

# RANCANG BANGUN SISTEM DETEKSI PLAT NOMOR KENDARAAN BERBASIS *COMPUTER VISION* DENGAN METODE *OPTICAL CHARACTER RECOGNITION*

Syamsul Falah Annur, Rais, Hepatika Zidny Ilmadina

Email: [syamsulfalahannur23@gmail.com](mailto:syamsulfalahannur23@gmail.com)

DIII Teknik Komputer Politeknik Harapan Bersama

Jln. Mataram No. 09 Tegal

Telp/Fax (0283) 352000

## ABSTRAK

Tingginya jumlah kecelakaan lalu lintas selalu menyita perhatian masyarakat. Salah satu penyebab terjadinya kecelakaan lalu lintas adalah tindakan melanggar aturan lalu lintas seperti menerobos *traffic light*, tidak mengurangi kecepatan pada saat lampu merah, berhenti dimarka jalan dan mengabaikan rambu-rambu jalan. Tanda Nomor Kendaraan Bermotor atau TNKB merupakan identitas atau kode unik yang menjadi pengenalan sebuah kendaraan bermotor. *LPR* (Licence Plate Recognition) dan *OCR* (Optical Character Recognition) merupakan salah satu dari beberapa kemajuan teknologi yang kini berkembang pesat. Dari permasalahan tersebut, maka diperlukan solusi untuk dapat membantu pihak kepolisian dalam memonitoring dan memantau pelanggaran yang terjadi di jalan raya. Sistem ini dibuat berdasarkan pengenalan plat nomor kendaraan berbasis *Computer Vision*. Dari hasil pengujian didapatkan bahwa *Optical Character Recognition* mampu mendeteksi plat nomor kendaraan pada jalan raya berdasarkan sisa jarak yang terbaca. Keakuratan pembacaan *Optical Character Recognition* yaitu sebesar 53,04%.

Kata Kunci: *Traffic Light*, *License Plate Recognition*, *Optical Character Recognition*, *Computer Vision*.

## I. PENDAHULUAN

Tingginya jumlah kecelakaan lalu lintas selalu menyita perhatian masyarakat. Sebab, kecelakaan lalu lintas terus merenggut korban jiwa serta menyebabkan kerugian finansial dan dapat memengaruhi psikologis korban. Menurut data POLRI, terdapat 107.500 kecelakaan lalu lintas sepanjang tahun 2019. Angka ini merupakan peningkatan 3% dari 2018, yakni sebanyak 103.672 kecelakaan[1].

Salah satu penyebab terjadinya kecelakaan lalu lintas adalah tindakan melanggar aturan lalu lintas seperti menerobos *traffic light*, tidak mengurangi kecepatan pada saat lampu merah, berhenti dimarka jalan dan mengabaikan rambu-rambu jalan. Aturan mengenai batas kecepatan di Indonesia telah diatur dalam Undang Undang Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan (LLAJ). Dalam undang-undang ini menyatakan setiap orang yang mengendarai kendaraan bermotor di jalan wajib mematuhi ketentuan kecepatan maksimal dan minimal. Rambu-rambu serta peringatan batas kecepatan pada

zona jalan tertentu sudah banyak dipasang. Namun belum terdapat indikator yang dapat dijadikan acuan terhadap pelanggaran lalu lintas jenis tersebut, sehingga kurang efektif penggunaannya[2].

Tanda Nomor Kendaraan Bermotor atau TNKB merupakan identitas atau kode unik yang menjadi pengenalan sebuah kendaraan bermotor. TNKB sebuah kendaraan terdiri dari baris yang pertama terdiri dari huruf yang menunjukkan kode wilayah, biasanya merupakan area Karesidenan sebuah wilayah administratif yang terdiri dari beberapa kabupaten, Kemudian diikuti dengan nomor bagian kedua berupa angka yang menunjukkan nomor polisi kendaraan dan bagian ketiga adalah huruf yang menunjukkan kabupaten domisili pemilik kendaraan[3].

*LPR* (Licence Plate Recognition) dan *OCR* (Optical Character Recognition) merupakan salah satu dari beberapa kemajuan teknologi yang kini berkembang pesat. *LPR* (Licence Plate Recognition) atau yang disebut dengan pendeteksi plat nomor kendaraan

merupakan aplikasi yang telah banyak dikembangkan di luar negeri yang telah berhasil diimplementasikan untuk sistem parkir, sistem tol, sistem lalu lintas, dan lain sebagainya. *OCR* (Optical Character Recognition) merupakan aplikasi yang bisa mengolah gambar (*image*) menjadi teks, aplikasi ini dapat memanipulasi *image* yang bertulisan tangan, tulisan mesin ketik atau *computer text*[4].

Dari latar belakang tersebut, sehingga akan dikembangkan sistem deteksi plat nomor kendaraan dengan metode *OCR* (Optical Character Recognition) yang dapat membantu pihak kepolisian dalam memonitoring dan memantau pelanggaran yang terjadi di jalan raya dengan mendeteksi plat nomor kendaraannya.

## II. METODE PENELITIAN

### 1. Observasi

Metode pengumpulan data melalui pengamatan yang meliputi lokasi pada objek terkait untuk mengumpulkan data yang diperlukan untuk pembuatan produk. Kegiatan observasi dilakukan di POLRES Kota Tegal dan mendapatkan hasil jumlah pelanggaran lalu lintas yang terjadi pada tahun 2019 dan 2020. Pada tahun 2019 jumlah pelanggaran lalu lintas sebanyak 35.511, dan pada tahun 2020 jumlah pelanggaran lalu lintas sebanyak 18.509. Selain itu didapatkan hasil kejadian kecelakaan lalu lintas pada tahun 2019 dan 2020. Pada tahun 2019 jumlah kecelakaan lalu lintas sebanyak 293 kasus, sedangkan pada tahun 2020 jumlah kecelakaan lalu lintas sebanyak 228 kasus.

### 2. Wawancara

Selain observasi, Dalam penelitian ini pengumpulan data menggunakan metode wawancara, yaitu mendapatkan keterangan yang berhubungan dengan pembuatan sistem deteksi plat nomor kendaraan berbasis *Computer Vision* menggunakan metode *Optical Character Recognition* .

Dalam penelitian ini wawancara dilakukan dengan petugas polres terkait.

### 3. Studi Literatur

Pada proses peneyelesain ini, pengumpulan referensi yang diambil dari berbagai *literatur* yang berkaitan dengan judul penelitian yaitu Perpustakaan, Jurnal, *E-Book* dan Laporan Penelitian. Setelah data penelitian terkumpul, maka perlu adanya proses pemilihan data kemudian dianalisis sehingga diperoleh suatu kesimpulan yang objektif dari suatu penelitian.

## III. HASIL DAN PEMBAHASAN

### 1. Analisa Permasalahan

Permasalahan pelanggaran lalu lintas yang terjadi dapat menyebabkan dampak yang cukup kompleks dalam kehidupan masyarakat. Salah satunya terjadinya kecelakaan lalu lintas yang dapat menyebabkan kerugian material ataupun ancaman terhadap keselamatan masyarakat.

*LPR* (Licence Plate Recognition) dan *OCR* (Optical Character Recognition) merupakan salah satu dari beberapa kemajuan teknologi yang kini berkembang pesat. *LPR* atau yang disebut dengan pendeteksi plat nomor kendaraan merupakan aplikasi yang telah banyak dikembangkan di luar negeri yang telah berhasil diimplementasikan untuk sistem parkir, sistem tol, sistem lalu lintas, dan lain sebagainya. *OCR* merupakan aplikasi yang bisa mengolah gambar (*image*) menjadi teks , aplikasi ini dapat memanipulasi *image* yang bertulisan tangan, tulisan mesin ketik atau *computer text*.

Penggunaan *Computer Vision* sudah banyak di terapkan di berbagai industri di era 4.0, karena penerapan *Computer Vision* memiliki tingkat akurasi lebih baik, salah satunya penerapan dalam

deteksi plat nomor kendaraan menggunakan *Computer Vision*.

Penelitian tentang sistem deteksi plat nomor kendaraan sudah banyak dilakukan. Namun, kebanyakan *dataset* yang tersedia untuk deteksi plat nomor menggunakan *format* negara lain yang menjadikan sulitnya untuk mendeteksi *format* plat nomor indonesia.

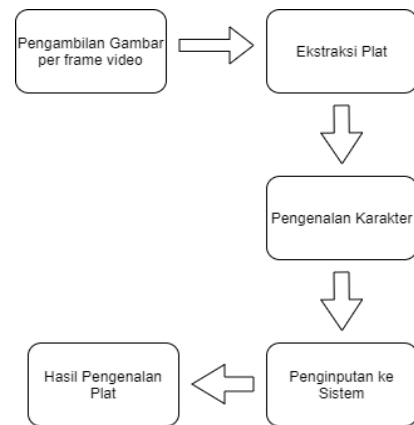
Sehingga diusulkan metode yang tepat untuk mendeteksi plat nomor kendaraan indonesia adalah dengan metode *OCR* (Optical Character Recognition) dan menggunakan algoritma *template matching*. Algoritma *template matching* merupakan salah satu algoritma yang efektif untuk diterapkan dalam sistem *OCR* dan juga kerap kali digunakan sebagai *object detection* dalam sebuah *video* maupun gambar. Algoritma ini banyak dipilih karena penerapannya yang sangat sederhana dan mudah..

## 2. Perancangan Sistem

Perancangan sistem ini dilakukan dengan perencanaan sistem, implementasi sistem, dan uji coba sistem. Untuk mempermudah dalam merancang dan membuat sistem deteksi plat nomor kendaraan berbasis *python* dengan metode *optical character recognition*, Maka dirancang sebuah *diagram* blok dan *flowchart*.

## 3. Diagram Blok

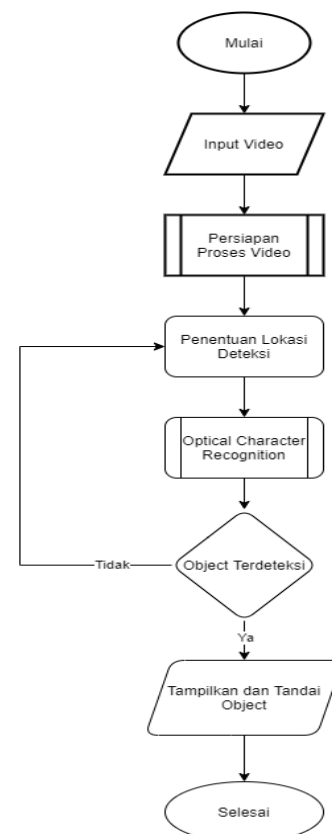
*Diagram* Blok digunakan untuk menggambarkan kegiatan yang ada di dalam sistem, agar dapat lebih memahami sistem yang akan dibuat maka perlu dibutuhkan gambaran tentang sistem yang berjalan. Berikut gambar *diagram* blok sistem dalam proses deteksi plat nomor kendaraan pada penelitian ini seperti dalam Gambar



Gambar 1 Diagram Blok Sistem

## 4. Flowchart

*Flowchart* adalah bagian alur yang menggambarkan tentang urutan langkah jalannya suatu *program* dalam sebuah bagan dengan simbol-simbol bagan yang sudah ditentukan seperti pada gambar



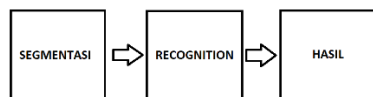
Gambar 2 Flowchart

## 5. Implementasi Sistem

Setelah melakukan analisis permasalahan dan telah dibuatnya sebuah sistem yang dapat menjawab permasalahan yang ada, maka tahap selanjutnya adalah implementasi sistem. Pada tahap ini peneliti menerapkan metode *Optical Character Recognition* pada sistem deteksi plat nomor kendaraan.

### a. Implementasi *optical character recognition*

Pada tahap *OCR* ini proses pengenalan karakter, yang akan dijelaskan pada gambar



Gambar 3 Tahap *OCR*

Langkah pertama adalah proses segmentasi, yang bertujuan untuk memisahkan wilayah objek dengan latar belakang agar objek mudah dianalisis, sehingga citra sebagian besar adalah segmentasi masing-masing karakter. Langkah selanjutnya adalah *Recognition*, merupakan proses untuk mengenali karakter yang diamati dengan cara membandingkan ciri-ciri karakter yang diperoleh dengan ciri-ciri karakter yang ada pada basis data. Setelah itu maka akan keluar hasil dari deteksinya.

### b. *source code* :

```
import cv2, pytesseract
as pt

pt.pytesseract.tesseract
_cmd = r"C:\Program
Files\Tesseract-
OCR\tesseract.exe"
#cfg="--psm 7 oem 3 -c
tessedit_char_whitelist=
ABCDEFGHJKLMNOPQRSTUWX
YZ0123456789"
```

```
#cfg="--psm 8 oem 1 -c
tessedit_char_whitelist=A
BCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789"
```

```
cfg="--psm 8 oem 3 -c
tessedit_char_whitelist=A
BCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789"
```

```
font=cv2.FONT_HERSHEY_SIM
PLEX
```

```
color = [int(c) for c in
COLORS[classIds[i]]]
```

```
cv2.imwrite('plate.jpg',f
rm[y:y+h,x:x+w])
```

```
roi=cv2.cvtColor(~frm[y+5
:y+h,x:x+w],cv2.COLOR_BGR
2GRAY)
```

```
roi=cv2.resize(roi,
dsize=None, fx=rasio,
fy=rasio)
```

```
cv2.putText(frm,
("+str(w)+" "+str(h)+" "
, (x, y - 5),
cv2.FONT_HERSHEY_SIMPLEX,
0.5, (255, 0, 0), 2)
cv2.rectangle(frm, (x,
y), (x + w, y + h),
(0,0,255), 2)
```

```
blur
=cv2.GaussianBlur(roi, (5,
5),0)
roi = cv2.threshold(blur,
200, 255, cv2.THRESH_OTSU
| cv2.THRESH_BINARY)[1]
```

```
cv2.imshow('img',roi)
```

```
text=pt.image_to_string(r
oi,lang=None,config=cfg)
```

```
if len(text)>=5:
```

```
cv2.putText(frm, text,
(x, y -
5),cv2.FONT_HERSHEY_SIMPL
EX, 0.5, (255, 0, 0), 2)
print(text)
```

## 6. Hasil Pengujian

### a. Hasil pada sistem

Berikut merupakan hasil yang akan tampil pada sistem

apabila mendeteksi plat nomor kendaraan.



Gambar 4 Hasil Pada sistem

b. Hasil pengujian dengan video

Tabel 1 Hasil Pengujian dengan video

No	Plat terdeteksi	Karakter yang dikenali	Akurasi (%)
1.		D 1627 ace	75 %
2.		H 1332	57 %
3.		H 1042 c1	71 %
4.		o7e7 az	57 %
5.		E 1852 ac	71 %
6.		E S198 BZ	85,7 %
7.		£72	28 %
8.		Tidak keluar output	0%

No	Plat terdeteksi	Karakter yang dikenali	Akurasi (%)
		text	
9.		Tidak keluar output text	0%
10.		E 1863 CH	85,7 %
Rata-rata akurasi			53,04%

Dari hasil pengujian didapatkan bahwa *Optical Character Recognition* mampu mendeteksi plat nomor kendaraan pada jalan raya berdasarkan sisa jarak yang terbaca. Keakuratan pembacaan *Optical Character Recognition* yaitu sebesar 53,04%.

c. Hasil Pengujian Secara *Realtime*

Tabel 2 Hasil Pengujian *realtime*

No	Plat terdeteksi	Karakter yang dikenali	Akurasi
1		Tidak keluar Output Text	0%

Dari hasil pengujian secara *realtime* didapatkan data plat kendaraan yang terdeteksi, namun untuk hasil *output* karakter yang dikenali tidak keluar dikarenakan selain pengaruh dari pencahayaan yang kurang adapun dari *noise*

*video* yang banyak pada daerah plat nomor dapat menyebabkan sistem sulit untuk membaca karakter dari plat tersebut.

#### IV. KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan maka dapat diambil beberapa kesimpulan yang diambil dari hasil pengujian, sistem deteksi plat nomor kendaraan berbasis *Computer Vision* menghasilkan akurasi sebesar 53,04%. Implementasi *Computer Vision* pada sistem deteksi plat nomor kendaraan dapat dikembangkan dengan metode *Optical Character Recognition*.

#### V. DAFTAR PUSTAKA

- [1] D. Ariyoga, "Penelitian Terkini Tentang Sistem Pendeteksi Pelanggaran Lalu Lintas Berbasis Deep Learning: Sebuah Kajian Pustaka."
- [2] N. Kuswandi and F. A. Rakhmadi, "Prototipe Sistem Pengukuran Laju Kendaraan Bermotor Sebagai Upaya Pengawasan Terhadap Pelanggaran Rambu-Rambu Lalu Lintas," *Integr. Lab J.*, vol. 5, no. 1, pp. 35–44, 2017.
- [3] A. Budiarto, T. B. Adji, and R. Hartanto, "Deteksi Nomor Kendaraan Dengan Metode Connected Component Dan Svm," *J. Teknol. Inf. Magister Darmajaya*, vol. 1, no. 01, pp. 106–117, 2015.
- [4] K. Budiarta, I. M. D. Susila, and K. Ariasa, "Aplikasi Tilang dengan Pengenalan Plat Nomor Kendaraan Dan Pelaku Pada Platform Mobile," *TKSS Bali*, pp. 1–12, 2015.
- [5] M. A. Lamanele, D. P. P. Siwi, M. R. F. Gugutu, and W. S. Pambudi, "Software Pendeteksi Plat Nomor Kendaraan Bermotor Untuk Pelanggaran Traffic Light Dengan Menggunakan Kamera," pp. 315–320, 2018.
- [6] S. Aulia, P. Maria, and R. Ramiati, "Aplikasi Pendeteksi Plat Nomor Kendaraan Berbasis Raspberry Pi Menggunakan Website Untuk Pelanggaran Lalu Lintas," *Elektron J. Ilm.*, vol. 11, no. 2, pp. 84–89, 2019, doi: 10.30630/eji.11.2.126.
- [7] K. Kusumawati and D. W. Cahyadi, "Penerapan Teknologi Optical Character Recognition Untuk Mendeteksi Plat Nomor Kendaraan," *Pros. Semin. Nas. Inov. Teknol.*, pp. 12–20, 2017.
- [8] S. W. Utama and A. Kusumawardhani, "Aplikasi Pendeteksi Plat Nomor Negara Indonesia Menggunakan OpenCV dan Tesseract OCR pada Android Studio," no. December, pp. 1–6, 2018.
- [9] N. H. Harani, C. Prianto, and M. Hasanah, "Deteksi Objek Dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network ( CNN ) Berbasis Python," *J. Tek. Inform.*, vol. 11, no. 3, pp. 47–53, 2019.
- [10] M. P. Putri and H. Effendi, "Implementasi Metode Rapid Application Development Pada Website Service Guide 'Waterfall Tour South Sumatera,'" *J. SISFOKOM*, vol. 07, no. September, pp. 130–136, 2018.