

Prototipe Pendeteksi dan Pengenalan Wajah Berbasis Web Menggunakan Algoritma *Local Binary Pattern Histogram* untuk Absensi

Ita Augustina Tarigan¹, Asep Kurniawan²

Teknik Informatika dan Komputer, Politeknik Negeri Jakarta

Depok, Jawa Barat

itatarigan99@gmail.com, asep.kurniawan@tik.pnj.ac.id

Diterima: 13 Juni 2022. Disetujui: 25 Oktober 2022. Dipublikasikan: 5 November 2022.

Abstract-Face Detection and Recognition Prototype for Attendance is a prototype used to perform attendance by using a face as an object. The attendance system, which usually uses fingerprints as an object, has become less effective given the Covid-19 pandemic in Indonesia. Attendance tools that have been contaminated by fingers will become a container for the spread of Covid-19. The design of this attendance system aims to break the chain of spread of COVID-19 by conducting attendance without being directly contaminated with attendance devices. In this study, facial detection and recognition analysis was carried out with the Local Binary Pattern Histogram (LBPH) algorithm feature approach for web-based attendance, where software development used the prototype method. LBPH is an algorithm combination between Local Binary Pattern (LBP) and Histogram of Oriented Gradients (HOG). LBP is a well-known method of recognizing an object by distinguishing the object from the background. Face images taken in real time using a webcam will be compared and matched using histograms that have been extracted with facial images that already exist in the database. The results of the facial recognition process in this system are very dependent on lighting, facial angles and the use of accessories on the face. The face attendance system is able to detect faces well up to 100% accuracy and recognize faces that have been detected up to 60% accuracy.

Keywords : attendance, face image, covid-19, Local Binary Pattern Histogram, face detection and recognition,.

Abstrak-Prototipe Pendeteksi dan Pengenalan Wajah untuk Absensi adalah *prototipe* yang digunakan untuk melakukan absensi dengan menggunakan wajah sebagai objek. Sistem absensi yang biasanya menggunakan sidik jari sebagai objek menjadi kurang efektif mengingat adanya pandemi *Covid-19* di Indonesia. Alat absensi yang sudah terkontaminasi oleh jari akan menjadi wadah penyebaran *Covid-19*. Perancangan sistem absensi ini bertujuan untuk pemutusan rantai penyebaran *COVID-19* dengan melakukan absensi tanpa terkontaminasi langsung dengan alat absensi. Pada penelitian ini dilakukan Analisa pendeteksi dan pengenalan wajah dengan pendekatan fitur algoritma *Local Binary Pattern Histogram* (LBPH) untuk absensi berbasis web, di mana pengembangan perangkat lunak menggunakan metode prototipe. LBPH adalah sebuah kombinasi algoritma antara *Local Binary Pattern* (LBP) dengan *Histogram of Oriented Gradients* (HOG). LBP merupakan metode yang terkenal dalam mengenali sebuah objek dengan cara membedakan objek dengan *background*. Citra wajah yang diambil secara *realtime* menggunakan *webcam* akan dibandingkan dan dicocokkan menggunakan histogram yang sudah diekstraksi dengan citra wajah yang sudah ada pada database. Hasil proses pengenalan wajah pada sistem ini sangat bergantung pada pencahayaan, sudut pandang wajah dan pemakaian aksesoris pada wajah. Sistem absensi wajah mampu mendeteksi adanya wajah dengan baik sampai akurasi 100% dan mengenali wajah yang telah terdeteksi sampai akurasi 60%.

Kata Kunci : absensi, citra wajah, covid-19, Local Binary Pattern Histogram, pendeteksi dan pengenalan wajah.

I. PENDAHULUAN

Diawali dengan munculnya *COVID-19* yang menyebar dengan sangat cepat. Proses penyebaran terjadi melalui droplet dan kontak dengan virus. Kontak dengan virus dapat terjadi jika bersentuhan dengan orang yang terinfeksi virus atau terkontaminasi oleh barang yang telah disentuh orang tersebut. Kegiatan absensi adalah aktivitas yang tidak pernah lepas baik dalam dunia pekerjaan maupun pendidikan. Alat absensi telah mengalami kemajuan menyesuaikan dengan perkembangan teknologi pada era sekarang ini, di mana proses absensi secara manual

sudah lama ditinggalkan karena dapat berakibat terjadi kecurangan pada data. Alat absensi yang banyak digunakan adalah alat absensi menggunakan *finger print* sebagai objek absen. Akan tetapi dengan kondisi pandemi *COVID-19* menyebabkan kegiatan absensi menggunakan *finger print* menjadi tidak efektif karena jika alat absensi digunakan oleh lebih dari 1 orang makan akan menjadi media penyebaran *COVID-19*. Oleh karena itu dibutuhkan alat absensi di mana proses absensi dapat dilakukan tanpa terkontaminasi langsung dengan alat absensi. Alat absensi menggunakan wajah sebagai objek adalah sistem

usulan pada penelitian ini. Di mana *user* dapat melakukan absensi tanpa terkontaminasi langsung dengan alat sehingga meminimalisir penyebaran COVID-19.[1]

Berdasarkan masalah yang sudah dipaparkan sebelumnya dan dengan jurnal penelitian yang sudah ada sebagai referensi, peneliti merancang sebuah prototipe sistem absensi yang menggunakan wajah sebagai objek absensi sehingga tidak akan ada alat yang terkontaminasi langsung oleh anggota tubuh. Pada penelitian ini dilakukan Analisa pengenalan wajah dengan pendekatan fitur algoritma Local Binary Pattern Histogram di mana pengembangan perangkat lunak menggunakan metode Prototype. Sistem ini dibuat dengan Bahasa pemrograman Python 3, Pustaka OpenCV 3 dan framework Flask. Sistem ini berfungsi untuk melakukan pengenalan wajah secara otomatis berdasarkan basis data wajah yang telah dibuat sebelumnya sebagai acuan. Hasil proses pengenalan wajah pada sistem ini sangat bergantung pada pencahayaan, sudut pandang wajah, dan pemakaian aksesoris pada wajah.

II. TINJAUAN PUSTAKA

A. Absensi

Absensi adalah hal yang paling penting dalam kehidupan sehari-hari sebagai salah satu cara pencatatan kehadiran seseorang baik pada lingkup tempat kerja, sekolah, maupun kampus[2].

B. Prototype

Prototype adalah versi awal dari sistem perangkat lunak yang digunakan untuk mendemonstrasikan konsep-konsep, percobaan rancangan, dan menemukan lebih banyak masalah dan solusi yang memungkinkan. sistem dengan model *prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik [3].

C. Python

Python merupakan bahasa pemrograman model skrip (*scripting language*) yang berorientasi objek. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi. Python merupakan bahasa pemrograman yang *freeware* atau perangkat bebas dalam arti sebenarnya, tidak ada batasan dalam penyalinannya atau mendistribusikannya. Lengkap dengan *source codenya*, *debugger* dan *profiler*, antarmuka yang terkandung di dalamnya untuk pelayanan antarmuka, fungsi sistem, GUI (antarmuka pengguna grafis), dan basis datanya[4].

D. OpenCV

OpenCV (Open Computer Vision Library) merupakan *library* perangkat lunak *open source* yang memiliki lisensi *BSD-licensed product*. OpenCV memiliki lebih dari 2500 algoritma yang telah di optimasi dan disediakan untuk menangani hal mengenai *computer vision* dan *machine learning*. Algoritma yang ada dapat digunakan untuk mendeteksi wajah, mengenali wajah, mengidentifikasi objek, dan lain-lain. Di dalam OpenCV telah tersedia tiga algoritma pengenalan wajah, diantaranya *Eigenface*, *Fisherface* dan *Local Binary Pattern Histogram (LBPH)*[5].

E. Local Binary Pattern Histogram (LBPH)

LBPH merupakan metode yang disukai dalam *computer vision*, *image processing*, dan *pattern recognition*, ini sesuai untuk ekstraksi fitur karena menggambarkan tekstur dan struktur gambar[6]. Cara kerja algoritma *Local Binary Pattern Histogram (LBPH)* adalah dengan membandingkan nilai piksel pada pusat citra dengan 8 nilai piksel di sekelilingnya. Misal pada sebuah citra berukuran 3x3, nilai biner pada pusat citra dibandingkan dengan nilai sekelilingnya. Dengan cara mengurangi nilai piksel pada pusat citra dengan nilai piksel di sekelilingnya, jika hasilnya lebih atau sama dengan 0 maka diberi nilai 1 dan jika hasilnya kurang dari 0 maka diberi nilai 0. Nilai 0 dan 1 (8 nilai biner) Setelah itu, disusun searah jarum jam. Dari susunan 8 nilai biner tersebut dikonversi menjadi nilai desimal [7]. Algoritma LBPH merupakan algoritma yang paling akurat dan efisien dibandingkan dengan algoritma *Eigenface* dan *Fisherface*. Algoritma LBPH memiliki rata-rata akurasi sebesar 83%. Sedangkan, algoritma *Eigenface* memiliki rata-rata akurasi sebesar 46% dan algoritma *Fisherface* sebesar 54%[8].

F. Unified Modelling Language (UML)

Unified Modelling Language merupakan alat perancangan sistem yang berorientasi pada objek. UML diagram memiliki tujuan utama untuk membantu tim pengembangan proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program. UML mempunyai tiga kategori utama yaitu struktur diagram, behaviour diagram dan interaction diagram. Di mana masing-masing kategori tersebut memiliki diagram yang menjelaskan arsitektur sistem dan saling terintegrasi[9]. Beberapa jenis diagram UML yang digunakan pada penelitian ini:

1. Use Case Diagram yang menggambarkan *external view* dari sistem yang akan dibuat modelnya [10].
2. Diagram activity menunjukkan aktivitas sistem dalam bentuk kumpulan aksi-aksi, bagaimana masing-masing aksi tersebut dimulai, keputusan

yang mungkin terjadi hingga berakhirnya aksi. *Activity diagram* juga dapat menggambarkan proses lebih dari satu aksi dalam waktu bersamaan[11]

3. *Class Diagram*, kelas sebagai suatu set objek yang memiliki atribut dan perilaku yang sama[12].

III. RANCANGAN & PENELITIAN

Pada penelitian ini proses absensi dilakukan dengan pendeteksian dan pengenalan wajah. Metode yang digunakan pada penelitian ini yaitu: 1) Analisa Kebutuhan, 2) Pengumpulan data, 3) Perancangan, 4) Tahapan Penelitian. Berikut penjelasan langkah dalam metode penelitian.

A. Analisa Kebutuhan

Analisis kebutuhan dalam perancangan sistem terbagi menjadi 3, yaitu :

1. Analisa Kebutuhan User

95,1% *user* mengetahui dan pernah menggunakan alat absensi menggunakan *Finger Print* dan sebanyak 81,7% merasa alat absensi *finger print* tidak aman digunakan disaat pandemi ini dan memilih setuju untuk menggunakan alat absensi menggunakan pendeteksi dan pengenalan wajah untuk mengurangi angka penyebaran *covid-19* dengan tidak bersentuhan langsung dengan alat absensi.

2. Kebutuhan Antarmuka

-Aplikasi harus mampu mendeteksi dan mengenali wajah *user* yang menjadi sorotan kamera

-Aplikasi harus mampu menampilkan informasi berupa nama *user* tersebut yang selanjutnya menjadi referensi untuk absensi

-Aplikasi harus mampu menyimpan citra yang telah diambil beserta dengan waktu kegiatan tangkap citra sebagai referensi kegiatan (*log*) untuk *user* tersebut.

3. Kebutuhan Fungsional

-Mampu memanfaatkan dan mengatur citra yang diinput oleh proses tangkap citra *user* dengan benar dan efisien

-Memberikan output berupa informasi nama *user* sebagai tanda dia terdaftar pada basis data.

B. Pengumpulan Data

1. Citra wajah. Data citra wajah yang dapat digunakan sebagai referensi untuk absensi.

2. *Classifier Cascade (Frontal Face)* digunakan sebagai referensi fitur wajah yang digunakan oleh algoritma LBPH untuk melakukan deteksi fitur wajah(tempat pengolahan dataset dan mempelajarinya hingga gambar dikenal).

3. *Training Set* digunakan sebagai referensi identitas pemilik wajah dengan citra wajah sehingga dapat dilakukan identifikasi kepemilikan citra wajah.

C. Perancangan Sistem

Prototipe pendeteksi dan pengenalan wajah untuk absensi dirancang dengan menggunakan metode *prototyping*. Cara kerja prototipe yang dirancang dapat dilihat melalui Gambar 1.

1. Flowchart Sistem Absensi

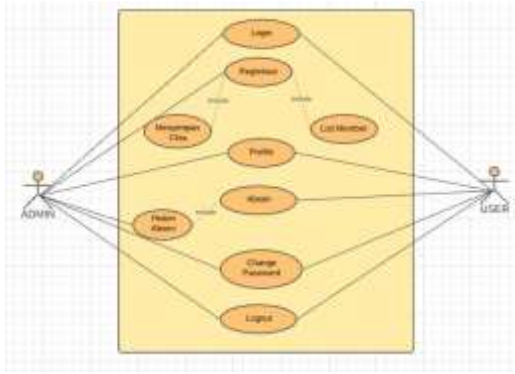


Gambar 1. Flowchart Sistem Absensi

Saat *user* melakukan absensi maka *user* hanya perlu mengarahkan wajah ke arah kamera lalu webcam akan menangkap citra dan melakukan proses pendeteksian citra. Setelah citra di deteksi maka sistem akan melakukan proses pengenalan citra, jika citra wajah dikenali maka akan muncul identitas pemilik wajah lalu proses absensi berhasil dilakukan dan sistem akan secara otomatis menyimpan data aktivitas absensi.

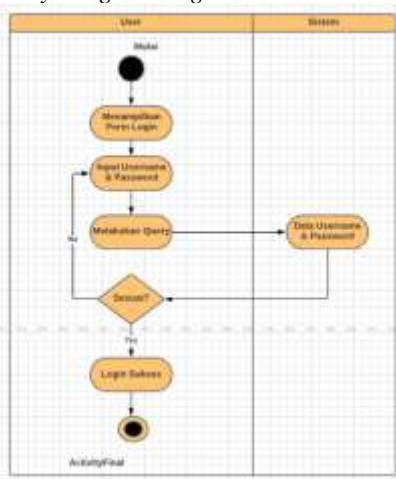
2. Use Case Diagram

Pada Gambar 2 ditampilkan proses sistem usulan yang diimplementasikan pada studi kasus. Admin melakukan proses *login* untuk menambahkan *User* baru. Admin dapat melihat daftar *user*, admin dan *user* dapat mengakses halaman profil dan me-*reset password* masing-masing.



Gambar 2. Use Case Diagram

3. Activity Diagram Login



Gambar 3. Activity Diagram Login

Pada Gambar 3, admin dan *user* melakukan proses *login* dengan mengakses halaman *login* dan menginput *username* dan *password* lalu sistem akan melakukan pengecekan terhadap data di database, jika data input benar maka akan masuk ke halaman utama.

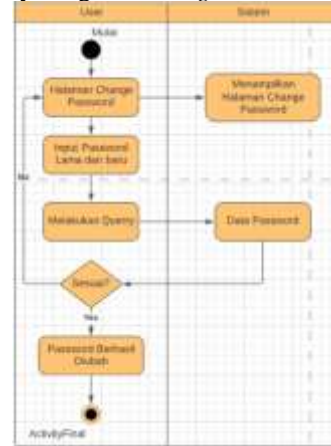
4. Activity Diagram Register



Gambar 4. Activity Diagram Register

Pada Gambar 4, proses register dilakukan secara manual oleh admin. Admin akan memberikan *username* dan *password user* yang telah berhasil register.

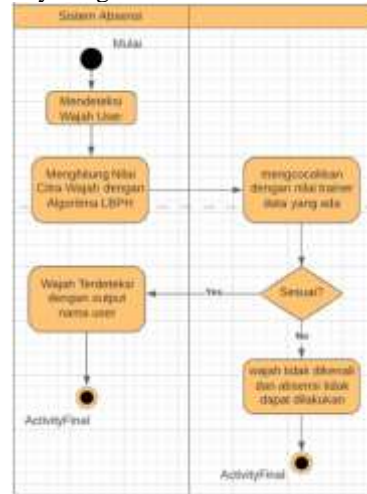
5. Activity Diagram Change Password



Gambar 5. Activity Diagram Change Password

Pada Gambar 5, admin dan *user* dapat merubah *password* pada halaman *Change Password* dengan menginput *password* sebelumnya dan *password* baru yang akan dipakai

6. Activity Diagram Absensi

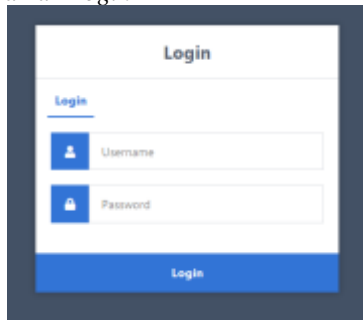


Gambar 6. Activity Diagram Absensi

Pada Gambar 6 terdapat halaman absensi, sistem akan menyort wajah *user* dan menghitung nilai *confidence* dan mencocokkan hasil perhitungan dengan data hasil *training* yang sudah ada dan jika sesuai maka absensi berhasil dilakukan.

7. Implementasi Sistem

- Halaman Login



Gambar 7. Halaman Login

Pada Gambar 7, terdapat halaman *login*, halaman ini dibuat sebagai pemberian hak akses untuk admin dan *user* mengakses website absensi.

- Halaman Dashboard



Gambar 8. Halaman Dashboard Admin

Pada Gambar 8, terdapat halaman Dashboard Admin yang terdapat fitur *Recognizer*, *Register* dan *Member* yang hanya dapat diakses oleh Admin dan bukan *User*.



Gambar 9. Halaman Dashboard User

Gambar 9 menampilkan halaman *Dashboard User* yang memiliki fitur berbeda dengan halaman *Dashboard Admin*

- Halaman Recognizer



Gambar 10. Halaman Recognizer

Pada Gambar 10 ditampilkan halaman *Recognizer*, yang hanya dapat diakses oleh Admin, di mana admin harus membuka halaman ini terlebih dahulu untuk menjalankan sistem absensi.

- Halaman Register



Gambar 11. Halaman Register

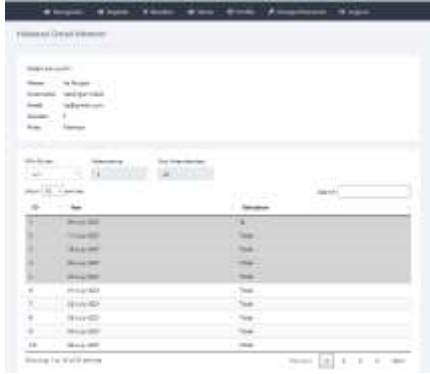
Pada Gambar 11, terdapat halaman *Register* hanya dapat di akses oleh Admin. *User* yang akan melakukan register akan didaftarkan secara manual oleh admin. Admin akan mengisi data *user* yang akan registrasi sesuai dengan info data yang diberikan *user*.

- Halaman Member



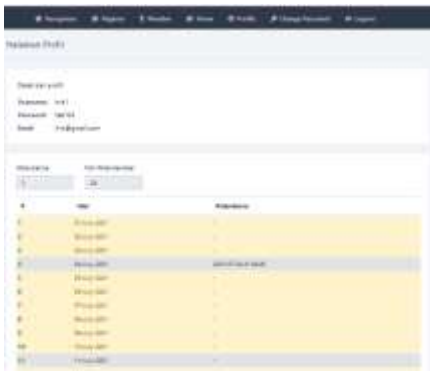
Gambar 12. Halaman Member

Pada gambar 12 terdapat halaman member. Pada halaman ini admin dapat melihat seluruh anggota yang menggunakan sistem absensi ini, baik itu *user* maupun admin. Admin juga dapat melihat hasil rekap data absensi *user* seperti pada gambar 13.



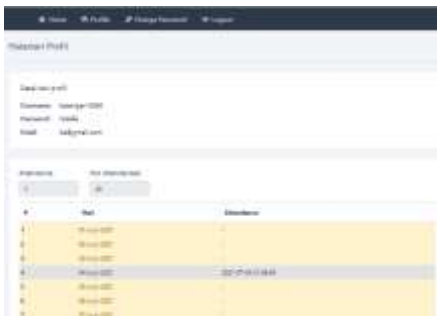
Gambar 13. Rekap Data Absen

- Halaman *Profile*



Gambar 14. Halaman Profil Admin

Pada gambar 14 ditampilkan halaman *Profile Admin*. Admin dapat melihat *username*, *password*, dan email admin itu sendiri. Admin juga dapat melihat seluruh data absensi *user*. Tetapi Admin tidak bisa mengetahui daftar nama *user* yang sudah melakukan absensi pada halaman profil, hanya jumlah kehadiran yang menjadi patokan data absensi bagi admin di halaman profil.



Gambar 15. Halaman Profil *User*

Pada Gambar 15 terdapat halaman *profile user*. *user* dapat melihat *username*, *password*, dan email *user* itu sendiri dan hanya dapat melihat histori absensinya sendiri.

- Halaman *Change Password*



Gambar 16. Halaman *Change Password*

Pada gambar 16 terdapat halaman *Change Password*, tidak ada perbedaan antara halaman yang dapat diakses admin maupun *user*. Saat akan merubah *password user* dan admin akan diminta untuk menginput *password* lama dan menginput *password* baru serta mengkonfirmasi sekali lagi *password* baru yang akan dibuat.

D. Tahapan Penelitian

Tahapan penelitian menggunakan metode *Local Binary Pattern Histogram* (LBPH). Cara kerja prototipe yang dirancang yaitu ketika *user* ingin melakukan absensi maka *user* hanya perlu mengarahkan wajah ke depan webcam yang disediakan, lalu webcam akan menangkap citra dan melakukan proses pendeteksian citra. Setelah citra di deteksi maka sistem akan melakukan proses pengenalan citra, jika citra wajah dikenali maka akan muncul identitas pemilik wajah lalu proses absensi telah berhasil dilakukan dan sistem akan secara otomatis menyimpan data aktivitas absensi. Tetapi jika citra tidak dikenali maka berarti *user* belum meregistrasikan wajahnya. Proses registrasi dilakukan manual oleh admin dan setelah proses registrasi selesai sistem akan menyimpan data wajah baru kedalam database dan *user* yang sudah registrasi bisa melakukan absensi.

1. Dataset

Terdapat dua hal penting untuk keberhasilan pengenalan wajah pada aplikasi, yaitu menyediakan *Cascade Classifier* yang sudah terdapat di OpenCV dan citra dataset yang sudah dilakukan training terlebih dahulu. Proses pengenalan citra wajah berhasil dilakukan Ketika sample citra wajah tersebut telah di training oleh sistem. Pengambil sampel citra sebanyak 200 sampel dan kemudian disimpan dalam folder "dataset". Dalam tahap pembuatan dataset ini, web camera sudah berada posisi siap untuk merekam wajah *user*. Gambar yang sudah terekam kemudian dimasukkan ke folder dataset [13], seperti ditampilkan pada gambar 17.



Gambar 17. Dataset Sample Citra

2. Training Dataset

Proses training dataset wajah adalah bagian dalam pembuatan *database* wajah yang diambil dari citra wajah *user* pada folder “dataset”. Citra tersebut kemudian diolah menjadi data matriks dalam bentuk byte array dan seluruh matriks wajah yang terkumpul dalam folder dataset disatukan menjadi file .xml. Data training dibutuhkan untuk keakuratan data, semakin banyak data training yang diinputkan, akan semakin akurat hasil dari deteksi wajah [14].

3. Pre-processing

Tahapan *Pre-Processing* terdiri dari pendeteksian wajah, mengubah ukuran wajah yang terdeteksi menjadi resolusi piksel tetap dan mengekstraksi dan pengenalan fitur [15].

Pada proses ini terjadi 2 tahap *Pre-Processing*. *Pre-Processing* I ialah inputan citra RGB dari webcam dikonversi ke grayscale dan dilakukan perhitungan nilai histogram dengan metode LBPH. Tahap *Pre-processing* II Selanjutnya dilakukan proses pendeteksian wajah pada *output* citra *pre-processing* I. Proses pendeteksian wajah dilakukan menggunakan *Classifier Cascade* sebagai referensi fitur wajah yang digunakan oleh algoritma LBPH untuk pendeteksian wajah. Jika terdeteksi wajah pada citra, maka dilakukan tahap *Pre-processing* II untuk pengenalan wajah. Tahap ini terdiri dari pemotongan area citra yang terdeteksi wajah, kemudian diubah ukurannya untuk penyesuaian dengan ukuran citra wajah pada dataset.

Hasil implementasi metode LBPH pada dataset yang ada :

1. Sampel Citra Wajah pada Dataset



Gambar 18. Citra Wajah pada Database

Gambar 18 adalah input wajah yang telah diubah kedalam *grayscale* oleh sistem. Setelah itu citra diubah kedalam banyak pixel yang memiliki nilai berbeda pada setiap bloknya.

2. Perhitungan Nilai Keabu-abuan LBP

Melakukan perhitungan nilai LBP dengan menggunakan rumus LBP (1).

$$LBP_{P,R}(xc,xy) = \sum_{p=0}^{P-x-1} s(gp - gc)2^p \quad (1)$$

Di mana, R= Radius, P = Sampling Point, gp = nilai derajat keabuan pada piksel tetangga ke - p dan gc = nilai derajat keabuan pada piksel pusat. Berikut adalah matriks citra berukuran 5x5.

Gambar 19 adalah matriks wajah dari inputan sampel citra wajah.

40	100	250	30	90
210	67	35	56	34
122	0	65	67	45
211	80	33	84	67
143	125	45	32	89

Gambar 19. Matriks Wajah Inputan

Dari matriks diatas kita akan mengambil matriks 3x3 dengan titik tengah atau *threshold* 67 dan dengan perhitungan LBP kita akan mengubahnya ke dalam nilai biner.

40	100	250
210	67	35
122	0	65

Gambar 20. Matriks Wajah Inputan 3x3

Dari matriks pada gambar 20 dapat ditentukan :
gc = 67

gp = {g0 = 40, g1 = 100, g2 = 250, g3 = 35, g4 = 65, g5 = 0, g6 = 122, g7 = 210}, maka perhitungannya adalah :

- $S(g_0 - gc)2^p = S(40-67)2^0$
= $S(-21)$ nilai $-21 < 0$ maka nilai binernya adalah 0
- $S(g_1 - gc)2^p = S(100-67)2^1$
= $S(33)$ nilai $33 > 0$ maka nilai binernya adalah 1
- $S(g_2 - gc)2^p = S(250-67)2^2$
= $S(183)$ nilai $183 > 0$ maka nilai binernya adalah 1
- $S(g_3 - gc)2^p = S(35-67)2^3$
= $S(-32)$ nilai $-32 < 0$ maka nilai binernya adalah 0
- $S(g_4 - gc)2^p = S(65-67)2^4$

= S(-32) nilai $-32 < 0$ maka nilai binernya adalah 0

• $S(g_5 - gc)_{2p} = S(0-67)_{25}$

= S(-2144) nilai $-2144 < 0$ maka nilai binernya adalah 0

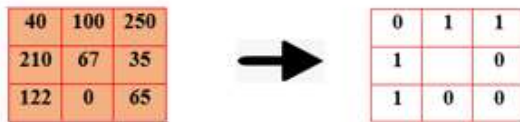
• $S(g_6 - gc)_{2p} = S(122-67)_{26}$

= S(3520) nilai $3520 > 0$ maka nilai binernya adalah 1

• $S(g_7 - gc)_{2p} = S(210-67)_{27}$

= S(18304) nilai $18304 > 0$ maka nilai binernya adalah 1

Dari perhitungan diatas kita mendapatkan pattern pixel value pada gambar 21.



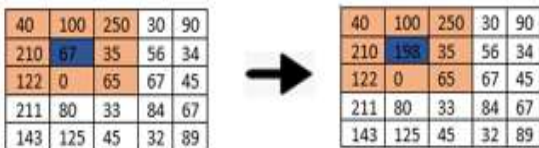
Gambar 21. Matrix Pattern Pixel

Setelah mendapatkan *pattern pixel value* pada tiap blok yang berupa bilangan biner, maka akan dikonversi ke bilangan desimal dengan dikali 2^p seperti pada rumus LBP.



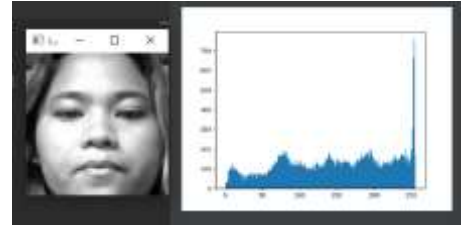
Gambar 22. Matrix Hasil Konversi

$$LBP = 0x1 + 1x2 + 1x4 + 0x8 + 0x16 + 0x32 + 1x64 + 1x128 = 198$$



Gambar 23. Matrix Hasil Perkalian

Setelah dikonversikan ke bilangan desimal dan menghasilkan nilai 198. Angka 198 akan membentuk histogram baru yang menjadi ciri dari setiap citra. Proses yang sama dilakukan untuk mengisi semua matriks dan menghasilkan diagram histogram seperti contoh gambar 24.



Gambar 24. Diagram Histogram dari Sample Citra pada Dataset

A. Pengujian Tingkat Kemiripan Histogram



Gambar 25. Perbandingan Kemiripan Histogram

Dari gambar 25 didapat nilai *Confidence* 28. Nilai *Confidence* adalah suatu pengukuran dari hasil pencocokan nilai histogram pada dataset dengan nilai histogram input citra yang baru. Semakin rendah nilai *confidence* maka tingkat kemiripan suatu citra histogram yang sudah ada terhadap citra yang baru bernilai tinggi. Ini disebabkan karena nilai *confidence* merupakan nilai *distance* atau jarak dari histogram satu ke histogram lainnya.

IV. HASIL DAN PEMBAHASAN

Hasil perancangan meliputi beberapa parameter yang memungkinkan dapat mempengaruhi hasil pendeteksian dan pengenalan wajah, seperti jarak wajah dengan webcam, intensitas cahaya, posisi wajah dengan webcam, dan banyaknya wajah yang terdeteksi dan dikenali.

1. Pendeteksian dan Pengenalan Wajah Berdasarkan Jarak Wajah dengan Webcam

TABEL I. HASIL PENDETEKSIAN DAN PENGENALAN WAJAH DENGAN JARAK BERBEDA

No.	Jarak (cm)	Wajah Terdeteksi	Wajah Dikenali
1	10	Tidak	Tidak
2	30	Ya	Ya
3	100	Ya	Ya

No.	Jarak (cm)	Wajah Terdeteksi	Wajah Dikenali
4	150	Ya	Tidak
5	300	Ya	Tidak

Berdasarkan Tabel 1 maka dapat dihitung tingkat akurasi dari pengujian pendeteksian dan pengenalan wajah berdasarkan kalkulasi berikut:
 Pendeteksian :

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Terdeteksi}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{4}{5} \times 100\%$$

$$\text{Hasil} = 80\%$$

Pengenalan :

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Dikenali}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{2}{5} \times 100\%$$

$$\text{Hasil} = 40\%$$

2. Pendeteksian dan Pengenalan Wajah Berdasarkan Intensitas Cahaya

TABEL II. HASIL PENDETEKSIAN DAN PENGENALAN WAJAH DENGAN INTENSITAS CAHAYA BERBEDA

No.	Intensitas Cahaya	Wajah Terdeteksi	Wajah Dikenali
1	Cahaya Terang	Ya	Ya
2	Cahaya Redup	Ya	Ya
3	Cahaya Terang Salah Satu Sisi	Ya	Tidak
4	Minim Cahaya	Ya	Ya
5	Gelap	Ya	Tidak

Berdasarkan Tabel 2 maka dapat dihitung tingkat akurasi dari pengujian pendeteksian dan pengenalan wajah berdasarkan kalkulasi berikut :
 Pendeteksian :

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Terdeteksi}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{5}{5} \times 100\%$$

$$\text{Hasil} = 100\%$$

Pengenalan :

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Dikenali}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{3}{5} \times 100\%$$

$$\text{Hasil} = 60\%$$

3. Pendeteksian dan Pengenalan Wajah Berdasarkan Kemiringan Wajah

TABEL III. HASIL PENGUJIAN PENDETEKSIAN DAN PENGENALAN WAJAH DENGAN TINGKAT KEMIRINGAN

No.	Kemiringan	Wajah Terdeteksi	Wajah Dikenali
1	10°	Ya	Ya
2	20°	Tidak	Tidak

Berdasarkan TABEL 3 maka dapat dihitung akurasi dari pengujian pendeteksian dan pengenalan wajah berdasarkan kalkulasi berikut :

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Terdeteksi dan Dikenali}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{1}{2} \times 100\%$$

$$\text{Hasil} = 50\%$$

4. Pendeteksian dan Pengenalan Wajah yang Terhalang Objek

TABEL IV. HASIL PENDETEKSIAN DAN PENGENALAN WAJAH YANG TERHALANG OBJEK

No.	Objek	Wajah Terdeteksi	Wajah Dikenali
1	Masker	Tidak	Tidak
2	Topi	Ya	Tidak
3	Kacamata	Tidak	Tidak

Berdasarkan Tabel 4 maka dapat dihitung tingkat akurasi dari pengujian pendeteksian dan pengenalan wajah berdasarkan kalkulasi berikut:

$$\text{Pendeteksian :}$$

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Terdeteksi}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{1}{3} \times 100\%$$

$$\text{Hasil} = 33,3\%$$

Pengenalan :

$$\text{Hasil} = \frac{\text{Jumlah Wajah yang Dikenali}}{\text{Total Pengujian}} \times 100\%$$

$$\text{Hasil} = \frac{0}{3} \times 100\%$$

$$\text{Hasil} = 0\%$$

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Setelah melakukan perancangan dan pengujian serta analisisnya, maka dapat diambil beberapa kesimpulan dari kinerja secara keseluruhan dari sistem absensi wajah. Sistem absensi wajah mampu mendeteksi adanya wajah dengan baik sampai akurasi 100%. Sistem absensi wajah dapat mengenali wajah yang telah terdeteksi sampai akurasi 60%. Pada jarak 20 cm hingga 100 cm dari web cam sistem dapat

mendeteksi dan mengenali wajah dengan baik, ketika jarak ≤ 10 cm dan ≥ 100 cm maka sistem tidak mampu mengenali wajah tetapi masih dapat mendeteksi wajah dan pada jarak yang sudah sangat jauh di mana wajah sudah berukuran sangat kecil maka wajah tidak akan terdeteksi. Sistem absensi wajah mampu mendeteksi dan mengenali wajah dengan baik sampai akurasi 100% pada kondisi ruangan dengan intensitas cahaya terang, redup, dan minim cahaya. Akurasi akan berkurang hingga 60% untuk kondisi ruangan tidak ada cahaya. Sistem juga tidak dapat mengenali wajah bila salah satu sisi wajah disinari sinar terang dan sisi lainnya tidak. Sistem absensi wajah mampu meminimalisir tingkat kecurangan karena wajah dalam database hanya dapat diisi oleh 1 wajah tiap ID. Pendeteksian dan pengenalan wajah sangat bergantung pada intensitas cahaya ruangan saat melakukan absensi dan juga pemakaian aksesoris pada wajah seperti masker, kacamata, topi, dll.

B. Saran

Adapun saran yang dapat diberikan untuk pengembangan selanjutnya antara lain, melakukan penelitian lebih lanjut untuk mendeteksi dan mengenali wajah dalam kondisi menggunakan aksesoris wajah seperti masker, topi, dan kacamata. Melakukan penelitian lebih lanjut untuk mendeteksi dan mengenali wajah dengan jumlah wajah lebih dari 1 wajah. Memperhatikan pencahayaan yang dibutuhkan untuk meningkatkan akurasi pengenalan wajah, karena sangat mempengaruhi gambar.

REFERENSI

- [1] P. D. O. Davies, "Multi-drug resistant tuberculosis," *CPD Infect.*, vol. 3, no. 1, pp. 9–12, 2002.
- [2] H. Habiburrahman, "Sistem Presensi Berbasis Face Recognition," *Electrices*, pp. 15–21, 2019.
- [3] W. Nugraha and M. Syarif, "Penerapan Metode Prototipe Dalam Perancangan Sistem Informasi Penghitungan Volume Dan Cost Penjualan Minuman Berbasis Website," *JUSIM (Jurnal Sist. Inf. Musirawas)*, vol. 3, no. 2, pp. 94–101, 2018, doi: 10.32767/jusim.v3i2.331.
- [4] S. P. Python *et al.*, "Konsep dasar python," pp. 1–6, 2000.
- [5] L. W. Alexander, S. R. Sentinuwo, A. M. Sambul, T. Informatika, U. Sam, and R. Manado, "Implementasi Algoritma Pengenalan Wajah Untuk Mendeteksi Visual Hacking," *J. Tek. Inform.*, vol. 11, no. 1, 2017, doi: 10.35793/jti.11.1.2017.16969.
- [6] F. Setiawan and D. A. R., "Sistem Pengenalan Wajah Dengan Metode Local Binary Pattern Histogram Pada Firebase," *Semin. Nas. Teknol. Inf. dan Komun. STI&K*, vol. 4, no. September, pp. 1–7, 2020.
- [7] S. Al-Aidid and D. Pamungkas, "Sistem Pengenalan Wajah dengan Algoritma Haar Cascade dan Local Binary Pattern Histogram," *J. Rekayasa Elektr.*, vol. 14, no. 1, pp. 62–67, 2018, doi: 10.17529/jre.v14i1.9799.
- [8] Q. Mutiara and E. Prasetyo, "LBPH pada Sistem Pengenalan Wajah," *J. Ilm. Komputasi*, vol. 18, 2019.
- [9] Haviluddin, "Memahami Penggunaan UML (Unified Modelling Language)," *Memahami Pengguna. UML (Unified Model. Lang.)*, vol. 6, no. 1, pp. 1–15, 2011, [Online]. Available: <https://informatikamulawarman.files.wordpress.com/2011/10/01-jurnal-informatika-mulawarman-feb-2011.pdf>.
- [10] Suendri, "Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)," *J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 1–9, 2018, [Online]. Available: <http://jurnal.uinsu.ac.id/index.php/algoritma/article/download/3148/1871>.
- [11] M. Haidar, "Rancang Bangun Sistem Informasi Manajemen Kelompok Keahlian Rekayasa Komputer Universitas Telkom Modul : Pengelolaan Tugas Akhir," *e-Proceeding Eng.*, vol. 6, no. 3, pp. 10359–10363, 2019.
- [12] D. Lahay and S. Sulaeman, "Perancangan Sistem Informasi Pengelolaan Tagihan Iuran Pemasangan Jaringan Pada PT. Molindo Media Persada Berbasis Android," *Semin. Nas. Teknol. Sains dan Hum.*, vol. 2019, no. November, pp. 44–49, 2019.
- [13] A. W. Wibowo, A. Karima, Wiktasari, A. Yobioktabera, and S. Fahriah, "Pendeteksian dan Pengenalan Wajah Pada Foto Secara Real Time Dengan Haar Cascade dan Local Binary Pattern Histogram," *JTET (Jurnal Tek. Elektro Ter.)*, vol. Vol. 9 No., pp. 6 – 11, 2020.
- [14] L. R. Yesy, D. Rosita, and D. Hanum, "Sistem Absensi Mahasiswa Berdasarkan Citra Wajah Menggunakan Metode Principal Component Analysis (Pca)," pp. 1–7, 2019.
- [15] K. Jain and A. Assistant Professor, "Comparison of Face Recognition Algorithms Using Opencv for Attendance System," *Int. J. Sci. Res. Publ.*, vol. 8, no. 2, p. 268, 2018, [Online]. Available: www.ijsrp.org.