

RANCANGAN SISTEM DETEKSI PELAT NOMOR KENDARAAN DENGAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN) DAN OPTICAL CHARACTER RECOGNITION (ORC) UNTUK KLASIFIKASI WILAYAH ADMINISTRATIF

Rian Saputra¹, Bambang Krismono², Muhammad Zulfikri³
riansaputra291200@gmail.com¹, bkrismono@universitasbumigora.ac.id²,
mzulfikri@universitasbumigora.ac.id³
Universitas Bumigora

ABSTRAK

Pelat nomor kendaraan merupakan identitas resmi kendaraan yang juga memuat informasi mengenai wilayah administratif pendaftaran kendaraan. Proses identifikasi wilayah berdasarkan pelat nomor secara manual memiliki keterbatasan dalam hal efisiensi dan pengolahan data dalam jumlah besar. Oleh karena itu, diperlukan suatu rancangan sistem yang mampu mendeteksi pelat nomor kendaraan secara otomatis, mengenali karakter pada pelat, serta mengklasifikasikan wilayah administratif berdasarkan kode yang terdapat pada pelat nomor. Penelitian ini bertujuan merancang sistem deteksi pelat nomor kendaraan menggunakan metode Convolutional Neural Network (CNN) dan Optical Character Recognition (OCR) untuk melakukan klasifikasi wilayah administratif. Metode penelitian yang digunakan adalah pendekatan kualitatif dengan tahapan studi literatur dan perancangan sistem menggunakan metode Rapid Application Development (RAD). Rancangan sistem terdiri dari proses input citra kendaraan, preprocessing citra yang meliputi resize, grayscale, noise reduction, dan normalisasi, kemudian dilanjutkan dengan deteksi lokasi pelat menggunakan CNN. Area pelat yang terdeteksi selanjutnya dilakukan cropping dan diproses menggunakan OCR untuk mengenali karakter huruf dan angka. Hasil pembacaan karakter kemudian dinormalisasi dan dicocokkan dengan database kode wilayah administratif. Rancangan klasifikasi difokuskan pada wilayah Pulau Lombok, yaitu Kota Mataram, Kabupaten Lombok Barat, Lombok Tengah, Lombok Timur, dan Lombok Utara. Hasil perancangan menghasilkan alur sistem yang mengintegrasikan deteksi objek menggunakan CNN, pembacaan karakter menggunakan OCR, serta pencocokan kode wilayah untuk menghasilkan informasi nomor pelat dan wilayah administratif kendaraan. Sistem yang dirancang dapat menjadi dasar pengembangan sistem identifikasi kendaraan secara otomatis. Keberhasilan rancangan dapat dievaluasi berdasarkan indikator deteksi pelat, akurasi pembacaan karakter OCR, akurasi nomor pelat, akurasi klasifikasi wilayah, waktu proses, dan keberhasilan keseluruhan sistem.

Kata Kunci: Pelat Nomor Kendaraan, Convolutional Neural Network, Optical Character Recognition, Klasifikasi Wilayah Administratif, Pengolahan Citra, Rapid Application Development.

PENDAHULUAN

Di setiap negara khususnya Indonesia pelat nomor kendaraan tidak hanya memiliki fungsi sebagai identitas yang melekat pada kendaraan, tetapi sebagai representasi dari wilayah administrative tertentu untuk memenuhi informasi penting dalam berbagai kebutuhan seperti analisis mobilitas kendaraan serta pengawasan kendaraan.

Di sisi lain, munculnya Industri 4.0 atau revolusi industri 4.0 telah mendorong kemajuan teknologi, informasi dan komunikasi untuk memodifikasi dan menegakkan perubahan dalam manajemen di seluruh domain, dari perusahaan hingga pemerintahan. (Izzuddin, 2022) konsep smart city atau kota cerdas telah menjadi gagasan pemerintah pusat yang diperjuangkan oleh kota-kota di seluruh penjuru dunia untuk mengatasi berbagai persoalan yang menyangkut tata kelola

Sehingga implementasi Automatic License Pelat Recognition (ALPR) yang terintegrasi langsung dengan klasifikasi wilayah administratif berdasarkan pelat nomor menjadi sumber data yang sangat penting sebagai analisis mobilitas antar wilayah serta perencanaan kebijakan dan tata kelola transformasi sehingga dalam konsep kota cerdas

sistem deteksi pelat kendaraan ini tidak bisa dihindari.

Dengan kemajuan deep learning berbasis convolution neural network (CNN) terbukti efektif dalam mendeteksi objek dan melakukan klasifikasi citra dengan tingkat akurasi yang tinggi, CNN mampu melakukan ekstraksi fitur special secara otomatis melalui proses convolution dan pooling, sehingga sesuai untuk melakukan pendeteksian area pelat nomor kendaraan dalam berbagai kondisi pencahayaan sudut pengambilan citra dan berbagai gangguan visual lain, penelitian implementasi CNN dengan menggunakan data training sebanyak 1615 menunjukkan bahwa CNN merupakan salah satu metode yang tepat dalam persoalan mengidentifikasi huruf dan angka. (Harani et al., 2019)

Meskipun penelitian telah banyak dilakukan dalam pengembangan sistem deteksi pelat kendaraan bermotor atau ALPR, namun masih begitu banyak tantangan dalam pengembangan serta peningkatan akurasi selain itu integrasi CNN dan OCR untuk melakukan klasifikasi wilayah administratif masih belum dikaji secara spesifik dalam konteks implementasi lokal. Ini membuka peluang baru dalam melakukan penelitian untuk menghasilkan system yang tidak hanya mendeteksi keberadaan pelat kendaraan tetapi juga mampu mengolah informasi menjadi data administrative yang bernilai strategis.

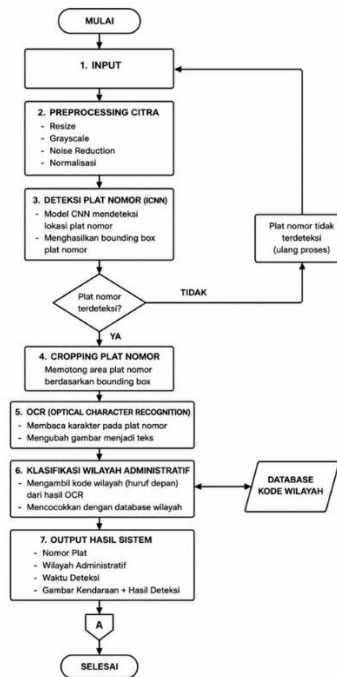
Dengan demikian, penelitian ini menjadi sangat penting dan relevan untuk mendukung tata kelola wilayah berbasis data dan kebutuhan transformasi cerdas selain itu penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem pengawasan berbasis komputer vision yang lebih efektif adaptif dan aplikatif.

Metode Penelitian

Untuk merancang sebuah sistem deteksi pelat nomor kendaraan yang mana dimulai dari perancangan sistem sampai dengan penulisan karya ilmiah, penulis menggunakan metode RAD (Rapid Application Development) yang mana merupakan proses pengembangan perangkat lunak dengan titik tekan pada waktu yang relative singkat. (Rahman, 2020) RAD dibuat oleh James Martin dan dirancang untuk melakukan pengembangan yang cepat dengan hasil yang berkualitas. Metode ini terdiri dari beberapa tahapan yaitu; perencanaan, analisis, perancangan dan implementasi.

METODOLOGI

Penelitian ini menggunakan pendekatan penelitian kualitatif karena penelitian bertujuan untuk menyusun rancangan (desain, model, atau prototipe) tanpa menguji atau mengimplementasikannya, Metode CNN digunakan untuk mendeteksi lokasi pelat nomor kendaraan pada gambar, sedangkan metode OCR digunakan untuk mengenali karakter huruf dan angka pada pelat nomor. Selanjutnya sistem melakukan identifikasi kode wilayah administratif berdasarkan hasil pembacaan karakter pelat nomor kendaraan.



Gambar 1 flowchart rancangan sistem deteksi pelat kendaraan bermotor

Rancangan Desain Sistem

Dalam karya tulis ini penulis membuat rancangan hingga pembentukan system menggunakan metode Rapid Application Development dimana dalam metode memiliki tahapan-tahapan yang terstruktur dan memiliki siklus yang cukup pendek dalam melakukan pengembangan perangkat lunak dan terbagi menjadi empat siklus dimana dua diantaranya yaitu; analisis persyaratan, dan analisis modeling telah dilakukan pada awal penulisan karya tulis atau tugas akhir ini selanjutnya yaitu desain modeling atau rancangan desain sistem dan konstruksi.

HASIL DAN PEMBAHASAN

Hasil perancangan sistem merupakan tahapan implementasi dari rancangan yang telah dibahas pada bab sebelumnya. Sistem yang dirancang bertujuan untuk melakukan deteksi pelat nomor kendaraan bermotor dari citra kendaraan, mengenali karakter yang terdapat pada pelat nomor menggunakan **Optical Character Recognition (OCR)**, kemudian mengidentifikasi kode wilayah administratif berdasarkan nomor kendaraan yang berhasil dibaca.

Input Sistem

Input sistem berupa citra kendaraan bermotor yang diperoleh menggunakan kamera atau file gambar. Citra yang digunakan dapat berupa kendaraan roda dua dengan posisi pelat terlihat pada gambar. karakteristik input yang dirancang meliputi:

Tabel 4. 1 Karakter Input

Parameter	Keterangan
Input	Citra kendaraan bermotor
Format	JPG/PNG/JPEG
Objek	Kendaraan roda dua
Komponen utama	Pelat nomor kendaraan
Sumber	file citra

Parameter	Keterangan
Output awal	Citra kendaraan yang akan diproses

Kualitas citra input menjadi salah satu faktor yang memengaruhi keberhasilan sistem. Citra dengan pencahayaan yang cukup, posisi pelat terlihat jelas, dan resolusi yang memadai akan memberikan hasil deteksi dan OCR yang lebih baik.

Preprocessing

Secara sederhana, Tujuan preprocessing adalah menghasilkan citra yang lebih sesuai untuk diproses oleh model CNN maupun OCR. proses preprocessing dapat digambarkan pada gambar 1 sebagai berikut:



Gambar 1 Preprocessing

Implementasi Kode Python:

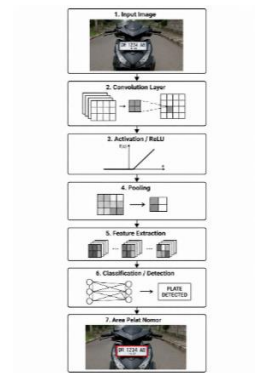
```

import cv2
def preprocess_image(image_path):
    # 1. Input Citra
    img = cv2.imread(image_path)
    # 2. Resize
    img_resized = cv2.resize(img, (640, 640), interpolation=cv2.INTER_AREA)
    # 3. Konversi Citra ke Grayscale
    gray = cv2.cvtColor(img_resized, cv2.COLOR_BGR2GRAY)
    # 4. Peningkatan Kontras menggunakan CLAHE
    clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))
    contrast_enhanced = clahe.apply(gray)
    # 5. Noise Reduction menggunakan Bilateral Filter
    ready_image = cv2.bilateralFilter(contrast_enhanced, d=9, sigmaColor=75, sigmaSpace=75)
    # Citra Siap Deteksi
    return ready_image
  
```

Model CNN

CNN digunakan sebagai metode *computer vision* untuk mengenali objek pelat nomor pada citra kendaraan. CNN dipilih karena mampu melakukan ekstraksi fitur secara otomatis berdasarkan pola visual yang terdapat pada citra.

Secara umum rancangan proses CNN digambarkan seperti pada gambar 2 :



Gambar 2 Proses CNN

Implementasi kode Python

```
import torch
import torch.nn as nn
import torch.nn.functional as F
class CNNPlateDetector(nn.Module):
    def __init__(self, num_classes=1):
        super(CNNPlateDetector, self).__init__()
        # 1. Convolution Layer
        # Input: 1 channel (grayscale), Output: 16 feature maps, kernel 3x3
        self.conv1 = nn.Conv2d(in_channels=1, out_channels=16, kernel_size=3, padding=1)
        # 2. Activation Layer (ReLU)
        self.relu = nn.ReLU()
        # 3. Pooling Layer (Max Pooling 2x2)
        self.pool = nn.MaxPool2d(kernel_size=2, stride=2)
        # Convolution kedua untuk memperdalam Feature Extraction
        self.conv2 = nn.Conv2d(in_channels=16, out_channels=32, kernel_size=3, padding=1)
        # 4. Feature Extraction to Fully Connected (Flatten)
        # Asumsi citra input berukuran 64x64, setelah 2x pooling menjadi 16x16
        self.fc1 = nn.Linear(32 * 16 * 16, 128)
        # 5. Classification / Detection Output Layer
        # Menghasilkan skor probabilitas / bounding box koordinat
        self.fc_out = nn.Linear(128, num_classes)
        def forward(self, x):
            # --- Tahap 1: Input Image ---
            # x adalah tensor citra input [Batch_Size, Channel, Height, Width]
            # --- Tahap 2, 3, 4: Conv -> ReLU -> Pooling ---
            x = self.conv1(x)    # Convolution Layer
            x = self.relu(x)     # Activation/ReLU
            x = self.pool(x)     # Pooling
            # Ekstraksi fitur tingkat lanjut
            x = self.conv2(x)    # Convolution Layer 2
            x = self.relu(x)     # Activation/ReLU
            x = self.pool(x)     # Pooling
            # --- Tahap 5: Feature Extraction ---
            # Ratakan matriks fitur (Flatten) menjadi vektor 1D
            x = x.view(x.size(0), -1)
            x = self.relu(self.fc1(x))
            # --- Tahap 6: Classification / Detection ---
            out = torch.sigmoid(self.fc_out(x)) # Output nilai antara 0 - 1 (Ada pelat / Tidak ada pelat)
            return out
```

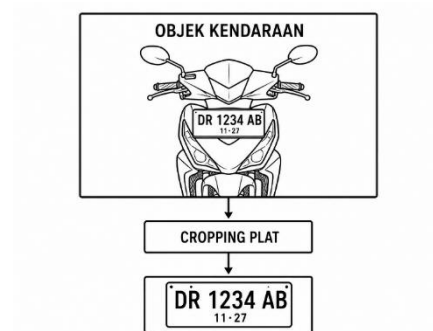
Cropping Pelat Nomor

Setelah area pelat berhasil dideteksi, sistem melakukan proses *cropping*. Proses ini bertujuan memisahkan area pelat dari keseluruhan citra kendaraan.

Misalnya citra kendaraan memiliki ukuran keseluruhan 1280×720 piksel. CNN mendeteksi area pelat pada koordinat tertentu, kemudian sistem mengambil bagian tersebut

untuk diproses lebih lanjut, Hasil *cropping* kemudian menjadi input untuk tahap OCR seperti pada gambar 3 dibawah ini.

Citra Kendaraan

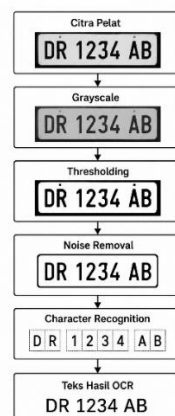


Gambar 3 cropping pelat

Model OCR

OCR digunakan untuk mengubah karakter yang terdapat pada citra pelat menjadi data teks. Tahapan ini merupakan bagian penting karena hasil OCR akan digunakan untuk menentukan kode wilayah administratif kendaraan.

Proses OCR dirancang melalui beberapa tahapan seperti pada gambar 4.4 :



Gambar 4 Proses OCR

Sebagai contoh, apabila sistem memperoleh citra pelat dengan tulisan: **DR 1234 AB** maka OCR diharapkan menghasilkan teks: **"DR1234AB"** Hasil tersebut kemudian diteruskan ke proses pengolahan teks.

Implementasi code Python

```

import cv2
import easyocr
import matplotlib.pyplot as plt
def process_plate_ocr(image_path):
    # 1. Citra Pelat (Input)
    img = cv2.imread(image_path)
    if img is None:
        raise FileNotFoundError("Gambar tidak ditemukan!")
    # 2. Grayscale
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    # 3. Thresholding (Binarisasi Otomatis menggunakan Otsu)
    _, thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
    # 4. Noise Removal (Menggunakan operasi morfologi Opening)
    kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3, 3))
    clean_img = cv2.morphologyEx(thresh, cv2.MORPH_OPEN, kernel)
  
```

```
# 5. Character Recognition (OCR)
# Inisialisasi EasyOCR reader (Bahasa Inggris/Alfanumerik)
reader = easyocr.Reader(['en'], gpu=False)
# Membaca teks dari gambar yang sudah dibersihkan
results = reader.readtext(clean_img)
# 6. Teks Hasil OCR
ocr_text = " ".join([text for _, text, prob in results if prob > 0.3])
return clean_img, ocr_text
```

Pengolahan Hasil OCR

Hasil OCR tidak selalu langsung dapat digunakan karena kemungkinan terdapat kesalahan pembacaan karakter. Oleh karena itu, sistem dirancang melakukan normalisasi terhadap hasil OCR, Tahapan pengolahan hasil OCR meliputi:

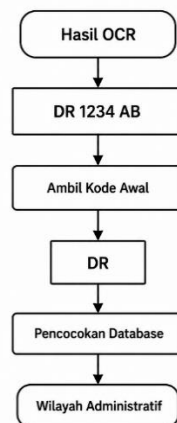
- Menghilangkan spasi yang tidak diperlukan.
- Mengubah seluruh karakter menjadi huruf kapital.
- Menghilangkan karakter khusus.
- Memisahkan kode wilayah dengan nomor kendaraan.
- Melakukan pencocokan terhadap database kode wilayah.

Tabel 2 normalisasi

Hasil OCR	Setelah Normalisasi
dr 1234 ab	DR1234AB
DR-1234-AB	DR1234AB
Dr 1234 Ab	DR1234AB

Klasifikasi Wilayah Administratif

Setelah nomor kendaraan berhasil dibaca, sistem melakukan klasifikasi berdasarkan kode huruf pada bagian awal pelat. Proses klasifikasi dirancang menggunakan pencocokan (*matching*) antara hasil OCR dengan database kode wilayah seperti yang dijelaskan dalam gambar 5 dibawah ini .



Gambar 5 Matching Hasil OCR dengan kode Wilayah

Implementasi Kode Python

```
import re
# Database Mapping Kode Pelat Nomor ke Wilayah Administratif
DATABASE_WILAYAH = {
    "A,B,C,D,E": "kota mataram",
    "H,J,K" : "Lombok barat",
    "S" : "Lombok Tengah",
    "L" : "Lombok Timur",
    "f,G" : "Lombok Utara",
}
```

```
def klasifikasi_wilayah(teks_ocr):
    # 1. Normalisasi teks OCR (ubah ke huruf kapital & hapus spasi berlebih)
    teks_clean = teks_ocr.strip().upper()
    # 2. Ambil Kode Awal (Huruf di bagian depan sebelum angka)
    match = re.match(r"^[A-Z]{1,2}", teks_clean)
    if not match:
        return None, "Kode pelat tidak valid"
    kode_awal = match.group(1)
    # 3. Pencocokan dengan Database Wilayah
    wilayah = DATABASE_WILAYAH.get(kode_awal, "Wilayah tidak ditemukan dalam database")
    return kode_awal, wilayah
```

Database Kode Wilayah

Database digunakan untuk menyimpan dan mengelola informasi mengenai kode pelat kendaraan bermotor dan wilayah administratif yang berkaitan. Database ini menjadi acuan sistem dalam melakukan proses klasifikasi setelah karakter pelat berhasil dikenali menggunakan OCR. Hasil pembacaan OCR akan dibandingkan dengan data kode pelat yang tersimpan untuk menentukan wilayah administratif kendaraan. Dalam perancangan sistem ini, database dibuat dalam bentuk tabel kode_wilayah dengan struktur sebagai berikut:

Tabel 3 struktur tabel database

ID	Kode	Wilayah	Keterangan
1	A,B,C,D,E	Mataram	Kode registrasi
2	H,J,K	Lombok Barat	Kode registrasi
3	S	Lombok tengah	Kode registrasi
4	L	Lombok Timur	Kode registrasi
5	F,G	Lombok Utara	Kode Registrasi

Database tersebut dapat dikembangkan dengan menambahkan informasi seperti:

- Kode wilayah.
- Nama kabupaten/kota.
- Provinsi.
- Jenis kendaraan.
- Keterangan kode.
- Data historis bila diperlukan.

Antarmuka Sistem Deteksi Pelat

Antarmuka sistem dirancang agar pengguna dapat memasukkan citra kendaraan dan memperoleh hasil deteksi secara sederhana, Komponen antarmuka yang dirancang meliputi:

Halaman utama seperti pada gambar 4.6 menampilkan menu untuk memilih gambar kendaraan.



Gambar 6 Halaman Utama

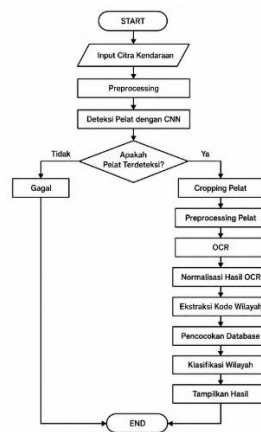
Setelah proses selesai pada gambar 7 , sistem menampilkan:

HASIL DETEKSI	
Nomor Pelat	: DR 1234 AB
Kode	: AB
Wilayah	: Lombok / Mataram
Status	: TERIDENTIFIKASI

Gambar 7 hasil deteksi

Alur Kerja Sistem dan Rancangan Sistem

Berdasarkan keseluruhan rancangan yang telah dibuat, alur kerja sistem deteksi pelat kendaraan bermotor dirumuskan untuk menggambarkan proses mulai dari **input citra kendaraan hingga diperoleh informasi nomor pelat dan wilayah administratif**. Setiap tahapan saling berhubungan, sehingga keluaran dari satu proses menjadi masukan bagi proses berikutnya seperti pada gambar 8 .



Gambar 8 alur kerja sistem

Rancangan Sistem

proses sistem dapat ditampilkan pada tabel berikut.

Tabel 4 poses sistem

Tahapan	Input	Proses	Output
1	Citra kendaraan	Input	Citra kendaraan
2	Citra kendaraan	Preprocessing	Citra terproses
3	Citra terproses	CNN	Area pelat
4	Area pelat	Cropping	Citra pelat
5	Citra pelat	Preprocessing OCR	Citra karakter
6	Citra karakter	OCR	Teks pelat
7	Teks pelat	Normalisasi	Teks bersih
8	Teks bersih	Ekstraksi kode	Kode wilayah
9	Kode wilayah	Database matching	Wilayah administratif

Pembahasan Hasil Perancangan Dan Indikator Keberhasilan Sistem

Berdasarkan hasil perancangan, sistem deteksi pelat kendaraan bermotor mengintegrasikan dua tahapan utama, yaitu **deteksi objek menggunakan CNN** dan **pengenalan karakter menggunakan OCR**. Integrasi kedua metode tersebut

memungkinkan sistem tidak hanya menemukan lokasi pelat pada citra kendaraan, tetapi juga memperoleh informasi teks yang terdapat pada pelat.

CNN berperan pada tahap pengenalan pola visual dan deteksi area pelat, sedangkan OCR berperan dalam menerjemahkan karakter pada pelat menjadi teks. Selanjutnya, hasil teks tersebut dibandingkan dengan database kode wilayah untuk memperoleh informasi administratif.

Pendekatan tersebut memiliki keuntungan karena proses deteksi objek dan pembacaan karakter dipisahkan menjadi beberapa tahapan. Dengan demikian, apabila terjadi kesalahan pada hasil OCR, sistem dapat melakukan preprocessing atau normalisasi sebelum menentukan wilayah administratif.

Namun, keberhasilan sistem sangat dipengaruhi oleh kualitas citra. Faktor seperti pencahayaan, sudut pengambilan gambar, jarak kamera, ukuran pelat, kondisi pelat, keberadaan objek yang menutupi pelat, serta kualitas karakter dapat memengaruhi hasil deteksi CNN dan pembacaan OCR.

Indikator Keberhasilan Sistem

Untuk mengetahui keberhasilan rancangan, beberapa indikator yang dapat digunakan dalam pengujian adalah:

Tabel 5 indikator keberhasilan

No.	Indikator	Parameter
1	Deteksi pelat	Pelat berhasil/tidak berhasil ditemukan
2	Akurasi OCR	Persentase karakter yang terbaca benar
3	Akurasi nomor pelat	Kesesuaian hasil OCR dengan pelat sebenarnya
4	Akurasi wilayah	Kesesuaian kode dengan wilayah administratif
5	Waktu proses	Waktu yang dibutuhkan sistem
6	Keberhasilan sistem	Persentase seluruh proses yang berhasil

Untuk menghitung akurasi klasifikasi wilayah dapat digunakan:

$$[\text{Accuracy} = \frac{\text{Jumlah prediksi benar}}{\text{Jumlah seluruh data uji}} \times 100\%]$$

Sedangkan akurasi pembacaan karakter dapat dihitung dengan:

$$[\text{Accuracy}_{\text{OCR}} = \frac{\text{Jumlah karakter benar}}{\text{Jumlah seluruh karakter}} \times 100\%]$$

Dengan indikator tersebut, rancangan sistem dapat diuji secara kuantitatif sehingga hasil penelitian tidak hanya berupa tampilan sistem, tetapi juga dapat menunjukkan tingkat keberhasilan metode yang digunakan.

KESIMPULAN

Hasil perancangan menghasilkan sebuah sistem yang terdiri atas proses input citra kendaraan, preprocessing, deteksi pelat menggunakan CNN, cropping area pelat, preprocessing citra pelat, pembacaan karakter menggunakan OCR, normalisasi hasil OCR, serta klasifikasi wilayah administratif berdasarkan kode pelat.

Sistem yang dirancang diharapkan mampu membantu proses identifikasi kendaraan berdasarkan nomor pelat secara otomatis. Hasil akhir sistem berupa nomor pelat kendaraan dan informasi wilayah administratif yang sesuai dengan kode yang berhasil dikenali.

Dari hasil tulisan yang disajikan penulis memberikan saran agar rancangan sistem deteksi pelat nomor dan klasifikasi wilayah ini dapat diimplementasikan secara optimal

secara nyata (Real-Time) dan pada lingkungan produksi (real-world deployment), beberapa langkah teknis yang direkomendasikan adalah:

1. Implementasi Perangkat Keras dan Kamera (Edge Deployment):

Spesifikasi Kamera: Menggunakan kamera tipe Automatic Number Plate Recognition (ANPR) atau kamera IP bernilai FPS tinggi (minimal 60 FPS) yang dilengkapi dengan Infrared (IR) dan fitur Wide Dynamic Range (WDR) untuk mengantisipasi silau lampu kendaraan dan kondisi gelap malam hari.

Perangkat Edge: Mempertimbangkan penggelaran model pada perangkat edge computing seperti NVIDIA Jetson (Nano/Xavier) agar proses inferensi deteksi CNN dan OCR dapat berjalan langsung di lokasi (on-premise) tanpa ketergantungan penuh pada jaringan internet.

2. Optimasi Algoritma dan Model Deep Learning:

Penggunaan Model CNN Ringan: Mengonversi model CNN ke format TensorRT atau ONNX Runtime guna mempercepat inference time sehingga sistem dapat membaca pelat kendaraan yang melaju dengan kecepatan tinggi secara real-time.

Custom OCR Font Training: Melakukan fine-tuning dataset OCR khusus menggunakan font standar Korlantas Polri untuk mengatasi kesalahan pembacaan angka "1" dengan huruf "I", atau angka "0" dengan huruf "O".

3. Pengembangan Fitur dan Arsitektur Sistem:

Integrasi Basis Data & API: Menghubungkan modul OCR dengan Database Management System (DBMS) berbasis Cloud atau API Samsat/E-TLE untuk menyajikan data jenis kendaraan, status pajak, dan masa berlaku pelat.

Antarmuka Pengguna (UI/UX) Real-Time: Membangun Dashboard berbasis Web (menggunakan React/Vue dan FastAPI) yang dilengkapi fitur visualisasi statistik asal wilayah kendaraan.

DAFTAR PUSTAKA

- Akbar, R. A., Telkom, Y. P., Sidoarjo, S. M. K. T., Keahlian, P., Perangkat, P., & Dan, L. (2024). Pemanfaatan Aplikasi Visual Studio Code Dalam Pembuatan Web Di Era Modern. 1–13.
- Akhmad, N., Romadhon, G., Usman, K., & Patmasari, R. (2020). Deteksi Pelat Nomor Kendaraan dengan Hough Transform dan Harris Corner Menggunakan Akuisisi Melalui Raspberry Pi Detection of Vehicle Pelate with Hough Transform and Harris Corner using Acquisition through Raspberry Pi. *Telka*, 6(2), 93–103.
- Burta, F. S. (2018). No (1), 430–439.
- Daya, B., Asmara, Y., Wulaningrum, R., & Helilintar, R. (2023). Implementasi Region of Interest (ROI) Untuk Segmentasi Citra Tanda Tangan. *Agustus*, 7, 2549–7952.
- Harani, N. H., Prianto, C., & Hasanah, M. (2019). Deteksi Objek Dan Pengenalan Karakter Pelat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python. *Jurnal Teknik Informatika*, 11(3), 47–53.
- I Wayan Suartika E. P., Arya Yudhi Wijaya, & Rully Soelaiman. (2021). Convolutional Neural Network Konsep, Penerapan, dan Implementasi Dengan Contoh Eksperimen. *Jurnal Teknik Its*, 5, A65–A69. <https://ejurnal.its.ac.id/index.php/teknik/article/viewFile/15696/2553>
- Imantaka Firswandy, R., Rahayu, S., & Achmad, A. (2019). Deteksi Pelat Nomor Kendaraan Bermotor Menggunakan Sliding Concentric Windows (SCW) Untuk Aplikasi Sistem Transportasi Cerdas. In *Institut Teknologi Sepuluh Nopember*.
- Izzuddin, F. N. (2022). Konsep Smart City Dalam Pembangunan Berkelanjutan. *Citizen : Jurnal Ilmiah Multidisiplin Indonesia*, 2(3), 376–382. <https://doi.org/10.53866/jimi.v2i3.96>
- Kharisma, O. B. (2021). Sistem Identifikasi Pelat Nomor Kendaraan Dalam Penerapan Regulasi Pajak Berbasis Citra Digital. *JST (Jurnal Sains Dan Teknologi)*, 10(1), 117–127. <https://doi.org/10.23887/jstundiksha.v10i1.32975>
- Oeliga, C. (2021). Pemalsuan Tanda Nomor Kendaraan. *Datin Law*, 1 No 2(Penegakan Hukum

- Terhadap Tindak Pidana Pemalsuan Tanda Nomor Kendaraan Bermotor (Tnkb)), 57–65.
<https://Ojs.Umb-Bungo.Ac.Id/Index.Php/DATIN/Article/View/565/559>
- Putra Ardinata, R., Rahmat, H. K., Andres, F. S., Waryono, W., Studi, P., Perhotelan, M., Pariwisata, F., & Perhotelan, D. (2022). Transformational leadership as a solution for the development of the smart city concept in the society era: a literature review. *Multidisciplinary Journal of Counseling and Social Research*, 1(1), 2962–8350.
- Rahman, A. (2020). Rapid Application Development Sistem Pembelajaran Daring Berbasis Android. *Intech*, 1(2), 20–25. <https://doi.org/10.54895/intech.v1i2.639>
- Reswan, Y., Raffles, R., Wijaya, A., & Apridiansyah, Y. (2024). Penerapan Algoritma Ocr Untuk Ekstraksi Informasi Dari Citra Kartu Tanda Mahasiswa (Ktm). *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(5), 11004–11009. <https://doi.org/10.36040/jati.v8i5.11006>
- Rifka Alkhilyatul Ma'rifat, I Made Suraharta, I. I. J. (2024). No Title 済無No Title No Title No Title. 2, 306–312.
- Sri Hartati. (2020). Perancangan Sistem Informasi Inventaris Barang Pada Kantor Notaris Dan Ppat Ra Lia Kholila, Sh Menggunakan Visual Studio Code. *Siskomti*, 2(2), 37–48.
- Zakiyamani, M., Cahyani, T. I., Riana, D., Hardianti, S., & Naren, B. (2022). Menggunakan Opencv Dan Deep Learning Berbasis Python Detection And Recognition Of Vehicle Number Character Pelate Using Python-Based Opencv And Deep Learning Program Studi Magister Ilmu Komputer , Universitas Nusa Mandiri Pendahuluan Sistem Pengenalan Citr. 5, 56–64.