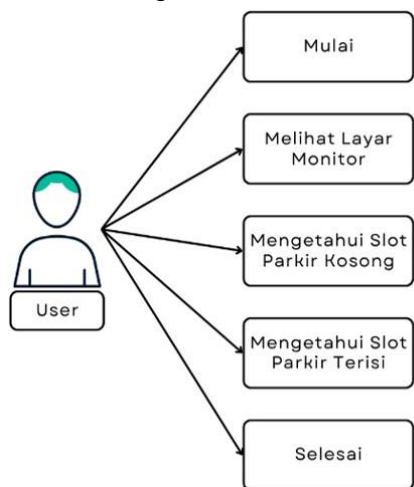
4. HASIL DAN PEMBAHASAN

4.1 Perancangan Sistem

Pada tahap perancangan sistem, peneliti merancang alur kerja dan struktur sistem secara terperinci untuk memastikan setiap komponen dapat berfungsi dengan optimal dalam memenuhi tujuan penelitian.

a. Flowchart Sistem

Flowchart sistem ini menggambarkan alur penggunaan aplikasi *monitoring* area parkir secara keseluruhan. Diagram ini memperlihatkan bagaimana pengguna berinteraksi dengan sistem, dimulai dari tampilan utama di mana pengguna dapat memilih untuk masuk sebagai user atau langsung ke fitur monitoring. Pada fitur monitoring, sistem akan menampilkan kondisi slot parkir secara real-time menggunakan indikator visual, seperti warna merah untuk slot terisi dan hijau untuk slot kosong.

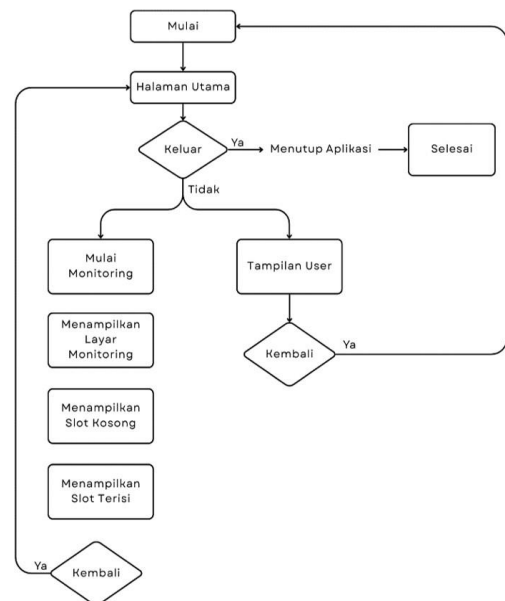


Gambar 6. Flowchart Sistem

b. Use Case Diagram

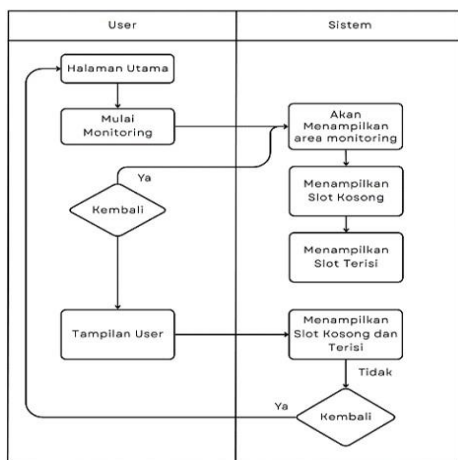
Use case diagram merupakan jenis diagram (UML) yang digunakan dalam rekayasa perangkat lunak untuk secara visual merepresentasikan interaksi antara berbagai aktor (pengguna atau sistem eksternal) dan suatu sistem. Diagram ini menggambarkan bagaimana pengguna berinteraksi dengan sistem untuk mencapai tujuan tertentu [Rasiban and Riyadi, 2024].

Gambar 7. Flowchart Sistem



c. Activity Diagram

Activity Diagram adalah jenis diagram dalam (UML) yang digunakan untuk menggambarkan aliran kerja atau aktivitas dalam suatu sistem atau proses. Diagram ini menyajikan serangkaian kegiatan, tindakan, dan keputusan yang terjadi sepanjang waktu [Rasiban and Riyadi, 2024].



Gambar 8. Activity Diagram

4.2 Alat dan Bahan Penelitian

Pada tahap pengembangan sistem pen-
erjemah bahasa isyarat ini, pene-
liti menggunakan berbagai alat dan bahan yang
mendukung keberhasilan proses penelitian.

a. Alat Penelitian

Perangkat Keras. Pada penelitian ini,
perangkat keras utama yang digu-
nakan untuk pengembangan dan pelatihan
model adalah sebagai berikut:

Tabel 1. Perangkat Keras

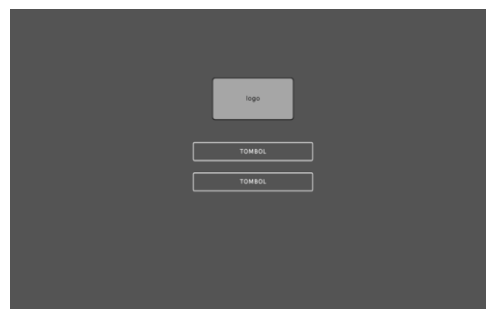
Nama Perangkat	Spesifi- kasi
Laptop Asus Vi- voBook Flip TM420	Ryzen 5 5500U

Tabel 2. Perangkat Lunak

Nama Perangkat Lunak	Versi
Jupyter Notebook	4.3.4
Tensorflow	2.19.0
Python	3.11.1
Windows	11

4.3 Tampilan Aplikasi

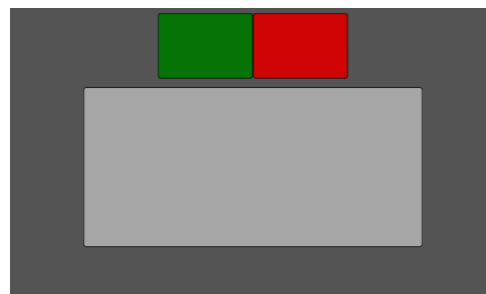
a. Tampilan Home Page



Gambar 9. Tampilan Home Page

Pada gambar 9 Halaman utama aplikasi
berisi dua tombol yaitu tombol mulai moni-
toring dan tampilan pengguna, keduanya
untuk memonitoring, perbedaanya mulai
monitoring menampilkan area parkir dan
slot parkir, sedangkan tampilan pengguna
hanya menampilkan slot ketersediaan
area parkir saja.

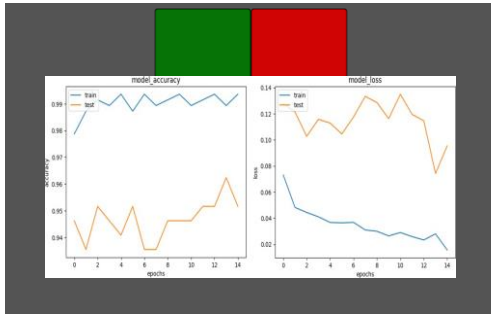
b. Tampilan Area Monitoring



Gambar 10. Tampilan area monitoring

Gambar 10 menampilkan antarmuka
utama dari fitur monitoring area parkir, di
mana pengguna dapat melihat kondisi
slot parkir secara *real-time*. Sistem secara
otomatis menghitung jumlah slot yang
tersedia dan terisi, yang ditampilkan me-
lalui indikator warna—hijau untuk slot
kosong dan merah untuk slot terisi.
Tampilan ini juga dilengkapi dengan tom-
bol “Kembali” yang berfungsi untuk kem-
bali ke halaman utama, serta tombol “Full-
screen” yang memungkinkan pengguna
memperbesar tampilan monitoring agar
lebih jelas dan nyaman saat melakukan
pemantauan.

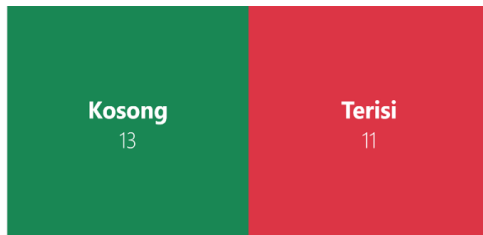
c. Tampilan *Full Screen Monitoring*



Gambar 11. Tampilan area monitoring *fullscreen*

Pada gambar 11 terdapat tampilan yang seluruh fokus pengguna diarahkan pada visual area parkir, sehingga pemantauan kondisi slot menjadi lebih jelas dan detail.

d. Tampilan Ketersediaan Slot Parkir

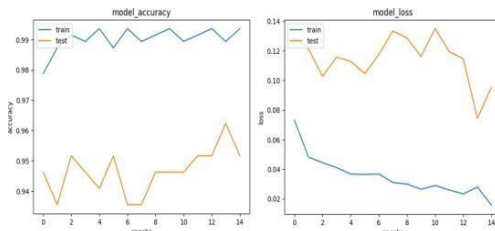


Gambar 12. Tampilan Slot ketersediaan Parkir

Pada gambar 12 terdapat tampilan slot ketersediaan parkir menunjukkan kondisi masing-masing lahan parkir dalam sistem secara *real-time*, dengan menggunakan indikator visual berupa warna. Slot yang kosong akan ditandai dengan warna hijau, sedangkan slot yang terisi ditandai dengan warna merah. Visualisasi ini dirancang untuk memudahkan pengguna dalam memahami status ketersediaan parkir secara cepat dan intuitif tanpa harus membaca data numerik.

4.4 Hasil Pelatihan Model

a. Kurva Hasil Pelatihan Model

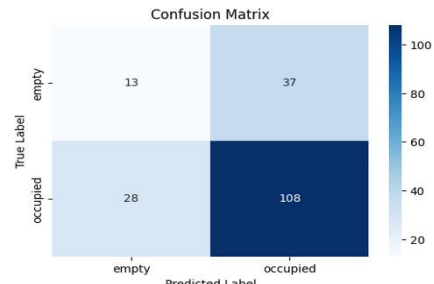


Gambar 13. Kurva Hasil Pelatihan Model

Gambar 13 menunjukkan grafik evaluasi model *CNN* berdasarkan akurasi dan loss selama 15 epoch pelatihan. Grafik sebelah kiri

menggambarkan akurasi model, di mana akurasi pada data latih (*train*) sangat tinggi mendekati 99%, sedangkan akurasi pada data uji (*test*) berada di kisaran 94%–96% dan terlihat tidak stabil.

b. Convolution Matrix



Gambar 14. ^{Predicted Label} *Confusion Matrix*

Pada gambar 14 di atas menunjukkan hasil evaluasi model CNN menggunakan confusion matrix dan laporan klasifikasi pada sistem pendeteksi ketersediaan parkir. Confusion matrix digunakan untuk melihat jumlah prediksi yang benar dan salah dari dua kelas, yaitu empty (kosong) dan occupied (terisi). Dari hasil tersebut, terlihat bahwa model memprediksi 108 slot parkir terisi dengan benar, dan 13 slot kosong juga berhasil diprediksi dengan benar.

c. Classification Report

	precision	recall	f1-score	support
empty	0.32	0.26	0.29	50
occupied	0.74	0.79	0.77	136
accuracy			0.65	186
macro avg	0.53	0.53	0.53	186
weighted avg	0.63	0.65	0.64	186

Gambar 15. Classification Report

Pada gambar 15 terdapat penjelasan mengenai model *CNN* yang mencakup nilai *accuracy*, *macro average* dan *weight average*. Ketiganya digunakan untuk mendeteksi status lahan parkir menunjukkan performa yang cukup baik dengan akurasi sebesar 65%.

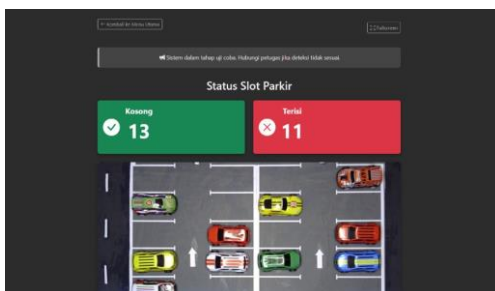
d. Tampilan *home page*



Gambar 16. Halaman *home page*

Pada Gambar 16 ditampilkan antarmuka awal dari sistem monitoring area parkir. Pada tampilan tersebut terdapat tombol "Mulai Monitoring" yang berfungsi sebagai navigasi untuk memulai proses pemantauan.

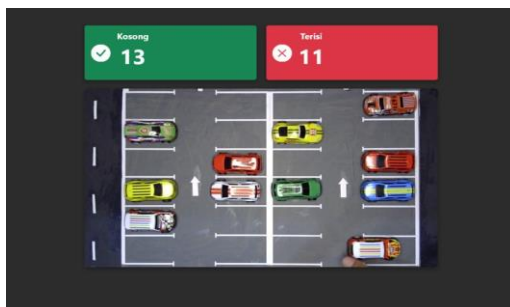
e. Tampilan Monitoring Area Parkir



Gambar 17. Halaman *Monitoring Area Parkir*

Pada Gambar 17 ditampilkan halaman utama dari sistem monitoring area parkir berbasis kamera. Antarmuka ini menyajikan informasi *real-time* mengenai status ketersediaan slot parkir, yang ditampilkan dalam dua kotak indikator utama: "Kosong" dan "Terisi". Masing-masing kotak menampilkan jumlah *slot* yang tersedia dan terisi, dengan warnahijau untuk slot kosong dan merah untuk slot terisi, guna memudahkan identifikasi visual bagi pengguna.

f. Tampilan *Full Screen Monitoring Area Parkir*



Gambar 18. Tampilan *Full Screen Area Monitoring*

Pada gambar 18 menampilkan antarmuka monitoring area parkir dalam mode tampilan penuh (*fullscreen*). Pada tampilan ini, sistem secara khusus menampilkan visualisasi area parkir secara lebih fokus dan mendalam, tanpa menampilkan elemen tambahan seperti tombol navigasi atau indikator status slot. Desain ini bertujuan untuk memberikan kesan tampilan yang lebih bersih dan jelas, sehingga pengguna dapat lebih mudah dalam melakukan pemantauan terhadap kondisi slot parkir secara *real-time*.

g. Implementasi Kode Program

```
# SETUP
total_images = 30
train_ratio = 0.6
image_size = (130, 65)
output_dir = 'C:/Users/ASUS/Downloads'
camera_index = 1

# Input label
label = input("Masukkan label (occupied / empty): ").strip().lower()
if label not in ['occupied', 'empty']:
    print("Label tidak valid. Gunakan 'occupied' atau 'empty'.")
    exit()

# Persiapan folder
train_path = os.path.join(output_dir, 'train', label)
test_path = os.path.join(output_dir, 'test', label)
os.makedirs(train_path, exist_ok=True)
os.makedirs(test_path, exist_ok=True)
```

Gambar 19. Kode Pengambilan Dataset

Pada Gambar 19 merupakan bagian awal dari proses pengambilan data gambar yang digunakan untuk melatih model klasifikasi slot parkir. Kode ini terdiri dari tiga bagian utama, yaitu setup parameter, input label, dan persiapan folder penyimpanan data. Setup parameter, bagian ini digunakan untuk mengatur parameter dasar akan digunakan dalam proses pengambilan dan pembagian data.

h. Kode Data Collection

```
width = 187
height = 105

save_dir = 'cropped_img'

if not os.path.exists(save_dir):
    os.makedirs(save_dir)

def save_cropped_img(img, pos, index):
    cropped_img = img[pos[1]:pos[1]+height, pos[0]:pos[0]+width]
    save_path = os.path.join(save_dir, f'roi_{index}.jpg')
    cv2.imwrite(save_path, cropped_img)
    print(f'saved cropped image: {save_path}')

try:
    with open('carposition.pkl', 'rb') as f:
        positionList = pickle.load(f)
except:
    positionList = []
```

Gambar 20. Kode Data *Collection*

Potongan kode pada Gambar 20 merupakan bagian dari sistem yang bertugas untuk melakukan proses pemotongan

(cropping) gambar area parkir berdasarkan koordinat posisi slot yang telah ditentukan sebelumnya.

i. Kode Inisialisasi Sistem Monitoring

```
app = Flask(__name__)
model = load_model('train_data/model_final (1).h5')
class_dictionary = {0: 'no_car', 1: 'car'}
cap = cv2.VideoCapture(1)
if not cap.isOpened():
    raise RuntimeError("Video tidak bisa dibuka. Periksa file video.")
with open('train_data/carposition.pkl', 'rb') as f:
    posisist = pickle.load(f)
width, height = 187, 105
```

Gambar 21. Kode Inisialisasi Sistem

Pada Gambar 21 merupakan bagian dari proses inisialisasi sistem monitoring parkir berbasis kamera. Pada bagian ini, dilakukan beberapa tahapan penting seperti Pada Gambar 21 merupakan bagian dari proses inisialisasi sistem monitoring parkir berbasis kamera. Pada bagian ini, dilakukan beberapa tahapan penting seperti pemanggilan model deteksi yang sudah dilatih, pembacaan koordinat posisi slot parkir, serta pengaturan sumber video yang digunakan sebagai input untuk proses deteksi kendaraan secara real-time.

4.5 Pengujian Sistem

a. Pengujian Black Box

Pengujian ini dilakukan untuk mengevaluasi fungsionalitas sistem deteksi slot parkir berdasarkan respons terhadap input video secara langsung. Tujuan dari pengujian ini adalah untuk memastikan bahwa setiap fitur utama dapat berjalan dengan baik dan sesuai harapan tanpa perlu mengetahui detail struktur kode program secara langsung. Berikut penjelasannya terdapat pada Gambar 22 mengenai *black box*.

No.	Fungsi	Output	Tampilan	Hasil Uji Coba
1.	Halaman Home Page	Fitur Tombol Mulai Monitoring		Berhasil
2.	Halaman Utama Sistem Monitoring	Menampilkan Area parkir, Tombol Full Screen dan Tombol Kembali ke Home page		Berhasil
3.	Halaman Fullscreen	Menampilkan Area Monitoring		Berhasil
4.	Tampilan Area Kosong	Menghitung Jumlah Slot parkir Kosong		Berhasil
5.	Tampilan Area Terisi	Menghitung Jumlah Slot Parkir Terisi		Berhasil

Gambar 22. Black Box

5. KESIMPULAN

Sistem parkir yang dirancang bisa membantu mengurangi antrean kendaraan dan mencegah kemacetan di area parkir. Ini karena sistem ini memberikan informasi ketersediaan slot parkir secara langsung. Dengan adanya teknologi seperti sensor dan kamera, petugas bisa memantau kondisi parkir dari jauh melalui layar komputer atau perangkat lainnya.

DAFTAR PUSTAKA

- [1]. Q. Aini, N. Lutfiani, H. Kusumah, and M. S. Zahran. "Deteksi dan pengenalan objek dengan model machine learning: Model YOLO." *CESS (Journal of Computer Engineering System and Science)*, vol. 6, no. 2, pp. 192–193, 2021.
- [2]. E. S. Budi, A. N. Chan, P. P. Alda, and M. A. F. Idris. "Optimasi model machine learning untuk klasifikasi dan prediksi citra menggunakan algoritma convolutional neural network." *RESOLUSI: Rekayasa Teknik Informatika dan Informasi*, vol. 4, no. 5, pp. 502–509, 2024.
- [3]. W. Hermawansyah and E. Kusmara. "Perancangan desain user interface dan user experience pada website Epic Tour dengan menggunakan metode user centered design (UCD)." *Informatics, Science, and Technologies Journal*, vol. 12, no. 2, pp. 48–49, 2022.
- [4]. A. Nur and P. R. Cahyani. "Evaluasi pengguna Jupyter Notebook pada Python dalam pembelajaran data science (Studi kasus: Kapal Titanic)." *Kohesi: Jurnal Multidisiplin Saintek*, vol. 4, no. 10, pp. 1–3, 2024.
- [5]. H. R. Pratama, D. P. Setiawan, and Yudiansyah. "Rancang bangun sistem parking space counter berbasis kamera webcam menggunakan OpenCV Python." *e-Proceeding of Engineering*, vol. 12, no. 1, pp. 33–34, 2025.
- [6]. Y. B. Pratama and N. P. Dalimunthe. "Implementasi teknik computer vision untuk deteksi viridiplantae pada lahan pasca tambang." *Bulletin of Computer Science Research*, vol. 3, no. 1, pp. 64–72, 2022.

[7]. A. I. Pulungan, Sumarno, I. Gunawan, H. S. Tambunan, and A. R. Damanik. "Rancang bangun sistem parkir dan ketersediaan slot parkir otomatis menggunakan Arduino." *Jurnal Ilmu Komputer dan Informatika (JIKI)*, vol. 2, no. 2, pp. 127–136, 2022.

[8]. D. Purnomo, M. D. Revansa, N. Shihira, H. Zikra, and M. L. Hamzah. "Perancangan sistem informasi penjualan sayuran berbasis website," in *IEEE Paper Template*, 2023, pp. 11–12.

[9]. S. Rahman and H. Dafitri. "Pengembangan convolutional neural network untuk klasifikasi ketersediaan ruang parkir." *EXPLORER: Journal of Computer Science and Information Technology*, vol. 2, no. 1, pp. 1–6, 2022.

[10]. Rasiban and S. Riyadi. "Analisis sentimen opini masyarakat terhadap stadion Jakarta Internasional Stadium (JIS) pada Twitter dengan perbandingan metode Naive Bayes dan Support Vector Machine." *Jurnal Sains dan Teknologi*, vol. 5, no. 3, pp. 801–809, 2024.

[11]. G. N. Rizkatama, A. Nugroho, and A. F. Suni. "Sistem cerdas penghitung jumlah mobil untuk mengetahui ketersediaan lahan parkir berbasis Python dan YOLO v4." *Edu Komputika*, vol. 8, no. 2, pp. 91–92, 2021.

[12]. A. Sani and D. H. Ayyasy. "Prototipe deteksi ketersediaan slot parkir berbasis pengolahan citra." *Journal of Applied Electrical Engineering*, vol. 6, no. 2, pp. 59–60, 2022.

[13]. I. G. A. N. C. Wardhana. "Monitoring slot parkir mobil berbasis komputer vision." Master's thesis, Politeknik Negeri Bali, 2022.

[14]. D. Wicaksono, D. P. Almeyda, I. M. M. Putra, and L. Malihatuningrum. "Analisis perbandingan metode pra pemrosesan citra untuk deteksi tepi Canny pada citra berbagai kondisi jalan menggunakan bahasa pemrograman Python." *Jurnal Teknologi dan Ilmu Komputer Prima (JUTIKOMP)*, vol. 7, no. 1, pp. 17–19, 2024.