

---

---

## RANCANG BANGUN APLIKASI PENGENALAN WAJAH UNTUK SISTEM PRESENSI DI PRODI TEKNIK INFORMATIKA UNIVERSITAS ISLAM RADEN RAHMAT MALANG MENGGUNAKAN METODE LBPH DAN LIBRARY OPENCV BERBASIS ANDROID

Gresy Ayu Wulandari✉, Muhammad Sonhaji Akbar, Bila Nastiti Tasaufi

Program Studi Teknik Informatika, Universitas Islam Raden Rahmat Malang, Indonesia

Email: [gresyayu15@gmail.com](mailto:gresyayu15@gmail.com)

DOI: <https://doi.org/10.46880/jmika.Vol7No2.pp247-253>

### ABSTRACT

*The attendance system is a mechanism to record the attendance of students in each class they take. The attendance system serves as evidence of students' presence in the learning process. In its journey, Universitas Islam Raden Rahmat has been using manual attendance through attendance journals in the learning process. However, with technological advancements, the attendance process has followed this trend, one of which is face recognition. In this research, the researcher designed a face recognition application for the attendance system that can be implemented on mobile platforms, especially Android. This application using the Local Binary Pattern Histogram (LBPH) method and the OpenCV library. The objective of this application is to enhance the flexibility and efficiency of the current attendance system by eliminating the need to repeatedly retrieve attendance forms, preventing fraud, and reducing paper usage. By using the waterfall system development method, testing showed a face recognition accuracy of 70%.*

**Keyword:** Attendance System, Face Recognition, LBPH, Android.

### ABSTRAK

*Sistem presensi adalah suatu mekanisme untuk mencatat kehadiran mahasiswa pada setiap perkuliahan yang diambil. Fungsinya sebagai bukti kehadiran mahasiswa dalam proses pembelajaran. Universitas Islam Raden Rahmat dalam perjalanannya menggunakan presensi manual berupa jurnal kehadiran dalam proses pembelajaran. Namun, dengan kemajuan teknologi, proses presensi telah mengikuti tren tersebut, salah satunya adalah pengenalan wajah. Dalam penelitian ini, peneliti merancang sebuah aplikasi pengenalan wajah (face recognition) untuk sistem presensi yang dapat diimplementasikan pada platform mobile, terutama Android. Aplikasi ini menggunakan metode Local Binary Pattern Histogram (LBPH) dan pustaka OpenCV. Tujuan aplikasi ini adalah meningkatkan fleksibilitas dan efisiensi sistem kehadiran saat ini karena tidak perlu berulang kali mengambil formulir daftar kehadiran, mencegah kecurangan, dan mengurangi penggunaan kertas. Dengan menggunakan metode pengembangan sistem waterfall, pengujian menunjukkan akurasi pengenalan wajah sebesar 70%.*

**Kata Kunci:** Sistem Presensi, Face Recognition, LBPH, Android.

---

### PENDAHULUAN

Universitas merupakan salah satu tempat pada jenjang perguruan tinggi dimana proses belajar mengajar berlangsung yang melibatkan civitas akademika yaitu dosen dan mahasiswa. Menurut Undang-Undang Nomor 12 Tahun 2012 Tentang Pendidikan Tinggi proses pembelajaran terjadi ketika mahasiswa berinteraksi dengan dosen dan menggunakan sumber belajar dalam suatu lingkungan belajar. Dalam peran sebagai anggota Sivitas Akademika, dosen memiliki tanggung jawab untuk mentransfer pengetahuan dan/atau teknologi yang dimilikinya kepada mahasiswa, dengan menciptakan

lingkungan belajar yang memungkinkan mahasiswa untuk aktif mengembangkan potensi mereka (Republik Indonesia, 2012).

Seperti halnya di sekolah, pada universitas pun mahasiswa juga masih memerlukan proses presensi ketika pembelajaran akan berlangsung. Sistem presensi ini dapat menjadi bukti yang menyatakan bahwa mahasiswa yang bersangkutan telah hadir dalam melaksanakan perkuliahan. Sistem presensi ini kemudian dapat digunakan sebagai acuan untuk penilaian akhir mahasiswa dan juga digunakan sebagai salah satu faktor penentu keikutsertaan mahasiswa dalam ujian akhir.

Pada proses presensi yang dilakukan khususnya di Prodi Teknik Informatika UNIRA Malang, peneliti mengamati secara langsung bahwa pada proses pengecekan daftar kehadiran mahasiswa dalam perkuliahan yang diambil masih dilakukan secara konvensional yaitu dengan mengambil formulir kehadiran di ruang fakultas dan kemudian membubuhkan tanda tangan pada formulir daftar kehadiran tersebut. Hal tersebut dapat memberikan kesempatan bagi mahasiswa untuk melakukan kecurangan seperti melakukan titip absen dan menggandakan tanda tangannya sendiri. Kesadaran mahasiswa untuk tidak melakukan titip absen serta menggandakan tanda tangannya sendiri guna tercatat dalam daftar kehadiran namun kenyataannya mahasiswa tersebut tidak hadir saat perkuliahan sedang berlangsung, dirasa masih kurang karena nyatanya masih ada mahasiswa yang melakukan perilaku tidak terpuji tersebut.

Dewasa ini *smartphone* menjadi salah satu benda yang penggunaannya tidak lepas dari sekian banyak kegiatan yang dilakukan oleh manusia. Sistem presensi menggunakan pengenalan wajah sendiri awalnya merupakan aplikasi komputer, namun kini penggunaannya semakin luas dan digunakan pada *platform mobile* seperti *Android*. *Face recognition* merupakan teknologi yang memungkinkan pengguna untuk mengenali wajah seseorang melalui citra digital (Clara, 2022).

Terdapat penelitian terdahulu yang telah mengkaji tentang metode pengenalan wajah menggunakan *lbph* yang dilakukan oleh Hilman (2022) yang berjudul “Implementasi *Face Recognition* Secara *Real-Time* Dengan Metode *Haar Cascade Classifier* Menggunakan *Opencv-Python*”. Hasil yang didapatkan adalah sistem dapat mengenali pengguna awal dengan pada jarak jauh dan dekat, namun pada faktor pencahayaan yang terlalu terang dan gelap serta posisi menghadap ke atas didapatkan nilai akurasi dibawah 60%.

Penelitian lainnya dilakukan oleh (Kosasih & Daomara, 2021) yang berjudul “Pengenalan Wajah Dengan Menggunakan Metode *Local Binary Patterns Histograms* (LBPH)”. Hasil yang didapatkan yaitu penggunaan metode LBPH untuk pengenalan wajah mampu memperoleh tingkat akurasi sebesar 86% dengan total data citra wajah sebanyak 750 citra

Selanjutnya penelitian oleh (Pratama, Lusni, Dara, & Sahertian, 2021) yang berjudul “Perbandingan Pengenalan Wajah Dengan Metode *Local Binary Pattern Histogram* Dan *Eigenface* Untuk Presensi”.

Hasil penelitian yang didapatkan yaitu aplikasi yang membandingkan pengenalan wajah menggunakan metode *Local Binary Pattern Histogram* dan *Eigenface* untuk keperluan presensi mampu mengidentifikasi wajah dengan bantuan metode identifikasi wajah, yaitu *viola-jones*. Dengan posisi wajah menghadap langsung ke kamera dan jarak kurang dari 40 cm, aplikasi ini mencapai tingkat akurasi yang tinggi, yaitu 91% untuk LBPH dan 79% untuk *Eigenface*.

Pada kajian akhir ini peneliti merancang sebuah aplikasi *face recognition* untuk sistem presensi yang penggunaannya dapat diimplementasikan ke *platform mobile* khususnya *Android*. Pengimplementasian aplikasi ini menggunakan pustaka *OpenCV*, dimana dalam didalamnya terdapat algoritma yang berhubungan dengan proses pengenalan wajah yaitu algoritma *Local Binary Pattern Histogram* (LBPH). Tujuan dirancangnya aplikasi ini diharapkan dapat meningkatkan tingkat fleksibilitas dan efisiensi dari sistem presensi yang sudah ada karena tidak perlu berulang kali mengambil formulir daftar kehadiran, dapat mengurangi kemungkinan terjadinya kecurangan saat proses presensi serta dapat mengurangi penggunaan kertas atau *paperless* dalam pelaksanaan pencatatan kehadiran mahasiswa khususnya pada Program Studi Teknik Informatika Universitas Islam Raden Rahmat Malang.

## TINJAUAN PUSTAKA

### *Open Computer Vision* (OpenCV)

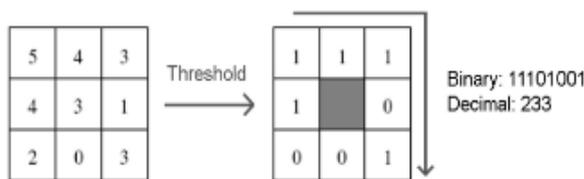
Menurut (Kadir, 2019), *OpenCV* adalah sebuah pustaka yang berbasis sumber terbuka yang memiliki lebih dari 500 fungsi yang dirancang khusus untuk memproses visi komputer. *Software* ini telah dirilis dengan lisensi *BSD* yang memungkinkan penggunaannya baik untuk kepentingan bisnis maupun komersial. Selain itu, *OpenCV* juga dapat dijalankan di berbagai platform seperti *Windows*, *Linux*, *Mac OS*, *iOS*, dan *Android*.

### *Local Binary Pattern Histogram*

Menurut (Kurniawan, 2017), *Local Binary Pattern Histogram* atau LBPH merupakan inovasi terbaru dari metode *Local Binary Pattern* (LBP) yang bertujuan untuk meningkatkan kinerja dalam pengenalan wajah. LBP sendiri merupakan deskriptor tekstur yang dapat digunakan untuk merepresentasikan wajah, karena wajah dapat dianggap sebagai kumpulan pola tekstur mikro yang terdapat dalam citra. LBPH merupakan suatu pendekatan non-parametrik yang menggambarkan pola tekstur lokal dalam citra wajah. Metode *Local Binary Pattern Histogram* (LBPH)

memiliki proses pengenalan yang mirip dengan metode *Local Binary Pattern* (LBP).

- 1) Pada tahap awal, gambar wajah yang telah dipotong dan dikonversi menjadi skala keabuan menjalani proses ekstraksi fitur. Proses ini melibatkan perubahan setiap piksel gambar menjadi nilai *thresholding* menggunakan 3x3- tetangga, di mana piksel tengah dari setiap 3x3- tetangga diambil sebagai pembanding.
- 2) Apabila nilai piksel tetangga memiliki nilai yang lebih besar atau sama dengan nilai piksel pembanding dalam skala keabuan, maka nilai piksel tersebut akan dikonversi menjadi 1. Sebaliknya, jika nilai piksel tetangga memiliki nilai yang lebih kecil dari nilai piksel pembanding dalam skala keabuan, maka nilai piksel tersebut akan dikonversi menjadi 0.
- 3) Setelah mengkonversi menjadi nilai biner, akan terbentuk deret biner yang nantinya dapat diubah menjadi nilai desimal. Nilai desimal ini akan digunakan sebagai nilai tengah untuk piksel selanjutnya. Deret biner ini dikenal sebagai kode LBP. Sebagai contoh, dalam Gambar 1, nilai desimal 233 akan menjadi nilai tengah untuk piksel selanjutnya



Sumber: (Kurniawan, 2017)

Gambar 1. Kode *Local Binary Pattern* (LBP)

### Android

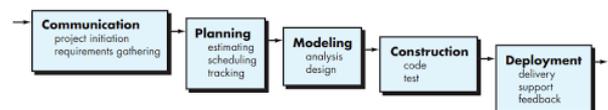
Android merupakan sebuah sistem operasi yang menggunakan dasar Linux yang telah diadaptasi khusus untuk digunakan pada perangkat bergerak, seperti perangkat mobile. Sistem operasi ini terdiri dari beberapa komponen penting, termasuk sistem operasi inti, *middleware*, dan berbagai aplikasi utama. Pada awalnya, Android dikembangkan oleh perusahaan bernama Android Inc. Namun, pada tahun 2005, perusahaan tersebut diakuisisi oleh Google. Aplikasi Android dapat ditulis menggunakan bahasa pemrograman Java. Umumnya, struktur Android terdiri dari komponen perangkat lunak yang terdiri dari lapisan aplikasi, lapisan kerangka kerja, Android runtime sebagai perantara dan kernel Linux yang mengelola perangkat keras di bawahnya (Juhara, 2016).

### METODE PENELITIAN

Metode penelitian ini mengulas tentang langkah-langkah yang diterapkan dalam penelitian. Tempat penelitian ini dilakukan di Program Studi Teknik Informatika Universitas Islam Raden Rahmat Malang dan waktu penelitian dilaksanakan pada semester genap yang dibagi menjadi dua periode, yaitu pada tanggal 8 -12 Mei 2023 yang menghasilkan 8 responden dan 13 Mei 2013 yang menghasilkan 2 responden.

Metode yang tepat untuk penelitian ini adalah metode penelitian dan pengembangan atau lebih dikenal dengan istilah *Research and Development* (R&D). Menurut (Sugiyono, 2017), Metode penelitian dan pengembangan, yang juga dikenal sebagai *Research and Development* dalam bahasa Inggris, adalah suatu metode penelitian yang bertujuan untuk menciptakan produk spesifik dan menguji tingkat efektivitas produk tersebut.

Metode pengembangan sistem yang dijadikan sebagai acuan dalam penelitian ini adalah metode *waterfall*. Model *waterfall* adalah sebuah pendekatan klasik yang berorientasi pada sistem dan mengikuti urutan yang terstruktur dalam proses pengembangan perangkat lunak (Pressman, 2012). Tahapan pengembangan sistem dengan metode *waterfall* ditampilkan pada gambar 2. Berdasarkan gambar 2, tahapan penelitian ini diawali dengan analisis kebutuhan sistem, kemudian menentukan desain sistem yang akan dibangun, dilanjutkan dengan penulisan kode program, tahap pengujian sistem.



Sumber: (Pressman, 2012)

Gambar 2. Tahapan Metode Pengembangan *Waterfall*

### Analisis Kebutuhan

Tahap pertama analisa kebutuhan yang dilakukan adalah dengan mengumpulkan data citra wajah mahasiswa. Dalam penelitian ini jumlah data mahasiswa yang diambil sebanyak 10 responden dengan pengambilan data wajah sejumlah masing-masing 2 citra, sehingga total citra wajah yang didapatkan adalah sejumlah 20 citra. Data citra wajah ini diperoleh pada saat mahasiswa melakukan pendaftaran wajah secara *real-time* pada aplikasi yang kemudian tersimpan pada *dataset* yang ditampilkan seperti gambar 3.

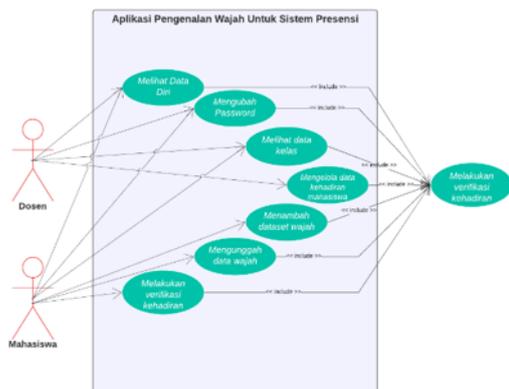


Gambar 3. Sample Pengambilan Wajah

Foto tersebut yang kemudian akan dijadikan sebagai tolak ukur keberhasilan dalam proses pengenalan wajah. Jika citra wajah yang diambil saat pengenalan wajah cocok dengan data tersebut atau data foto yang berada di *database*, maka proses pengenalan wajah berhasil.

**Perancangan Desain Sistem**

Setelah melakukan analisis kebutuhan, langkah selanjutnya adalah melakukan tahap desain sistem. Pada tahap ini, peneliti melakukan perancangan sistem dengan menggunakan *diagram use case* untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem yang dapat dilihat pada Gambar 4

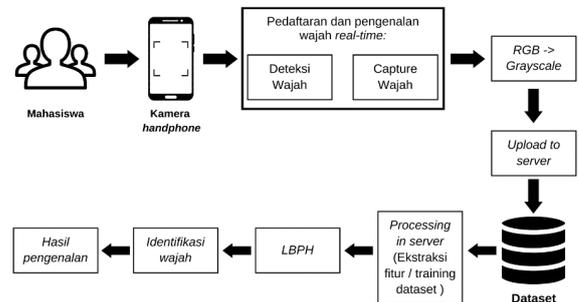


Gambar 4. Use Case Diagram

Grafik *use case* tersebut menerangkan bahwa pada aplikasi terdapat dua *user* yaitu dosen dan mahasiswa. Agar dapat melihat data diri, mengubah *password*, melihat data kelas yang diajar dan melakukan verifikasi kehadiran mahasiswa, dosen harus melakukan *login* terlebih dahulu. Begitu juga dengan mahasiswa, agar dapat menambah *dataset* wajah yang akan digunakan ketika proses pengenalan wajah saat presensi dilakukan dan mengunggah data wajah tersebut, kemudian melihat data diri, mengubah *password*, melihat data kelas serta melakukan verifikasi kehadiran, mahasiswa harus melakukan *login* dahulu

**HASIL DAN PEMBAHASAN**

Bagian ini membahas mengenai proses uji coba terhadap kinerja aplikasi berkaitan dengan data perkuliahan yang ditampilkan dan ketepatan sistem dalam melakukan pengenalan wajah yang dilakukan oleh mahasiswa.



Gambar 5. Alur Kerja Pengenalan Wajah

Gambar 5 merupakan alur kerja pengenalan wajah. Sistem presensi dengan pengenalan wajah diawali dengan mahasiswa menjalankan program hingga menampilkan kamera kemudian jika kamera berhasil mendeteksi wajah, maka akan di *capture* kemudian dikonversi dari RGB – *grayscale*. Setelah itu data citra wajah mahasiswa akan disimpan pada *dataset* dengan format nama *nim\_indexfoto*, *nim* didapatkan dari mahasiswa yang telah *login*. Selanjutnya dilakukan proses latih data wajah dengan menyimpan *file* menjadi *\*.yml*. Kemudian diproses dengan algoritma *Local Binary Pattern Histogram* untuk dapat dilakukan identifikasi wajah. Jika wajah yang terdeteksi adalah benar wajah mahasiswa yang bersangkutan maka akan muncul notifikasi yang menampilkan nama mahasiswa telah berhasil melakukan proses presensi.

**Batasan Pengujian Aplikasi**

Batasan pengujian aplikasi yang digunakan untuk sistem presensi ini adalah sebagai berikut:

1. Data citra wajah disimpan ke dalam satu *folder* hasil pendaftaran dan satu *folder* saat proses pengenalan wajah
2. Proses pedeteksi wajah dilakukan secara *real-time* dengan posisi wajah sejajar dan tegak lurus menghadap ke kamera
3. Proses pengujian aplikasi dilakukan di tempat kerja penulis dikarenakan keterbatasan waktu dan tempat pengujian. Dengan menggunakan kelas skenario yang terdiri dari 8 orang yang bertindak sebagai mahasiswa
4. Aplikasi ini hanya melakukan pengambilan citra wajah, kemudian dilakukan proses *grayscale*. Dilanjutkan dengan diproses menggunakan *library OpenCV* dan kemudian disimpan kedalam folder

5. Mahasiswa memilih satu matakuliah untuk melakukan proses presensi menggunakan pengenalan wajah

Rincian mengenai perangkat keras dan perangkat lunak yang digunakan dalam proses implementasi perangkat lunak ini dipaparkan dalam Tabel 1 di bawah ini:

**Tabel 1.** Spesifikasi Perangkat Keras Dan Perangkat Lunak Yang Digunakan

<b>Perangkat Keras (Hardware)</b>	<i>Android</i>	Redmi Note 10s
<b>Perangkat Lunak (Software)</b>	Sistem Operasi	MIUI 12.5 berbasis <i>Android 11</i>
	<i>Processor</i>	Media Tek Helio G95
	Memori Primer	6 GB
	Memori Sekunder	64 GB

Aplikasi Sistem Presensi Menggunakan Pengenalan Wajah Berbasis *Android* ini dikembangkan dengan bahasa pemrograman Javascript menggunakan *framework* React Native dan menggunakan *database* MySQL. Aplikasi ini disimpan pada *web server* yang menggunakan python dengan *framework* Flask sebagai perantaranya.

Implementasi Aplikasi Sistem Presensi Menggunakan Pengenalan Wajah Berbasis *Android* ini menggunakan spesifikasi perangkat keras dan perangkat lunak yang dijabarkan pada Tabel 4.1. Selain menggunakan spesifikasi tersebut, aplikasi ini juga dapat dijalankan pada perangkat *mobile* lain dengan minimal versi *Android 5* (Lollipop)

**Pengujian Aplikasi**

Hal pertama yang dilakukan pada saat pengujian aplikasi adalah dengan melakukan *login*. Pengujian dilakukan dengan memasukkan *username* dan *password* dengan ketentuan *username* dan *password* bagi mahasiswa dengan menggunakan nim dan bagi dosen dapat menggunakan nidn.

Setelah *user* berhasil melakukan *login*. *User* akan diarahkan menuju halaman beranda atau *home*. Terdapat perbedaan menu pada halaman beranda mahasiswa dan dosen. Dimana pengenalan wajah hanya dapat dilakukan oleh mahasiswa seperti pada Gambar 6 dan halaman dosen dapat dilihat pada Gambar 7



**Gambar 6** Home Mahasiswa

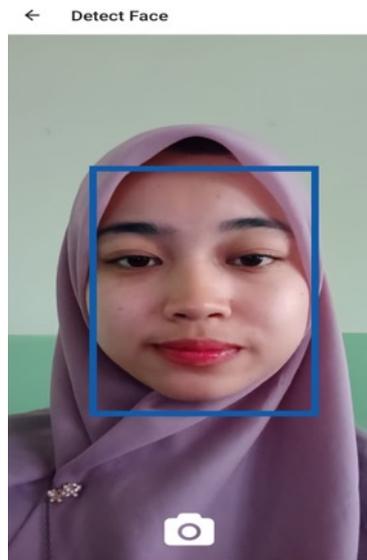


**Gambar 7** Home Dosen

Ketika mahasiswa memilih menu “Daftarkan wajah”, sistem akan menampilkan kamera untuk mendeteksi wajah. Jika wajah terdeteksi kemudian sistem akan mengambil 2 (dua) kali foto mahasiswa kemudian akan dilakukan proses *grayscale* dan disimpan ke dalam *dataset* dan *database* berdasarkan

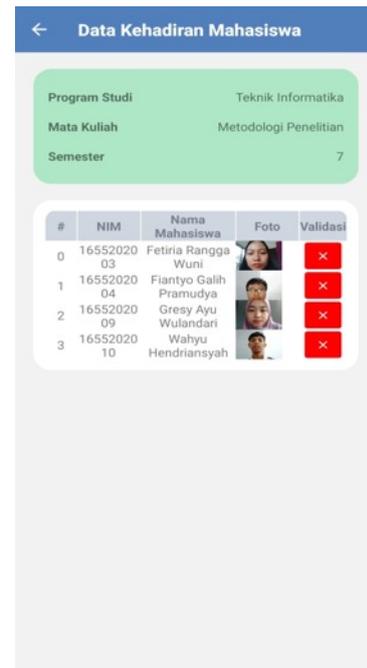
nim mahasiswa. Proses pendaftaran wajah mahasiswa dapat dilihat pada Gambar 8

mahasiswa beserta citra wajah mahasiswa yang melakukan presensi yang terlihat seperti Gambar 10.



Gambar 8 Halaman Pendaftaran Wajah

Setelah data citra tersebut berhasil di *upload* dan dilatih, mahasiswa dapat memilih kelas berdasarkan matakuliah yang akan diambil untuk melakukan proses verifikasi data kehadiran menggunakan pengenalan wajah.



Gambar 10 Halaman Tampil Data Mahasiswa Yang Hadir

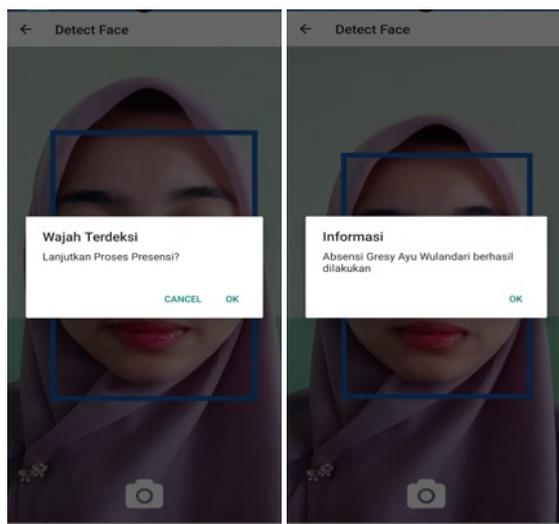
Hasil perhitungan dari uji coba program yang telah dilakukan dengan menggunakan beberapa sampel, ditampilkan pada Tabel 2 berikut:

Tabel 2. Akumulasi dari Uji Coba Produk Yang Dilakukan

Parameter	Hasil
Banyaknya Citra <i>Testing</i>	10
Banyaknya Citra <i>Testing</i> Yang Dikenali	7
Banyaknya Citra <i>Testing</i> Yang Gagal Dikenali	3
Akurasi	70%
<i>Error Rate</i>	30%

Penjelasan Tabel 2 adalah sebagai berikut:

1. Tingkat Akurasi  
 Tingkat akurasi yang diperoleh dari uji coba produk ini yaitu 70%, yang berarti dari total 10 wajah yang telah didaftarkan terdapat 7 citra wajah yang berhasil dikenali.
2. *Error Rate*  
 Tingkat kesalahan yang diperoleh dari uji coba produk ini yaitu 30%, yang berarti dari total 10 wajah yang telah didaftarkan terdapat 3 citra wajah yang tidak dikenali dengan benar.



Gambar 9. Antarmuka Mahasiswa Yang Berhasil Dikenali

Gambar 9 menampilkan keterangan bahwa data mahasiswa yang melakukan proses presensi berhasil dideteksi dan dikenali oleh sistem. Citra wajah mahasiswa yang diambil ketika melakukan proses presensi akan ditampilkan pada halaman dosen untuk dilakukan verifikasi ulang yang menampilkan data

Proses uji coba program dilaksanakan dengan menggunakan sejumlah 10 data responden yang berasal dari mahasiswa. Dengan metode deteksi wajah menggunakan *local binary pattern histogram* dan *library OpenCv* dan dengan kondisi pengujian yang telah direncanakan sebelumnya, didapatkan hasil bahwa 7 data wajah berhasil dideteksi dan dikenali dengan benar dan 3 data wajah masih sering dideteksi sebagai orang lain.

Hal ini disebabkan karena tidak adanya proses *cropping* pada saat proses penyimpanan citra wajah. Sehingga ditentukan adanya jarak ideal agar wajah terdeteksi dan dapat dilakukan pengenalan wajah sejauh 25cm hingga 30cm dengan posisi wajah yang sejajar dan tegak lurus agar dapat dikenali dengan benar. Dan dengan beberapa kali percobaan, mahasiswa yang sebelumnya gagal dikenali selanjutnya telah berhasil dikenali dengan benar.

## KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, aplikasi pengenalan wajah untuk sistem presensi yang diberi nama "SiPMa" ini telah berhasil diimplementasikan dengan cukup lancar dan sesuai dengan yang diharapkan dengan tingkat keberhasilan 70%. Hal ini membuktikan bahwa metode *Local Binary Pattern Histogram* dan pustaka *OpenCV* dapat bekerja dengan cukup baik. Pada proses pendeteksian dan pengenalan wajah mahasiswa menggunakan aplikasi ini tidak menggunakan proses *cropping* sehingga untuk mendapatkan tingkat akurasi yang baik, proses deteksi dan pengenalan dapat dilakukan dengan jarak ideal yaitu mulai 25 – 30cm dengan posisi yang sejajar dan tegak lurus dengan kamera.

## DAFTAR PUSTAKA

- Clara, C. (2022). *Implementasi Algoritma Local Binary Pattern Dan Backpropagation Untuk Face Recognition*. Universitas Multimedia Nusantara.
- Hilmawan, S. V. (2022). *Implementasi Face Recognition Secara Real-Time Dengan Metode Haar Cascade Classifier Menggunakan Opencv-Python*. Universitas Gunadarma.
- Juhara, Z. P. (2016). *Panduan Lengkap Pemrograman Android*. Yogyakarta: Andi Offset.
- Kadir, A. (2019). *Langkah Mudah Pemrograman Opencv Dan Python*. Jakarta: PT Elex Media Komputindo.
- Kosasih, R., & Daomara, C. (2021). Pengenalan Wajah dengan Menggunakan Metode Local Binary Patterns Histograms (LBPH). *Jurnal Media Informatika Budidarma*, 5(4), 1258. <https://doi.org/10.30865/mib.v5i4.3171>
- Kurniawan, A. (2017). *Perbandingan Akurasi*

- Pengenalan Wajah Menggunakan Metode LBPH Dan Eigenface Dalam Mengenali Tiga Wajah Sekaligus Secara Real-Time*. Universitas Islam Negeri Sultan Syarif Kasim Riau.
- Pratama, Lusni, T., Dara, M. A. D. W., & Sahertian, J. (2021). Perbandingan Pengenalan Wajah Dengan Metode Local Binary Pattern Histogram Dan Eigenface Untuk Presensi. In *Seminar Nasional Inovasi Teknologi*.
- Pressman, R. S. (2012). *Rekayasa Perangkat Lunak (Pendekatan Praktisi)* (1st ed.). Yogyakarta: Andi Offset.
- Republik Indonesia. (2012). *Undang-Undang Nomor 12 Tahun 2012 tentang Pendidikan Tinggi*. Jakarta: Sekretariat Negara.
- Sugiyono. (2017). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung: Alfabet.