

RANCANG BANGUN *PROTOTYPE* SISTEM KEAMANAN PARKIR KENDARAAN MENGGUNAKAN KAMERA DENGAN METODE *EIGENFACE* DAN *HEARD CASCADE*

Zulfikar

Jurusan Teknik Otomasi Manufaktur dan Mekatronika
POLMAN Bandung
Jl. Kanayakan 21, Bandung
Zulfikar1aec@gmail.com

Fitria Suryatini, Suharyadi Pancono

Jurusan Teknik Otomasi Manufaktur dan Mekatronika
POLMAN Bandung
Jl. Kanayakan 21, Bandung

Abstrak— Smart parking adalah tempat parkir yang membantu pengemudi untuk memarkirkan kendaraan. Teknologi semakin berkembang, perkembangan teknologi salah satunya yang dapat dilihat adalah transportasi. Perkembangan transportasi khususnya kendaraan beroda empat (mobil) semakin mempengaruhi jumlah kendaraan yang dikeluarkan. Jumlah kendaraan yang semakin banyak mempengaruhi kelancaran lalu lintas karena pengemudi yang sedang mencari tempat parkir. Saat pengemudi menemukan tempat parkir, pengemudi masih harus berputar-putar untuk mencari tempat parkir yang masih tersedia. Untuk mengatasi permasalahan tersebut, pada tugas akhir ini dibuat sistem yang dapat mengatasi permasalahan yang dihadapi. Sistem yang akan dibuat bertujuan untuk memberikan informasi kepada pengemudi tempat parkir yang masih tersedia dan menjamin keamanan tempat parkir. Dari hasil pengujian yang dilakukan, proses pembacaan wajah dan pembacaan plat kendaraan telah berhasil. Hasil pengerjaan alat yang telah dibuat, mendapatkan hasil bahwa pada pembacaan karakter tulisan atau plat kendaraan mendapatkan hasil yang lebih baik yang mencapai tingkat keberhasilan 100% dengan menggunakan metode Optical Character Recognition (OCR) dan menambahkan metode pengolahan citra. Sedangkan pada pembacaan wajah dari hasil pengujian yang telah dilakukan menunjukkan bahwa sensor kamera dapat mendeteksi wajah dengan intensitas cahaya pada pagi hari 20-35 lux, siang 26-50 lux, dan malam hari 2-18 lux.

Kata kunci : OCR, smart parking

I. PENDAHULUAN

Di zaman modern ini, perkembangan teknologi semakin berkembang. Perkembangan teknologi salah satunya yang dapat dilihat adalah transportasi. Perkembangan transportasi khususnya kendaraan beroda empat (mobil) semakin mempengaruhi jumlah kendaraan yang dikeluarkan. Jumlah kendaraan yang semakin banyak mempengaruhi kelancaran lalu lintas karena pengemudi yang sedang mencari tempat parkir. Tempat parkir sangat dibutuhkan ditempat-tempat umum ataupun fasilitas publik seperti kantor, kampus, dan pusat-pusat pembelanjaan [1]. Sebuah survei menunjukkan bahwa pada jam-jam sibuk di kota besar, menunjukkan 40% kemacetan terjadi karena banyaknya kendaraan yang berada di jalanan, kemacetan tersebut terjadi karena pengemudi yang mencari-cari tempat parkir [2].

Susahnya menemukan tempat parkir membuat pengemudi yang kecewa karena tidak menemukan tempat parkir. Saat pengemudi menemukan gedung parkir, pengemudi masih harus berputar-putar untuk mencari tempat parkir yang masih tersedia. Hal ini dikarenakan kurangnya informasi yang diberikan gedung parkir mengenai tempat parkir yang masih kosong, dari kurangnya informasi membuat proses pemarkiran kendaraan kurang efektif, baik dari sisi waktu ataupun dari sisi jarak tempuh kendaraan untuk mencapai tempat parkir [3]. Keadaan diatas, dimanfaatkan oleh banyak pihak khususnya masyarakat atau pengusaha untuk menyediakan tempat parkir. Salah satu contoh tempat parkir yang telah dibuat adalah *Smart Tower* yang dikembangkan oleh *Nussbaum Group* yang berfungsi untuk menyimpan dan menghadirkan *smart forfour*

vehicle, dipasang di Jerman dan Perancis. Dengan adanya *Smart Tower*, pemanfaatan lahan semakin lebih efisien. Selain *Smart Tower*, beberapa pusat perbelanjaan di kota besar di Indonesia sudah menyediakan sistem parkir yang memberikan informasi kepada pengendara mengenai jumlah tempat parkir yang tersedia atau tempat parkir sudah penuh dengan diberikan *display* pada pintu masuk area parkir [4]. Hal ini menghindari pengendara memasuki area parkir yang sudah penuh. Walau demikian sistem parkir ini masih memiliki kekurangan yaitu pengendara tidak mengetahui secara spesifik dimana akan parkir, menyebabkan pengendara harus berputar mencari tempat parkir yang kosong sehingga memakan waktu dan tidak efisien.

Berdasarkan paparan diatas tugas akhir yang akan dilakukan atau dikerjakan berkaitan dengan pembuatan sistem yang memberikan informasi mengenai lokasi parkir yang masih tersedia pada gedung parkir dan memberikan keamanan kendaraan. Pada tahun sebelumnya pembuatan sistem tersebut telah dilakukan oleh banyak pihak atau lebih dikenal dengan *smart parking*. Menurut Indah P Perdana (2018:1), “*Smart Parking* adalah bagian dari internet yang dimana menggunakan sensor yang berkomunikasi dengan menggunakan *remote* melalui internet dan berbagi informasi dengan menggunakan *protocol* komunikasi yang telah ditetapkan. *Smart parking* merupakan sistem parkir otomatis yang berfokus pada *monitoring* dan keamanan akses parkir”.

II. METODE DAN PERANCANGAN SISTEM

Pada gambaran umum ini, sistem akan melakukan dua tugas yang pertama untuk sistem *monitoring* yang ditunjukkan untuk dapat *monitoring* parkir yang masih tersedia dengan jaringan lokal, dan yang kedua yaitu sistem keamanan parkir. Pada sistem keamanan parkir, yang diperlukan yaitu sken wajah dan pendaftaran plat kendaraan secara manual. Hal ini digunakan untuk melakukan pendaftaran kendaraan dan pemilik kendaraan. Data yang telah diambil kemudian akan disimpan di dalam *database* yang nantinya akan digunakan untuk keamanan saat pengendara hendak masuk dan keluar dari tempat parkir. Kendaraan dan pengemudi yang sesudahnya

telah melakukan pendaftaran/registrasi bisa dengan cepat memasuki tempat parkir dengan melakukan pengecekan kendaraan dan pengecekan pemilik kendaraan tanpa harus melakukan pendaftaran/registrasi terlebih dahulu, Pengemudi juga dapat melakukan Pengecekan dengan menggunakan android untuk memeriksa tempat parkir yang masih tersedia sebelum memasuki tempat parkir yang disediakan.



Gambar 1. Gambaran Umum Sistem

A. Penerapan Sistem

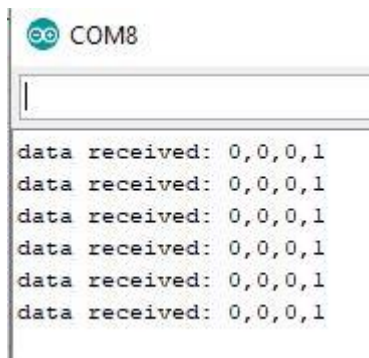
Pada sistem ini terdapat empat langkah kerja yaitu klarifikasi tugas, sistem mekanik, proses registrasi wajah, dan proses pendaftaran plat kendaraan. Sehingga membantu dalam proses pembuatan dan dapat digunakan.

B. Tahapan Program

1. program Web

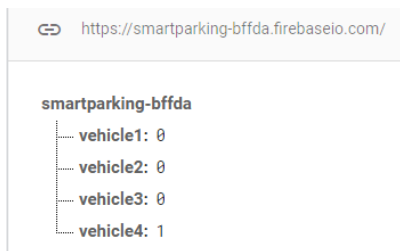
Pada tahapan ini dimulai dari pembacaan sensor *infrared*. Saat sensor *infrared* mendeteksi ada kendaraan maka *arduino wemos* akan mengirim variabel 1 pada *firebase* dan apabila sensor *infrared* tidak mendeteksi adanya kendaraan maka *arduiiono wemos* akan mengirim variabel 0 pada *firebase*, contoh variabel yang dikirim 0 0 0 1 yang dapat dilihat pada gambar 3. 6. Variabel 0 0 0 1 dapat berubah-ubah tergantung sensor yang terdeteksi, pada tugas akhir ini, sensor yang digunakan berjumlah 4 karena tempat parkir yang disediakan ada 4. Setelah *arduino wemos* mengirim variabel hasil pembacaan sensor pada *firebase*, kita dapat melihat perubahan tersebut pada gambar 3. 7. Apabilah ada perubahan nilai pada *firebase*,

maka akan ada perubahan pada tampilan disain *monitoring* yang telah dibuat menggunakan PHP yang dapat dilihat pada gambar 3. 8, untuk melihat program lebih jelas dapat dilihat pada lampiran.



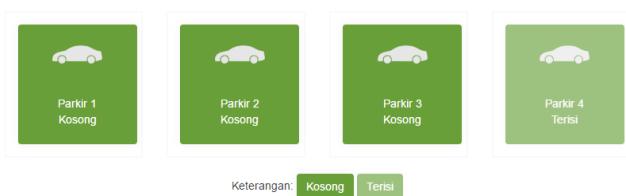
Gambar 2. Variabel Yang Dikirim

Pada Gambar 2 merupakan variabel nilai sensor yang terbaca pada *arduino*, nilai diatas dapat berubah-ubah tergantung sensor berapa terbaca pada *arduino*.



Gambar 3. Perubahan Pada *Firebase*

Pada gambar 3 adalah tampilan nilai sensor yang terbaca pada *firebase* yang akan dikirim melalui *arduino wemos* untuk merubah tampilan pada disain *monitoring* yang dapa dilihat pada gambar 4.



Gambar 4. Perubahan Pada Disain Monitoring

Pada gambar 4 merupakan tampilan *monitoring* yang dapat di cek melalui *website* yang sudah disediakan. Parkiran 4 berubah karena terjadi perubahan pada *firebase* yang dapat dilihat pada gambar 4, program yang digunakan untuk membuat *monitoring* dibuat menggunakan program bahasa PHP. Program PHP dapat dilihat pada lampiran. Berikut adalah diagram alir tahapan *web*.



Gambar 4. Tahapan Web

Pada tahapan ini, dimulai dari pembacaan sensor *infrared* dan apabila sensor mendeteksi adanya kendaraan maka sensor akan meneruskan ke arduino kemudian *arduino* akan mengelolah inputan yang diberikan oleh sensor *infrared* yang nantinya akan diteruskan ke *firebase* yang bertindak sebagai *database*. Kemudian apabila terjadi perubahan *database*, maka akan terjadi perubahan pada tampilan *monitoring* untuk memberitahukan apakah salah satu parkir terisi atau kosong.

2. Tahapan Program *Controller*

Pada tahapan ini dimulai dari pembacaan sensor kamera, sensor kamera akan mendeteksi wajah dan plat kendaraan pada *database* (Dapat dilihat pada lampiran program *firetest.py*). Apabilah salah satu tidak cocok atau kedua-duanya, maka pada program utama (Dapat dilihat pada lampiran program *zulfikar_TA.py*) akan mengirim perintah “ *kirim.gagal*” dan apabilah apabila sensor mendeteksi wajah dan plat kendaraan sesuai dengan *database* maka program utama akan mengirim perintah “ *kirim.sukses*” yang dapat dilihat pada potongan programnya pada gambar 3. 9, sedangkan pada serial monitornya akan terlihat perintah “ *terkunci*” apabilah salah satu atau kedua-duanya tidak ada pada *database* dan apabilah ada kedua-duanya pada *database* maka akan terlihat “ *terbuka*” pada serial monitor yang dapat dilihat pada gambar 3. 10. Pengecekan *database* melalui *firebase* yang dapat dilihat pada Gambar 3. 11 dan program pengecekannya *firebase* dapat dilihat pada lampiran “ *firetest.py*”.

```

print('id : ' + str(uid) + '\t nama : ' + nama + '\t plat nomer : ' + plat
if data_plate[uid] == plate and data_nama[uid] == nama:
    print('terbuka')
    kirim.sukses()
else:
    print('terkunci')
    kirim.gagal()

```

Gambar 5. Program perintah untuk mengirim ke servo

Pada gambar 3. 10 merupakan potongan program utama yang dapat dilihat pada lampiran (zulfekar_TA.py), program akan mengirim perintah sukses atau gagal pada *arduino* untuk menggerakkan motor servo. Sebelum dikirim ke *arduino*, perintah tersebut akan diolah pada program “*kirim.py*” untuk diubah dalam bentuk kode ASCII. Setelah diubah ke kode ASCII, barulah diteruskan ke *arduino* melalui kabel serial *arduino*.

```

id : 0  nama : unknown  plat nomer : mre
terkunci

```

- a. Tidak ada salah satu atau kedua-duanya pada *database*

```

id : 1  nama : Zulfekar      plat nomer : B 5974 TDM
terbuka

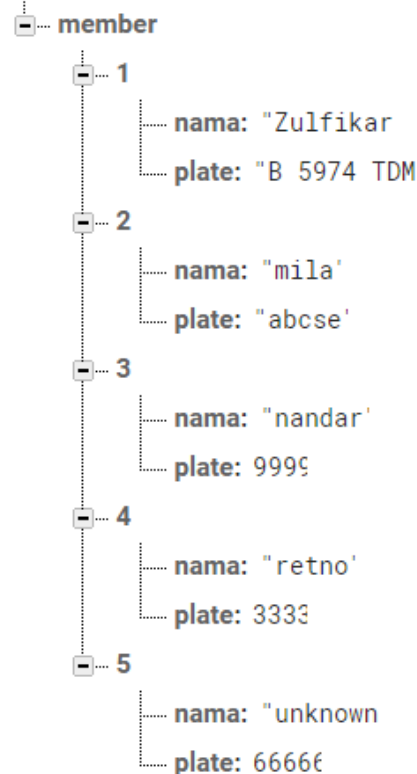
```

- a. Jika kedua-duanya ada pada *database*

Gambar 6. Perbandingan Pengecekan ID Pada Database

Pada gambar 6 menunjukkan hasil pengecekan ID yang berada pada *database*, apabila data cocok maka palang pintu akan terbuka dan apabila pengecekan pada *database* tidak ada maka palang pintu akan tertutup.

parking-zulfikar



Gambar 3. 1 Pengecekan Database di Firebase

Program utama (*zulfekar_TA.py*) akan memanggil program *import kirim* (Program lengkap dapat dilihat pada lampiran program *kirim.py*) untuk memproses perintah dari program utama, apabila pada program utama mengirimkan perintah “*kirim.sukses*” atau “*kirim.gagal*” maka akan diolah oleh program “*kirim.py*”, perintah program utama yang diterima akan diolah kembali untuk dikirimkan ke *arduino uno*. Program “*kirim.py*” akan mengirim kode ASCII pada *arduino*. Apabila program yang diterima adalah perintah “*kirim.sukses*”, maka akan dikirim cmd =’a’ yang diubah menjadi kode ASCII yang bernilai 97 untuk membuka palang pintu dan apabila menerima perintah “*kirim.gagal*”, maka akan diubah menjadi kode ASCII yang bernilai 98 untuk menutup palang pintu. Berikut potongan program “*kirim.py*” yang dapat dilihat pada

gambar 6, pada gambar 7 adalah potongan pada *arduino* untuk proses kendali palang pintu, dan pada gambar 3. 13 adalah program yang diterima pada *arduino uno* untuk kendali motor servo atau palang pintu.

```
def sukses():
    cmd='a'
    ser.write(cmd.encode('ascii')+'\r\n')
def gagal():
    cmd='b'
    ser.write(cmd.encode('ascii')+'\r\n')
```

Gambar 7. kirim.py

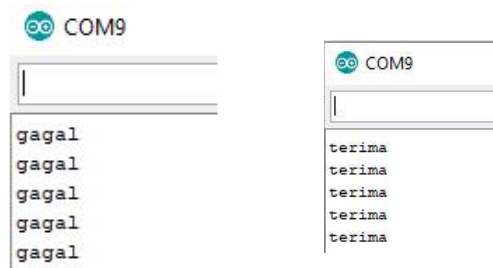
Pada gambar 6 adalah program yang akan dikirimkan ke *arduino* untuk menggerakkan motor servo atau pintu parkir, program diatas dibuat menggunakan python 2.7 dan dikirim menggunakan kaber serial ke *arduino*.

```
if (readString == 98)
{
    Serial.println("gagal");
    delay(4000);
    servo3();
    return;
}

if (readString == 97)
{
    Serial.println("terima");
    if (buttonState1 == LOW)
    {
        servol();
    }
    if (buttonState2 == LOW)
    {
        servo2();
    }
}
```

Gambar 8. Program Perintah Kendali Pintu (Servo)

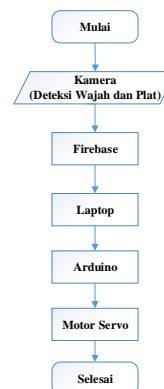
Pada gambar 8 merupakan program untuk menggerakkan motor servo atau palang pintu parkir, program akan menerima kode ASCII dari yang dapat dilihat pada gambar 7.



a. Pintu Tertutup b. Pintu Terbuka

Gambar 9. Tampilan Penerima Pada *Arduino*

Selanjutnya pada gambar 10 merupakan diagram alir *controller*, yang menunjukkan proses kerja dari program. Pada proses ini, sensor kamera akan mendeteksi wajah dan plat kendaraan dan setelah terdeteksi akan diperiksa pada *database* (*firebase*) untuk pencocokan. Apabila pencocokan deteksi wajah dan plat kendaraan sesuai dengan *database* maka *arduino* akan di perintahkan untuk menggerakkan motor servo untuk membuka pintu parkir, dan apabila wajah dan plat kendaraan tidak sesuai dengan *database* maka *arduino* diperintahkan untuk menutup pintu parkir. Berikut adalah tahapan *controller*.



Gambar 10. Tahapan Controller ke Servo

Untuk bisa menjalankan program *image processing* yang diatas, dibutuhkan beberapa modul. Berikut beberapa modul yang dibutuhkan:

1. Python 2.7.16
2. Opencv 4.1.1

3. *Pytesseract*
4. *Firebase*
5. *Serial*
6. *Pickle*
7. *Imutils*
8. *Threading*
9. *numpy*

Setelah semua modul telah ada, maka selanjutnya yaitu menginstall semua modul yang telah ada. Untuk menginstall python dan opencv dapat diikuti melalui petunjuk pada alamat *website* [16]. Modul ke tiga sampai ke Sembilan dapat mengikuti petunjuk melalui alamat *website* [17].

III. HASIL DAN DISKUSI

Gambar 11 merupakan hasil aktual rancangan *prototype* yang terdiri dari (a) panel dan (b) lahan parkir. Bahan yang digunakan untuk lahan parkir merupakan akrilik putih ketebalan 5 mm. Bahan akrilik dipilih karena pembuatannya yang mudah dengan *laser cut*, bobot yang lebih ringan dan merupakan bahan isolator yang baik. Stiker digunakan untuk melapisi akrilik yang akan digunakan sebagai pembuatan jalur kendaraan. Panel yang digunakan terbuat dari besi plat yang sudah terbentuk. Panel digunakan untuk merapikan komponen elektrik dan sebagai peletakan *prototype* yang dibuat.



Gambar 11. Purwarupa Sistem Keamanan Parkir

A. Pengujian Alat

Pengujian yang dilakukan dua tahap yaitu pengujian deteksi wajah dan plat kendaraan

- a. Pengujian Wajah
- b. Pegujian Pengenalan Wajah

Pengujian yang dilakukan untuk mengetahui jenis wajah yang dapat di deteksi.

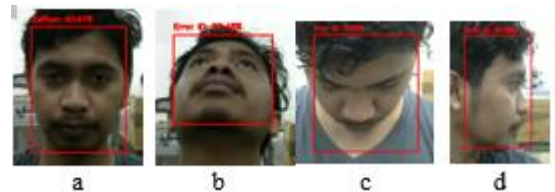
No	Wajah yang dideteksi	Pembacaan	Gambar
1	Manusia	Terdeteksi	
2	Hewan	Tidak terdeteksi	

Gambar 11. Hasil Pengujian Pengenalan Wajah

Terlihat bahwa sistem dapat membedakan wajah manusia dengan wajah binatang

c. Pegujian Deteksi Tampak Wajah

Pengujian dilakukan untuk mengetahui sudut wajah yang dapat terdeteksi oleh kamera.



Gambar 12. Hasil Pengujian Tampak Wajah

Terlihat bahwa akurasi pengenalan wajah dari beberapa sudut yang dapat terdeteksi oleh sistem, pada gambar diatas dapat dilihat sudut maksimum wajah yang dapat dideteksi oleh sistem.

d. Pegujian Deteksi Wajah

Pengujian dilakukan untuk mengetahui apakah sistem dapat mendeteksi wajah manusia.

No	Wajah	Jarak (cm)	Deteksi
1	Wajah	30 cm	Ya
2	Wajah	60 cm	Ya
3	Wajah	90 cm	Ya
4	Wajah	120 cm	Ya
5	Wajah	150 cm	Ya
6	Wajah	200 cm	Ya
7	Wajah	300 cm	Ya

Gambar 13. Deteksi Wajah

Pengujian dapat mendeteksi wajah pada jarak lebih dari 1 meter. Pengujian dilakukan pada malam hari di ruangan produksi AE, dari hasil pengujian yang dilakukan wajah dapat terdeteksi lebih dari 3 meter.

No	Jarak (cm)	Jumlah Benar				Akurasi (%)
		Database	Benar	Salah	Error	
1	40	5	5	0	0	100%
2	60	5	5	0	0	100%
3	80	5	5	0	0	100%

Gambar 14. Pengujian Deteksi Wajah Dengan Jarak Berbeda-Beda

Terlihat rata-rata akurasi dari pengujian sistem, pengujian dapat mendeteksi wajah sampai jarak pengujian yang dilakukan yaitu jarak 80 cm. Dilihat dari tingkat keberhasilan sensor kamera untuk mengenali wajah, semakin dekat wajah seseorang dengan kamera dan dengan tingkat pencahayaan yang cukup membuat tingkat keakuratan pengenalan wajah semakin tinggi.

e. Pengujian Tingkat kecocokan

Pengujian dilakukan untuk mengetahui tingkat kecocokan wajah, pengujian dilakukan pagi, siang, dan malam.

no	Pengujian	Waktu	Nilai Jarak (cm)	Nilai Cahaya (Lux)	Mirip	Sistem Mengenali Dengan Benar
1	Nandar	Malam	40	16	Nandar	Ya
2	Zulfikar	Malam	40	16	Zulfikar	Ya
3	Nandar	Pagi	40	30	Nandar	Ya
4	Zulfikar	Pagi	40	30	Zulfikar	Ya
5	Nandar	Siang	40	45	Nandar	Ya
6	Zulfikar	Siang	40	45	Zulfikar	Ya

Gambar 15. Pengujian Tingkat Kecocokan Wajah

Akurasi deteksi wajah berdasarkan intensitas cahaya (pagi 20-35 lux, siang 26-50 lux, dan malam 2-18 lux) dengan jarak 40 cm sebesar 100%. Apabila citra tidak dapat mendeteksi wajah dikarenakan faktor cahaya yang tidak merata pada permukaan wajah.

b. Pengujian Plat Kendaraan

Pengujian dilakukan untuk melihat apakah sistem dapat mengenal karakter tulisan. Metode yang digunakan adalah *Optical Character Recognition (OCR)*. Pada proses ini yang menjadi permasalahannya yaitu bagaimana menghasilkan gambar yang 'clear' sebelum diterjemakan oleh tesseract. Untuk itu, bisa melakukan perbaikan gambar terlebih dahulu, atau kombinasikan dengan pengolahan citra untuk meningkatkan kualitas gambar, mengurangi noise atau mempertajam teks pada gambar tersebut. Setelah proses selesai diatasi, tinggal mengubah format binary

(*Black and White*), mengatur *threshold* atau mengubah inversi colour, agar mudah diterjemahkan *tesseract*. Pada pengujian ini, posisi plat kendaraan di letakan tepat didepan kamera dengan jarak 40 cm, plat yang digunakan adalah hitam putih.



Gambar 16. Peletakan Kamera Pembaca Plat Kendaraan.

Pada gambar 16 Merupakan posisi peletakan kamera untuk membaca plat kendaraan yang akan masuk ke tempat parkir, posisi plat diletakan tepat didepan kamera yang berjarak 40 cm. Jarak peletakan plat bisa berubah tergantung dari cahaya yang didapatkan oleh plat, berikut adalah hasil pengujian pembacaan plat kendaraan.

Nomor Plat	Hasil Pembacaan	Kesalahan Pembacaan
B 4699 FXH	B 4699 FXH	-
B 5974 TDM	B 5974 TDM	-
B 4541 NIH	B 4541 NIH	-
B 4803 BWU	B 4803 BWU	-
B 3308 UUD	B 3308 UUD	-

Gambar 16. Pembacaan Plat Kendaraan

Dilihat dari hasil pengujian, sistem dapat bekerja lebih baik dibandingkan penelitian terdahulu yang dilakukan yang oleh Eko Didik Widianto tahun 2017 yang menggunakan metode yang sama yaitu metode *OCR*.

Kesimpulan yang didapat dari hasil pengujian yaitu penelitian terdahulu belum dapat menghasilkan gambar yang 'clear' sehingga pada saat diterjemakan oleh *tesseract*, hasil yang diinginkan tidak tercapai dimana dari 10 kali pengujian pembacaan plat kendaraan yang terbaca dengan baik tanpa kekurangan karakter adalah 6 plat kendaraan. Sehingga untuk mengatasi hal tersebut ditambahkan metode pengolahan citra untuk meningkatkan kualitas gambar, mengurangi *noise* atau mempertajam teks pada gambar. Setelah itu dilakukan pengujian kembali dengan yang dilakukan penelitian sebelumnya, dilihat dari pengujian yang telah dilakukan, dari 10 kali pengujian tidak ada karakter yang tidak terbaca atau 100% berhasil.

B. Faktor Yang Mempengaruhi Pengujian

Pada pengujian yang dilakukan ada beberapa faktor yang mempengaruhi pengujian yang dilakukan, salah satu faktor yang mempengaruhi adalah kamera. Kamera *Logitech Webcam C170* tidak dapat digunakan untuk integrasi dua kamera, hal itu dapat diketahui setelah melakukan pengujian pada kamera *Logitech Webcam C170*. Pada saat menjalankan program yang telah dibuat, program tidak dapat dijalankan atau terjadi *error*.

```
id : 0 nama : plat nomer : --Exception in thread Thread-2:
Traceback (most recent call last):
  File "C:\Python27\lib\threading.py", line 801, in __bootstrap_inner
    self.run()
  File "C:\Python27\lib\threading.py", line 754, in run
    self._target(*self._args, **self._kwargs)
  File "F:\complete_v2\complete_v2\zulfikar_TA.py", line 75, in wajah
    frame = imutils.resize(frame, width=600)
  File "C:\Python27\lib\site-packages\imutils\convenience.py", line 69, in resize
    (h, w) = image.shape[:2]
AttributeError: 'NoneType' object has no attribute 'shape'
```

Gambar 17. Program Error

Setelah melakukan pengujian beberapa kali menggunakan kamera *Logitech Webcam C170* dan hasil pengujian menunjukkan kamera *Logitech Webcam C170* bisa digunakan tetapi tidak bisa diintegrasikan untuk dua kamera kecuali digunakan secara bergantian. Sehingga penggunaan kamera *Logitech Webcam C170* tidak dapat digunakan untuk program yang telah dibuat sehingga diganti menggunakan kamera *Logitech Webcam C270* dan hasilnya dapat dilihat pada gambar 18.

```
Mengakses Database
jumlah member = 5
Wajah Mulai
Plat Nomer Mulai
[INFO] loading face detector...
[INFO] loading face recognizer...
Program Siap
id : 0 nama : plat nomer :
terkunci
id : 0 nama : plat nomer :
terkunci
```

Gambar 18. Program Tidak Terjadi Error

IV. KESIMPULAN

Berdasarkan pengujian yang dilakukan terhadap sistem selama proses penyusunan karya tulis ini, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Dapat mencegah pencurian kendaraan.

2. Dapat memberikan informasi parkir yang terisi dan belum terisi.
3. Dapat mengatasi identitas parkir yang tertinggal atau kehilangan.

DAFTAR PUSTAKA

- [1] Undang-Undang No. 14 Tahun 1992 tentang Lalu Lintas dan Angkutan Jalan. (1992). Jakarta: Sekretarian Negara.
- [2] P. (2019, Oktober 2). *Park Slopes Parking Problem And How to Fix It*. Retrieved from Internet: <http://www.transalt.org/newsroom/releases/126>
- [3] Rendi, C. (2011). Penentuan Lokasi Parkir pada Smart Parking System menggunakan Dynamic Weigting A*(DWA*). Bandung : Universitas Telkom.
- [4] *Sistem Informasi Parkir*. (2019, Oktobr Selasa). Internet: http://id.wikipedia.org/wiki/Sistem_informasi_parking
- [5] Eka, P. (2014). Smart City beserta Cloud Computing. Bandung: Informatika.
- [6] Agus, S. (2012). Pengertian dan Jenis Server. *Online*, <http://www.transiskom.com>.
- [7] Syahrul. (2006). Karakteristik dan Pengontrolan Servo. Bandung: Teknik Komputer, Universitas Komputer Indonesia.
- [8] Marjuni. (2015). Rancang Bangun Pemadam Api Menggunakan Sensor *Infrared* Berbasis Mikrokontroler ATmega 8535. *Jurnal Ilmiah Mahasiswa Vol 4 no.1*.
- [9] Arduino Mega 2560 Pinout. (2019, Oktober). <http://www.electroschematics.com/7963/arduino-mega-2560-pinout/>.
- [10] Ragil, F. G. (2015). Perancangan Aplikasi Pemantau Dan Pengendali Piranti Elektronik Pada Ruangan Berbasis Web. *Transmisi*, 17(2).
- [11] Wijatsongko, E. N. (2015). Sistem Pemantauan Ruangan Dengan Server Raspberry Pi. *IJEIS*.
- [12] Eko , D. W. (2017). Sistem Parkir Berbasis RFID dan Pengenalan Citra Pelat Nomor Kendaraan. *Program Studi Sistem Komputer, Fakultas Teknik, Universitas Diponegoro*, 115-122.
- [13] Arman, F. S. (2017). Sistem Parkir Berbasis QR Code Pada Perguruan Tinggi Raharja. *Seminar Nasional Multi Disiplin Ilmu*, 47-54.
- [14] Mohamad , D. A. (2018). Implementasi Sidik Jari sebagai Otentikasi Parkir Kendaraan. 760 – 767.
- [15] Hernanda , A. S. (2019). Deteksi dan Pengenalan Wajah sebagai Pendukung Keamanan. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1372-1380.
- [16] Ivanj. (2017, September 30). *Cara Install OpenCV Python 2.7 di Windows*. Diambil kembali dari Tutorial OpenCV & Android: <https://www.ivanjul.com/cara-install-opencv-dipython-2-7/>

[17] Ivanj. (2018, November 7). *Cara Install Numpy, Matplotlib, dll dengan Mudah Menggunakan PIP.*
Diambil kembali dari Tutorial OpenCV &

Android: <https://www.ivanjul.com/cara-install-numpy-matplotlib-dll-dengan-mudah-menggunakan-pip/>.