

# Implementasi *Face Recognition* Pada Smart Sistem Kamera Pada Gerbang Masuk

1<sup>st</sup> Kurniawan Azis  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

kurniawanazis@student.telkomuniversity.ac.id

2<sup>nd</sup> Sony Sumaryo  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

sonysumaryo@telkomuniversity.ac.id

3<sup>rd</sup> Azam Zamhuri  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

azamzamhurifuadi@telkomuniversity.a

**Abstrak** — Smart System Camera (Kamera Cerdas) untuk mendeteksi waktu masuk dan keluar dengan akurasi tinggi. Sistem ini terintegrasi dengan kamera CCTV yang dapat menampilkan waktu keluar masuk kendaraan secara real-time. Fitur-fitur seperti deteksi kendaraan, perhitungan jumlah kendaraan, deteksi plat nomor kendaraan, dan *face recognition* memungkinkan pengguna mengidentifikasi potensi ancaman dan berkomunikasi dengan orang lain cepat dan akurat. Sistem ini juga dapat dihubungkan dengan sistem keamanan lain seperti alarm untuk meningkatkan keamanan di area parkir. *Face recognition* sudah banyak pengimplementasiannya untuk absensi kehadiran pegawai, atau sistem keamanan pada tempat – tempat vital yang ada di wilayah tertentu, tetapi masih sedikit yang menggunakan *face recognition* untuk gerbang masuk suatu perumahan atau perkantoran. Pada pengimplementasi *face recognition* pada smart sistem camera pada gerbang masuk wajah pengguna akan terdeteksi pada kamera CCTV yang ada di gerbang masuk. Sebelumnya pengguna harus memasukan data wajah pengguna agar disaat melewati gerbang masuk atau keluar wajah pengguna terdeteksi oleh kamera CCTV, apabila pengguna tidak memasukkan data wajah ke dalam sistem maka pengguna tidak bisa menggunakan lahan parkir yang ada atau tidak bisa masuk ke dalam lingkungan perumahan. Pada sistem *face recognition* akan menampilkan nama, alamat tempat tinggal apabila di area perumahan, dan menampilkan jabatan atau pegawai apabila di area perkantoran

**Kata kunci:** Smart System Camera, Keamanan, Efisiensi, Tempat Parkir, Kamera CCTV, Mendeteksi kendaraan, Menghitung Jumlah Kendaraan, Mendeteksi Plat Nomor

## I. PENDAHULUAN

Transportasi telah menjadi bagian penting dalam kehidupan manusia, dan peningkatan transportasi di ruang publik telah menjadi masalah umum[1]. Ruang publik yang tidak terorganisir dan tidak menarik dapat menyebabkan masalah terkait keselamatan, keamanan, dan privasi. Oleh karena itu, diperlukan sistem yang mampu mengelola ruang publik secara efisien dan efektif[2].

*Face recognition* biasanya digunakan untuk absensi pegawai yang ada di perkantoran dengan bantuan *depth camera* dan pembuka kunci magnetis[3], sedangkan untuk

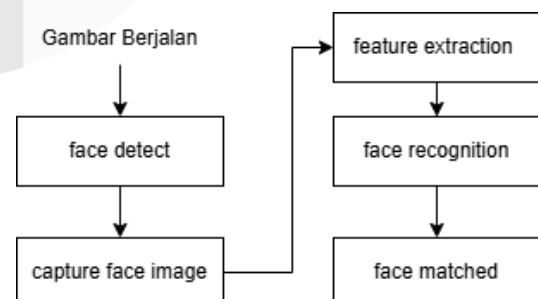
pengimplementasiannya pada lahan parkir atau gerbang masuk perumahan masih sangat jarang ditemukan.

Sistem keamanan di lahan parkir atau gerbang masuk suatu perumahan biasanya menggunakan kamera CCTV yang hanya dapat merekam lokasi dan kejadian, tetapi tidak dapat mencatat waktu kejadian[4]. Hal ini dapat menimbulkan kebingungan dalam pengelolaan ruang publik dan meningkatkan risiko keamanan.

Penelitian ini bertujuan untuk mengembangkan sistem Smart System Camera yang dapat mendeteksi wajah pengguna di gerbang masuk suatu perkantoran atau perumahan. Sistem ini bertujuan untuk meningkatkan keamanan dan privasi, serta menjadikan lahan parkir atau area perumahan lebih aman dan terorganisasi.

Pendeteksi wajah atau *face recognition* bekerja dengan cara memeriksa citra apabila memiliki citra maka secara otomatis akan dipisahkan dengan cara memotong citra wajah dari latar belakang. Citra wajah masih berupa perkiraan kasar serta memiliki kualitas yang kurang bagus, seperti ukuran yang belum sesuai, faktor pencahayaan, kejelasan citra yang buruk [5]. Teknologi ini dapat bekerja bersamaan dengan teknologi lain seperti *object detection*, dan *automatic license plate recognition*.

## II. METODE



GAMBAR 1  
diagram blok face recognition

Metode pengukuran yang digunakan untuk spesifikasi ini adalah metode Haarcascade dengan melakukan pemrosesan gambar, pengambilan objek

wajah dengan ukuran 218x218 pixel lalu akan menghasilkan gambar menjadi grayscale agar dapat mempermudah dalam proses training, selanjutnya dilakukan proses training dimana dataset wajah yang telah di capture di training dan dirubah menjadi .xml file. Prosedur pengujian pada spesifikasi ini yaitu pertama, melakukan perekaman wajah pengguna kendaraan kemudian melakukan training dari dataset wajah pengguna kendaraan yang berhasil di capture, lalu, memanggil hasil training, selanjutnya memasukkan library haarcascade dan terakhir menampilkan gambar dengan kamera dan program mendeteksi wajah dengan confident sendiri.

```

1 class VideoStream:
2     def __init__(self, src=0):
3         self.stream = cv2.VideoCapture(src)
4         self.stream.set(cv2.CAP_PROP_FRAME_WIDTH, 640)
5         self.stream.set(cv2.CAP_PROP_FRAME_HEIGHT, 360)
6         self.grabbed, self.frame = self.stream.read()
7         self.stopped = False
8
9     def start(self):
10        Thread(target=self.update, args=()).start()
11        return self
12
13    def update(self):
14        while True:
15            if self.stopped:
16                return
17            self.grabbed, self.frame = self.stream.read()
18
19    def read(self):
20        return self.frame
21
22    def stop(self):
23        self.stopped = True
24
25    @staticmethod
26    def face_recognition():
27        haarcascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
28        recognizer = cv2.face.LBPHFaceRecognizer_create()
29        recognizer.load('training_data.yml')
30        vs = VideoStream(0).start()
31        time.sleep(2.0)
32
33        while True:
34            start_time = time.time()
35            frame = vs.read()
36            if frame is None:
37                break
38
39            frame_gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
40            faces = haarcascade.detectMultiScale(frame_gray, scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))
41            for (x, y, w, h) in faces:
42                cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
43                id, conf = recognizer.predict(frame_gray[y:y+h, x:x+w])
44                if id > 0:
45                    cv2.putText(frame, str(id), (x+5, y-5), cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 255, 255), 2)
46                    cv2.putText(frame, str(conf), (x+5, y+h-5), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2)
47
48            end_time = time.time()
49            fps = 1 / (end_time - start_time)
50            cv2.putText(frame, f'FPS: {int(fps)}', (10, 60), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
51
52            ret, buffer = cv2.imencode('.jpg', frame)
53            frame = buffer.tobytes()
54            cv2.imshow('Face Recognition', frame)
55            if cv2.waitKey(1) &lt; 0:
56                break
57            vs.stop()
58
59 vs.stop()

```

GAMBAR 2 source code face recognition

dan tinggi 360 piksel menggunakan set() metode. Properti grabbed dan frame menyimpan status pembacaan frame terbaru dari video dan frame itu sendiri. Properti stopped digunakan untuk mengontrol kapan pengambilan frame dihentikan.

Fungsi ini bertanggung jawab untuk mendeteksi wajah dan mengenali identitasnya. Dimulai dengan memuat model deteksi wajah menggunakan CascadeClassifier dan model pengenalan wajah LBPHFaceRecognizer dari file pelatihan yang telah disediakan.

Kamera video dimulai menggunakan VideoStream dan diberi waktu untuk inialisasi selama 2 detik. Dalam loop utama, setiap frame diambil dari aliran video dan dikonversi ke skala abu-abu untuk deteksi wajah. Wajah-wajah dalam frame terdeteksi menggunakan detectMultiScale(). Setiap wajah yang terdeteksi kemudian diberi kotak hijau, dan ID serta tingkat kepercayaan hasil pengenalan wajah ditampilkan

Tingkat frame per detik (FPS) dihitung untuk setiap iterasi loop dan ditampilkan pada frame. Setiap frame kemudian dikodekan dalam format JPEG dan diubah menjadi byte untuk ditampilkan. Pada akhir fungsi, VideoStream dihentikan untuk menghentikan pengambilan video. Kode ini mengimplementasikan pendeteksian dan pengenalan wajah secara real-time menggunakan OpenCV dan model pengenalan wajah berbasis LBPH.

Cara kerja mendeteksi wajah pengguna adalah dengan mendeteksi wajah pengendara kendaraan yang menggunakan akses pintu masuk atau keluar dari area tersebut dengan menggunakan kamera yang sudah ditempatkan disisi pengemudi, yang dimana pengemudi harus membuka kaca kendaraannya agar terlihat oleh kamera yang sudah ditempatkan tersebut. Yang dimana output dari pendeteksi wajah ini akan keluar data nama dari wajah yang berhasil di deteksi

Haar Cascade digunakan dalam penelitian ini karena menawarkan metode yang cepat dan efisien untuk deteksi wajah. Algoritma Haar Cascade menggunakan fitur Haar yang sederhana namun efektif untuk mendeteksi wajah dalam gambar dan video real-time. Kecepatan deteksi yang tinggi memungkinkan pengolahan frame video secara langsung, sehingga ideal untuk penggunaan di lingkungan yang memerlukan respon cepat.

```

1 class VideoStream:
2     def __init__(self, src=ip_camera_url):
3         self.stream = cv2.VideoCapture(src)
4         self.stream.set(cv2.CAP_PROP_FRAME_WIDTH, 640)
5         self.stream.set(cv2.CAP_PROP_FRAME_HEIGHT, 360)
6         self.grabbed, self.frame = self.stream.read()
7         self.stopped = False
8
9     def start(self):
10        Thread(target=self.update, args=()).start()
11        return self
12
13    def update(self):
14        while True:
15            if self.stopped:
16                return
17            self.grabbed, self.frame = self.stream.read()
18
19    def read(self):
20        return self.frame
21
22    def stop(self):
23        self.stopped = True
24
25 vs.stop()

```

GAMBAR 3 source code face recognition

IV. HASIL DAN PEMBAHASAN



GAMBAR 4 output face recognition 1

Kelas VideoStream bertugas untuk menangani pengambilan video dari kamera. Kelas ini diinisialisasi dengan parameter src, yang merujuk pada URL kamera IP. Pada konstruktor (\_\_init\_\_), dilakukan inialisasi objek cv2.VideoCapture untuk mengambil video dari sumber yang diberikan. Resolusi video diatur pada lebar 640 piksel



GAMBAR 5  
output face recognition 2

#### A. Hasil pengujian

Dari hasil pengujian yang ada di lapangan sistem berjalan sesuai yang diharapkan yaitu dapat mendeteksi wajah pengguna dengan benar serta dapat menampilkan hasil dari deteksi sesuai apa yang diinginkan yaitu nama serta tempat tinggal.

Tetapi ada beberapa permasalahan yang didapatkan diantaranya adalah apabila posisi kamera berada di depan kendaraan maka wajah pengemudi tidak dapat terdeteksi, maka dari itu dilakukan penempatan yang berada disisi pengemudi agar wajah pengemudi dapat terdeteksi dengan syarat pengemudi harus membuka kaca pengemudi agar wajah lebih terlihat dengan jelas. Dan cahaya juga berpengaruh untuk hasil pendeteksian dikarenakan apabila cahaya tidak memadai maka wajah pengemudi tidak akan terdeteksi oleh sistem yang ada maka dari itu untuk di gerbang masuk atau keluar kendaraan diharapkan memasang pencahayaan yang cukup memadai agar wajah pengemudi dapat terlihat dengan jelas oleh sistem

#### V. KESIMPULAN

Berdasarkan hasil pengujian, dapat disimpulkan bahwa pengujian fitur Face Recognition pada Smart System Camera berjalan dengan baik sehingga menghasilkan output berupa pendeteksian wajah pengguna kendaraan beserta data pengguna kendaraan tersebut. Dengan beberapa catatan diantaranya adalah posisi kamera yang harus sesuai dengan posisi pengemudi serta pengemudi harus membuka kaca nya agar dapat terdeteksi oleh kamera, serta pencahayaan yang memadai pada gerbang masuk dan keluar kendaraan agar kamera dapat mendeteksi wajah dengan baik.

#### REFERENSI

- [1] Y. A. P. Ari Hidayatullah, "Perancangan Sistem Keamanan Perumahan Menggunakan," pp. 2-16, 2022.
- [2] Y. J. D. M. G. D. D. R. Desi Ramayanti, "Sistem Keamanan Perumahan Menggunakan Face Recognition," pp. 2-11, 2023.
- [3] C. D. Theresia Susim, "PENGOLAHAN CITRA UNTUK PENGENALAN WAJAH (FACE RECOGNITION) MENGGUNAKAN OPENCV," pp. 3-12, 2021.
- [4] R. F. Mukin, "Implementasi dan Reputasi Face Recognition Sebagai Kecerdasan Buatan di," pp. 2-9, 2023.
- [5] s. m. p. r. aulia, "aplikasi pendeteksi plat nomor kendaraan berbasis raspeberry pi menggunakan metode website untuk pelanggaran lalu lintas," pp. 84-89, 2019.