

Sistem Absensi dengan OpenCV Face Recognition dan Raspberry Pi

Astrid Nabila Prima[#], Cipto Prabowo[#], Rasyidah[#]

[#] *Jurusan Teknologi Informasi, Politeknik Negeri Padang, Padang, Indonesia*
E-mail: cipto@pnp.ac.id, rasyidah@pnp.ac.id

ABSTRACTS

Attendance data collection activities are routine. In general, this is done by signing an attendance sheet. This is considered slow and also causes disruption in carrying out lectures and activities. One solution to the problem is to use Face Recognition. Face Recognition is a biometric technology that has been widely applied in security systems in addition to eye retina recognition, fingerprint and iris recognition. In the application itself, face recognition uses a camera to capture a person's face and then compare it with faces that have previously been stored in a certain database. In the manufacturing process, Raspberry pi is used as the core of this tool. With the three things above, namely Attendance, Face Recognition, and Raspberry pi, a tool is formed that can meet the needs of an automatic, effective, and efficient attendance process.

ABSTRAK

Kegiatan pendataan kehadiran adalah hal yang rutin dilakukan. Pada umumnya, hal tersebut dilakukan dengan penandatanganan lembar kehadiran. Hal tersebut dinilai lambat dan juga menimbulkan gangguan dalam melaksanakan perkuliahan maupun kegiatan. Salah satu solusi dari permasalahan adalah menggunakan Face Recognition, Face Recognition adalah salah satu teknologi biometrik yang telah banyak diaplikasikan dalam sistem security selain pengenalan retina mata, pengenalan sidik jari dan iris mata. Dalam aplikasinya sendiri pengenalan wajah menggunakan sebuah kamera untuk menangkap wajah seseorang kemudian dibandingkan dengan wajah yang sebelumnya telah disimpan di dalam database tertentu. Dalam proses pembuatan digunakanlah Raspberry pi sebagai inti dari alat ini. Dengan tiga hal diatas yaitu Absensi, Face Recognition, dan Raspberry pi terbentuklah alat yang dapat memenuhi kebutuhan proses absensi yang otomatis, efektif, dan efisien.

KATA KUNCI

*Absensi,
Image Processing,
Face Recognition,
Eigenface,
Pi Camera,
Raspberry pi*

1. PENDAHULUAN

Kegiatan pendataan kehadiran adalah hal yang rutin dilakukan. Pada umumnya, hal tersebut dilakukan dengan penandatanganan lembar kehadiran. Hal tersebut dinilai lambat dan juga menimbulkan gangguan dalam melaksanakan perkuliahan maupun kegiatan.

Dalam proses absensi pada umumnya dapat menghabiskan banyak waktu dalam proses absensi seperti pemanggilan nama mahasiswa dan juga saat mendengarkan pemanggilan nama, mahasiswa bisa saja kehilangan fokus pada dosen. Selain itu jika menggunakan system absensi secara manual tidak menjamin kemungkinan adanya kekurangan dalam pengambilan absen tersebut. Sebagai contoh adanya yang memalsukan tanda tangan seorang rekan dalam pengambilan absen manual atau penduplikatan sidik jari bahkan menitipkan kartu identitas dalam mesin scan absensi.

Hasil dari fitur face recognition pada alat, akan membantu dalam proses identifikasi Dosen maupun Mahasiswa secara akurat untuk menambah fungsi dalam absensi. Dengan melakukan scanning pada mesin absensi, kemudian dilakukan pengolahan data dengan aplikasi maka akan menghasilkan informasi kehadiran yang lebih akurat dibandingkan dengan yang dilakukan secara manual dan tentunya menghabiskan lebih banyak waktu. Pada face recognition ini memerlukan library OpenCV yang dalamnya menggunakan algoritma Eigenface untuk pengenalan wajah.

2. METODOLOGI PENELITIAN

2.1. Studi Literatur.

Studi Literatur dilakukan untuk melakukan pengkajian atau penguraian mengenai sistem pengenalan wajah, konsep ekstraksi ciri, dan algoritma pengklarifikasian menggunakan berbagai referensi yang mendukung dalam menganalisis permasalahan yang ada.

2.2. Analisis Sistem.

Analisis pendeteksi pengambilan absen dengan menggunakan raspberry ini menggunakan sistem digital. Sistem akan dibangun dengan menggunakan raspberry pi yang digunakan untuk pengambilan absen dengan menggunakan kamera dalam penginputan wajah. Setelah itu data pengambilan absen akan disimpan di database dan outputnya akan terlihat pada web data absensi yang tersedia.

Dalam pembuatan Implementasi Sistem Absensi menggunakan Open-CV Face Recognition dan Raspberry pi dibutuhkan beberapa sistem perangkat lunak (software), perangkat keras (hardware), dan kebutuhan pengguna sistem (brainware).

1. Perangkat Lunak

Perangkat lunak yang digunakan untuk pembuatan Implementasi Sistem Absensi menggunakan Open-CV Face Recognition dan Raspberry pi adalah sebagai berikut :

TABEL 1. Spesifikasi Software

| NO | Software | Keterangan |
|----|----------------|--|
| 1 | Raspbian | Sistem Operasi yang digunakan untuk menjalankan program. |
| 2 | MySQL | <i>Software</i> yang digunakan untuk pembuatan Databbase system |
| 3 | Library OpenCV | <i>Library</i> yang digunakan untuk membantu di bidang <i>image processing</i> . |

2. Perangkat Keras (Hardware).

Perangkat keras yang mendukung dalam menjalankan Implementasi Sistem Absensi menggunakan Open-CV Face Recognition dan Raspberry pi adalah sebagai berikut :

TABEL 2. Spesifikasi Hardware

| No | Perangkat Keras (<i>Hardware</i>) | Keterangan |
|----|--|-------------|
| 1 | Processor | 1GHz |
| 2 | Pi Camera | Disesuaikan |
| 3 | RAM | 512 MB |
| 4 | SD Card | 16GB |

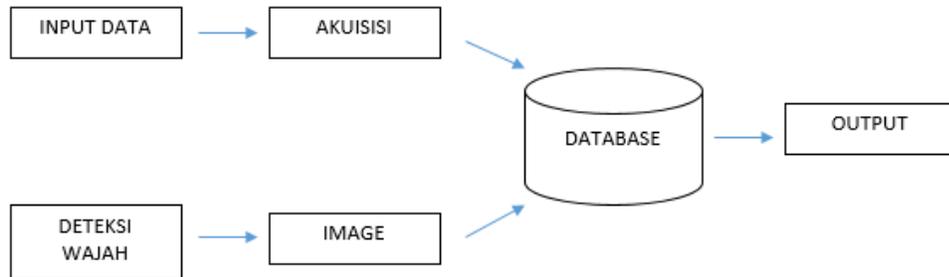
3. Perancangan Aplikasi.

- Perancangan Alur Kerja Sistem

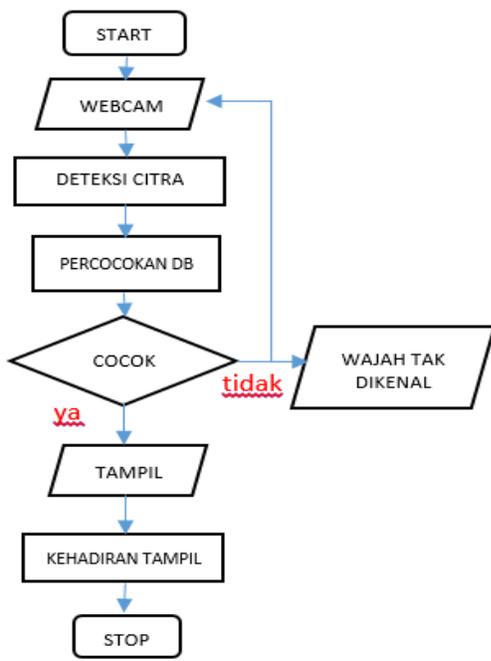
Pada tahap ini dilakukan proses perancangan desain sistem aplikasi. Perancangan desain sistem yaitu merancang sebuah sistem yang mampu mendeteksi citra wajah seseorang dan mampu mengenali citra wajah orang yang berada di ruangan tersebut. Pada aplikasi ini terdapat user Administrator yaitu Admin. Apabila user login sebagai Admin maka user akan masuk ke halaman Admin untuk melakukan proses pengolahan data.

Untuk proses pengumpulan data, pertama-tama capture objek wajah menggunakan webcam dari PiCamera. Selanjutnya sistem akan mendeteksi keberadaan wajah pada gambar yang ditampilkan.

Kemudian jika objek wajah tersebut telah dideteksi lalu akan dikonversi kedalam warna grayscale dengan menggunakan library OpenCV. Lalu objek wajah yang sudah dikonversi ke dalam grayscale akan di cropping dan scaling dengan tujuan untuk menyeragamkan ukuran lebar dan panjang citra wajah. Kemudian kita akan mengambil citra wajah tersebut sebanyak 5 sampai 10 kali dengan 5 ekspresi yang berbeda. Selanjutnya inputkan nama dan id orang tersebut. Setelah itu data akan disimpan ke dalam tabel yang ada di dalam Database.



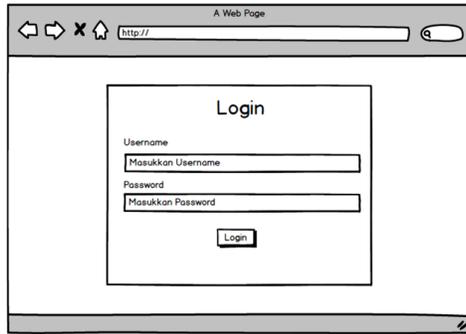
GAMBAR 1. Proses Pengumpulan dan Deteksi Citra Wajah



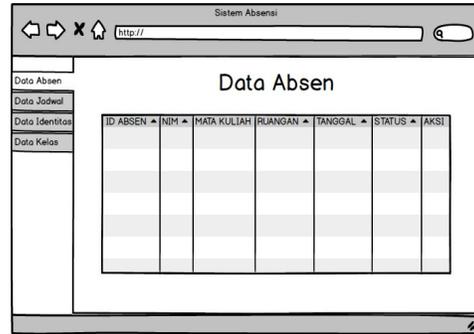
GAMBAR 2. Flowchart pengenalan wajah Citra

Pengisian data dilakukan secara manual dengan memasukan nama dan id seseorang. Setelah melakukan penginputan data, akan dilakukan akuisisi data dimana dalam proses ini data hasil pengambilan data wajah akan di proses dalam fitur Haar yang ada dalam Eigenface untuk mendapatkan data wajah masing – masing orang. Setelah proses akuisisi maka data sudah dapat dimasukkan ke dalam Database. Pada proses absensi harus menunjukkan wajahnya ke dalam alat perekam kemudian dibandingkan dengan data wajah yang sudah ada, jika cocok maka akan masuk kedalam Database, jika tidak maka orang tersebut harus mengulang.

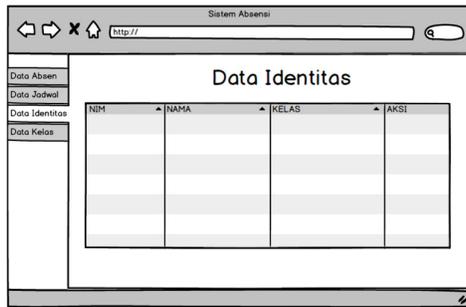
- Perancangan *Interface* (Antarmuka)
 Interface Sistem Absensi menggunakan Face Recognition dan Raspberry pi. Gambar 3 merupakan desain dari halaman awal aplikasi output yaitu login. Di dalam form login ini terdiri dari username, password, dan tombol login. Pada halaman home hanya akan menampilkan data absensi dengan tambahan empat button yaitu data mahasiswa yang akan menampilkan layer data jadwal, identitas, dan kelas jika menekan tombol tersebut. Pada halaman data identitas ini akan menampilkan data umum mahasiswa seperti nama, nomor induk mahasiswa, kelas, dan dilengkapi dengan kolom aksi (edit dan delete data). Pada halaman data jadwal akan menampilkan data jadwal kelas setiap harinya dengan opsi id jadwal, mata pelajaran, ruangan, kelas, hari, jam masuk, dan jam keluar beserta opsi aksi. Pada halaman data kelas menampilkan kumpulan data mahasiswa sesuai dengan kelas masing-masing



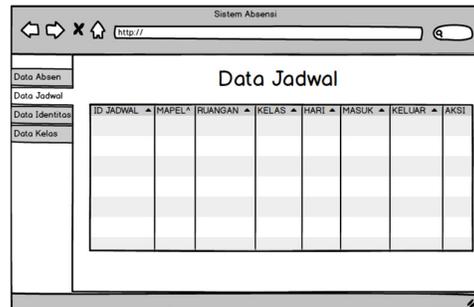
GAMBAR 3. Login



GAMBAR 4. Tampilan Data Absen



GAMBAR 5. Tampilan Layer Data Identitas



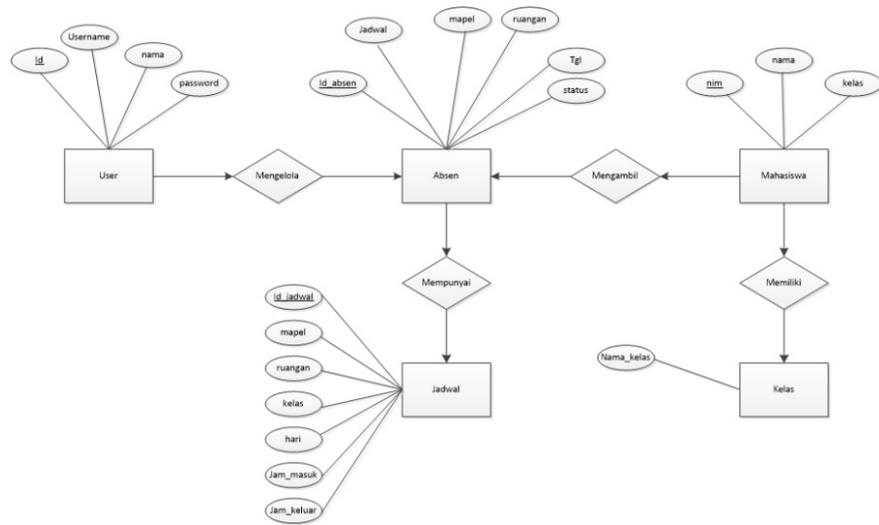
GAMBAR 6. Tampilan Layer Data Jadwal

- Perancangan Database

Data disimpan dalam Database SQL Server. Berikut ini adalah desain atau perancangan Database Implementasi Sistem Absensi Menggunakan OpenCV-Phyton Face Recognition dan Raspberry pi.

a. ERD (Entity Relation Diagram)

Entity Relational Diagram ini nantinya akan digunakan sebagai blueprint pembuatan Database pada Implementasi Sistem Absensi Menggunakan OpenCV-Phyton Face Recognition dan Raspberry pi.. Dalam ERD ini menjelaskan relasi antar tabel dan kerja sistem pada aplikasi. Tabel Mahasiswa berisi beberapa Mahasiswa yang akan melakukan absensi disini mahasiswa berelasi dengan Tabel Absen dan Tabel Kelas. Mahasiswa melakukan proses absensi untuk kehadiran dan akan tercatat data pada absensi,. Gambar 3.7 menunjukkan ERD dari sistem.



GAMBAR 7. Desain ERD database

Pada *Database* diatas, terdapat berbagai macam tabel, yaitu :

a) Tabel Identitas.

Tabel Identitas digunakan untuk merekam data-data mahasiswa. Mahasiswa adalah objek utama dalam lingkungan kampus terutama dalam hal absensi. Dengan adanya pencatatan data mahasiswa maka dapat dilakukan dokumentasi daftar mahasiswa yang hadir dalam perkuliahan maupun tidak

TABEL 3. Tabel Identitas

| Nama Field | Type Data | Panjang | Keterangan |
|------------|-----------|---------|------------------------------|
| Nim | Int | 11 | Nomor Induk Mahasiswa |
| nama | Varchar | 100 | Nama Mahasiswa |
| kelas | Varchar | 100 | Kelas atau tingkat Mahasiswa |

b) Tabel Jadwal

Tabel Jadwal digunakan untuk menentukan jadwal mata pelajaran setiap harinya dilengkapi dengan jam masuk dan keluar dari setiap mata pelajaran.

TABEL 4. Tabel Jadwal

| Nama Field | Type Data | Panjang | Keterangan |
|------------|-----------|---------|------------------------|
| Id_jadwal | Int | 11 | Id Program Studi |
| mapel | Varchar | 100 | Mata Pelajaran |
| ruangan | Varchar | 100 | Ruangan yang digunakan |
| kelas | Varchar | 100 | Kelas Mahasiswa |
| hari | text | | Hari waktu perkuliahan |
| jam_masuk | time | | Jam masuk perkuliahan |
| jam_keluar | time | | Jam keluar perkuliahan |

c) Tabel Kelas

Tabel Kelas digunakan untuk menentukan kelas berdasarkan dengan nim masing-masing mahasiswa.

TABEL 5. Tabel Kelas

| Nama Field | Type Data | Panjang | Keterangan |
|------------|-----------|---------|--------------------|
| nim | Int | 11 | Id Program Studi |
| namakelas | Varchar | 50 | Nama Program Studi |

d) Tabel Absen

Tabel absen digunakan untuk merekam absensi mahasiswa berdasarkan data-data yang sudah ada.

TABEL 6. Tabel Absen

| Nama Field | Type Data | Panjang | Keterangan |
|------------|-----------|---------|--|
| Id_absen | Int | 11 | Id Absensi |
| jadwal | Int | 11 | Jadwal jam kelas dimulai |
| nim | Int | 11 | Nomor Induk Mahasiswa |
| mapel | text | | Mata Pelajaran yang berlangsung. |
| ruangan | text | | Ruang kelas |
| tgl | date | | Tanggal pengambilan absen berlangsung. |
| status | text | | Status kehadiran mahasiswa. |

e) Tabel *Users*

Tabel users digunakan untuk merekam data *users* yang dapat mengakses aplikasi.

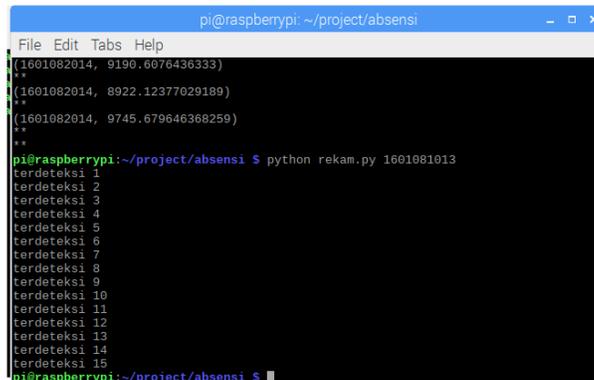
| TABEL 7. Tabel User | | | |
|---------------------|-----------|---------|-----------------------------|
| Nama Field | Type Data | Panjang | Keterangan |
| id | Int | 8 | Id <i>Users</i> |
| username | Varchar | 30 | <i>Username</i> untuk login |
| password | Varchar | 40 | <i>Password</i> untuk login |
| nama | Varchar | 60 | Nama <i>user</i> |

3. HASIL DAN PEMBAHASAN

Pengujian yang dilakukan mengenai perangkat yang telah dibangun bertujuan untuk mengetahui bahwa sistem yang dibangun telah berhasil sesuai dengan keinginan.

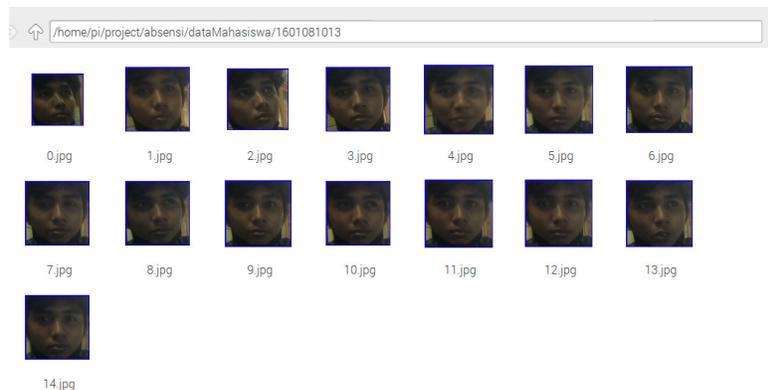
1. Pengujian Rekam Data Mahasiswa

Pengujian rekam data mahasiswa adalah pengujian dari *source code* yang telah dibuat dengan menggabungkan semua alat yang diperlukan. Yang perlu dilakukan adalah menjalankan sistem dengan memanggil *command file* rekam.py + NIM.



GAMBAR 8. Pengujian Rekam Data Mahasiswa

Hasil rekam dapat dilihat pada file '*absensi/dataMahasiswa/NIM*' sesuai dengan *source code* yang telah dibuat sebelumnya.



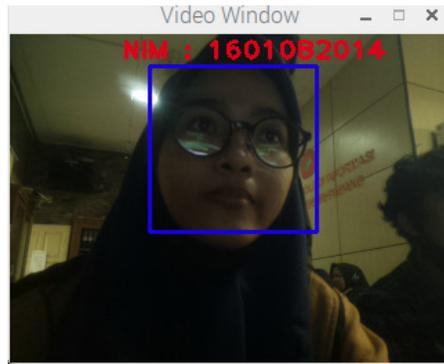
GAMBAR 9. Hasil Rekam Data Mahasiswa

2. Pengujian *Face Recognition*

Pengujian *Face Recognition* dapat dilakukan setelah melakukan uji rekam data mahasiswa. Pada proses pengujian ini, dilakukan dalam tiga kondisi yang membuktikan bahwa cahaya sangat berperan penting dalam proses *face recognition*. Faktor pencahayaan sangat berpengaruh pada proses pendeteksi dan pengenalan mahasiswa untuk proses presensi. Terang gelapnya cahaya dapat mempengaruhi data training pada wajah dan jika pada proses pengenalan juga kekurangan cahaya maka akan mempengaruhi proses absensi.

3. Pengujian Absensi Mahasiswa

Pada pengujian ini absensi mahasiswa, yang perlu dilakukan adalah menjalankan sistem dengan memanggil *command file* absensi.py maka akan menampilkan jendela kamera yang akan digunakan sebagai perantara absensi.



GAMBAR 13. Melakukan Absensi dengan Face Recognition

Setelah wajah mahasiswa terdeteksi maka akan tampil seperti gambar 4.20. NIM yang terdeteksi berdasarkan wajah yang disesuaikan dengan data mahasiswa tersebut, lalu data NIM yang terdeteksi dikirim ke server. Data Nim yang masuk akan diolah oleh sistem dan akan diteruskan untuk proses absensi, berikut data yang sudah masuk dan diolah pada Gambar 4.21.

| ID | NIM | Status | Tanggal | Waktu | Aksi |
|----|------------|--------|------------|--------|-------------|
| 4 | 123 | - | 2020-07-28 | keluar | EDIT DELETE |
| 5 | 1701902030 | - | 2020-07-28 | keluar | EDIT DELETE |
| 6 | 123 | - | 2020-07-28 | masuk | EDIT DELETE |
| 7 | 1701902030 | - | 2020-07-28 | masuk | EDIT DELETE |
| 8 | 123 | - | 2020-08-19 | keluar | EDIT DELETE |
| 9 | 123 | - | 2020-08-19 | masuk | EDIT DELETE |
| 10 | 123 | - | 2020-08-19 | keluar | EDIT DELETE |
| 11 | 123 | - | 2020-08-19 | masuk | EDIT DELETE |
| 12 | 1601082014 | - | 2020-08-19 | masuk | EDIT DELETE |
| 13 | 1601082014 | - | 2020-08-19 | keluar | EDIT DELETE |

GAMBAR 14. Hasil Pengujian Mahasiswa yang berhasil masuk ke dalam

4. Pengujian Pengaruh Intenstas Cahaya

Sistem bisa mendeteksi dan mengenali dengan sangat baik pada intensitas cahaya lebih dari 300 lux dan sistem bisa mendeteksi dan mengenali dengan baik pada intensitas cahaya antara 20 - 250 lux. Lalu sistem tidak bisa mendeteksi maupun mengenali citra wajah pada intensitas cahaya 0 lux. Tabel 4.1 dan 4.2 adalah hasil uji coba Pendeteksian dan pengenalan berdasarkan intensitas Cahaya.

TABEL 8. . Tabel Uji Coba Pendeteksian Berdasarkan Intensitas Cahaya

| NO | Intensitas Cahaya (lux) | Deteksi | | Keterangan |
|----|---------------------------|---------|---|------------------------------|
| | | Y | T | |
| 1. | Intensitas cahaya 0 lux | | ✓ | Tidak dapat mengenali wajah. |
| 2. | Intensitas cahaya 20 lux | ✓ | | |
| 3. | Intensitas cahaya 60 lux | ✓ | | |
| 4. | Intensitas cahaya 100 lux | ✓ | | |

| | | | | |
|----|-----------------------------|---|--|--|
| 5. | Intensitas cahaya 120 lux | ✓ | | |
| 6. | Intensitas cahaya 180 lux | ✓ | | |
| 7. | Intensitas cahaya 250 lux | ✓ | | |
| 8. | Intensitas cahaya > 300 lux | ✓ | | |

TABEL 9. . Tabel Uji Coba Pengenalan Berdasarkan Intensitas Cahaya

| NO | Intensitas Cahaya (lux) | Deteksi | | Keterangan |
|----|-----------------------------|---------|---|------------------------------|
| | | Y | T | |
| 1. | Intensitas cahaya 0 lux | | ✓ | Tidak dapat mengenali wajah. |
| 2. | Intensitas cahaya 20 lux | | ✓ | Tidak dapat mengenali wajah. |
| 3. | Intensitas cahaya 60 lux | ✓ | | |
| 4. | Intensitas cahaya 100 lux | ✓ | | |
| 5. | Intensitas cahaya 120 lux | ✓ | | |
| 6. | Intensitas cahaya 180 lux | ✓ | | |
| 7. | Intensitas cahaya 250 lux | ✓ | | |
| 8. | Intensitas cahaya > 300 lux | ✓ | | |

5. Pengujian Pengaruh Jarak Pada *Face Recognition*

Keberhasilan pendeteksian dan pengenalan wajah dengan variabel jarak 30 cm dapat mendeteksi dan mengenali mahasiswa dengan sangat baik. Berikut adalah tabel pengujian pengaruh jarak pada *face recognition*.

TABEL 10. . Uji Coba Berdasarkan Jarak (cm)

| NO | Jarak (cm) | Deteksi | | Pengenalan | | Keterangan |
|----|---------------------|---------|---|------------|---|--|
| | | Y | T | Y | T | |
| 1. | Deteksi pada <20 cm | | ✓ | | ✓ | |
| 2. | Deteksi pada 20 cm | ✓ | | ✓ | | |
| 3. | Deteksi pada 30 cm | ✓ | | ✓ | | Jarak yang paling sesuai. |
| 4. | Deteksi pada 90 cm | ✓ | | ✓ | | |
| 5. | Deteksi pada 150 cm | ✓ | | | ✓ | Sistem dapat melakukan pendeteksian namun tidak dapat melakukan pengenalan pada objek. |
| 6. | Deteksi > 150 cm | | ✓ | | ✓ | Tidak dapat melakukan pendeteksian maupun pengenalan. |

6. Pengujian Face *Recognition* Dengan Beberapa Objek
Keberhasilan pendeteksian dan pengenalan wajah dengan objek 2-3 mahasiswa dapat mendeteksi dan mengenali mahasiswa dengan sangat baik pada jarak tertentu. Berikut adalah tabel pengujian *face recognition* dengan beberapa objek.

TABEL 11. . Uji Coba Berdasarkan Jumlah Objek

| NO | Mahasiswa | Terdeteksi Jarak(cm) | | | | | | | | | |
|----|-------------|----------------------|---|-------|---|--------|---|--------|---|------|---|
| | | 30 cm | | 90 cm | | 100 cm | | 120 cm | | >120 | |
| | | Y | T | Y | T | Y | T | Y | T | Y | T |
| 1. | 1 Mahasiswa | ✓ | | ✓ | | ✓ | | | ✓ | | ✓ |
| 2. | 2 Mahasiswa | ✓ | | ✓ | | ✓ | | | ✓ | | ✓ |
| 3. | 3 Mahasiswa | ✓ | | ✓ | | | ✓ | | ✓ | | ✓ |
| 4. | 4 Mahasiswa | | ✓ | | ✓ | | ✓ | ✓ | | | ✓ |

4. KESIMPULAN

Kesimpulan yang dapat diambil dari pembahasan tugas akhir ini sebagai berikut.

1. Algoritma eigenface yang digunakan dalam proses pengenalan wajah sudah mampu berjalan dengan baik pada kondisi wajah tegak ke depan.
2. Tinggi rendahnya unsur cahaya dalam ruangan sangat mempengaruhi proses pendeteksi wajah dan jika kekurangan cahaya pada saat melakukan absensi maka, akan mempengaruhi ketepatan dalam sistem absensi.
3. Jarak pengambilan citra wajah menggunakan face recognition juga mempengaruhi ketepatan dalam melakukan pengenalan wajah. Pada saat melakukan absensi, sebaiknya tidak mengambil jarak yang terlalu jauh dengan kamera.
4. Keberhasilan dalam pedekteksi wajah ini berjalan dengan baik dalam jarak lebih kurang 30cm saat melakukan pendeteksian setiap mahasiswa.
5. Dalam proses pendeteksian dan pengenalan wajah mahasiswa, sistem dapat membaca 3 data mahasiswa dalam satu frame sekaligus pada jarak 30 – 90cm.

REFERENSI

- [1] Brahmhbhatt, S. (2013). *Practical OpenCV*. New York: Appres.
- [2] Ni Wayan Marti. (2010). *Pemanfaatan GUI Dalam Pengembangan Perangkat Lunak Pengenalan Citra Wajah Manusia menggunakan Metode Eigenface*.
- [3] Desa, P. K. (2017). *BUKU PANDUAN PEMOGRAMAN PYTHON*. relawanTIK.
- [4] Sepritahara. (2015). *SISTEM PENGENALAN WAJAH (FACE RECOGNITION) MENGGUNAKAN METODE HIDDEN MARKOV MODEL (HMM)*.
- [5] Singh, V. S. (2013). Face Detection By Haar Cascade Classifier With Simple And Complex Backgrounds Images Using OpenCV Implementation. *International Journal of Advanced Technology in Engineering and Science (IJATES)*, 1(12), 33-38.
- [6] “BBC – dot.Rory: A 15 pound computer to inspire younf programers” .bbc.co.uk
- [7] Prasetyo, Eri dan Rahmatun Isna. *Face Recognition System Design With Expression Position and Variation Method Using Eigenface*. pusatstudi.gunadarma.ac.id.
- [8] Karen E. Kalumuck (2000). *Human body explorations: hands-on investigates of what makes us tick*. Kendall Hunt.