

SIMULASI PENGENALAN WAJAH DENGAN METODE LOCAL BINARY PATTERN HISTOGRAM (LBPH)

Romanus Pradista¹, Tony Darmanto², Alfred Yulius A. P.³

^{1,2,3} Informatika Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak

e-mail: ¹18421239_romanus_p@widyadharm.ac.id, ²tony.darmanto@yahoo.com, ³alfredyulius703@gmail.com

Abstract

The principle of facial recognition is that the facial object captured by the camera will be processed and compared with all the images of the face in the existing data set, so that the identity of the face is known. One of the applications of facial recognition is to conduct attendance with individual faces. In this study, a system was created that can detect and recognize a person's face which is using the Local Binary Pattern Histogram (LBPH) method. The programming languages used are Python, openCV and Numpy modules. The Javascript programming language is used for the user interface, so the user scans the face through a browser with a MySQL database to store the identity data and name of the owner of the face. The results of the study using the LBPH face method were successfully identified and the data was saved to the database used for attendance data. The amount of training imagery data and the distance of the object to the camera as well as the quality of the camera resolution affect the results of facial recognition so several tests were carried out. Too far away about 120 cm and above the camera, the face cannot be recognized well because the system finds it difficult to capture the pixel area of facial features. The large number of training images for each test image is also quite influential, the more training images for each test image, the better the percentage of recognition success and vice versa. Furthermore, testing with different camera resolution qualities, the higher the resolution of the camera used, the accuracy of detection and recognition becomes accurate and faster.

Keywords – OpenCV, Numpy, Facial Recognition, LBPH Method

Abstrak

Prinsip dari pengenalan wajah adalah objek wajah yang tertangkap kamera akan diolah dan dibandingkan dengan semua gambar wajah dalam kumpulan data yang sudah ada, sehingga diketahui identitas dari wajah tersebut. Salah satu penerapan pengenalan wajah adalah melakukan presensi dengan wajah individu. Pada penelitian ini dibuat suatu sistem yang bisa mendeteksi serta mengenali wajah seseorang menggunakan metode *Local Binary Pattern Histogram (LBPH)*. Bahasa pemrograman yang digunakan adalah *Python*, modul *openCV* dan *Numpy*. Bahasa pemrograman *Javascript* digunakan untuk antarmuka pengguna, sehingga user melakukan *scanning* wajah melalui *browser* dengan *database MySQL* untuk tempat menyimpan data identitas dan nama pemilik wajah. Hasil penelitian dengan menggunakan metode LBPH wajah berhasil diidentifikasi dan data tersimpan ke database. Jumlah data citra latih dan jarak objek ke kamera serta kualitas resolusi kamera berpengaruh pada hasil pengenalan wajah sehingga dilakukan beberapa pengujian. Jarak terlalu jauh sekitar 120 cm keatas dari kamera, wajah tidak dapat dikenali dengan baik karena sistem sulit menangkap *pixel* area fitur wajah. Banyaknya jumlah citra latih untuk tiap citra uji juga cukup berpengaruh, semakin banyak citra latih untuk tiap citra uji maka persentase keberhasilan pengenalan akan semakin baik begitu juga sebaliknya. Selanjutnya pengujian dengan kualitas resolusi kamera yang berbeda, semakin tinggi resolusi kamera yang digunakan maka akurasi pendeteksian dan pengenalan menjadi akurat dan semakin cepat.

Kata kunci – OpenCV, Numpy, Pengenalan Wajah, Metode LBPH

1. PENDAHULUAN

Saat ini kemajuan ilmu pengetahuan teknologi yang semakin pesat, secara langsung akan memberikan pengaruh besar pada berbagai bidang. Salah satu contohnya pada bidang pendataan dan administrasi. Sistem pendataan merupakan sistem yang sangat penting dan wajib dilakukan sebagai pendukung keputusan. Suatu keputusan tentu harus didukung dengan data yang akurat. Salah satu sistem pendataan yang biasa dilakukan adalah proses presensi. Saat melakukan presensi, contohnya seperti dalam sistem presensi perkuliahan masih ada yang menggunakan metode manual yaitu setiap mahasiswa melakukan presensi secara langsung dengan menandatangani daftar kehadiran. Metode manual ini kurang efisien karena membutuhkan waktu yang lama dalam

mengisi daftar kehadiran dan memungkinkan adanya pelanggaran seperti melakukan manipulasi tanda tangan^[1]. Berikutnya yaitu menggunakan *fingerprint* masih memiliki beberapa kelemahan seperti sulitnya pengidentifikasian jari jika kondisi jari terkelupas, dan berminyak^[2].

Dengan demikian dibutuhkan suatu solusi untuk mengatasi hal tersebut sehingga proses presensi dapat dilakukan secara otomatis dan tidak mudah untuk dilakukan manipulasi. Untuk melakukan presensi secara otomatis tersebut dapat dilakukan dengan menggunakan ciri khusus yang ada pada manusia yaitu wajah. Pada penelitian ini melakukan pengenalan seseorang berdasarkan fitur biometrik yaitu wajah. Metode pengenalan dengan biometrik menggunakan kamera memiliki berbagai keunggulan dibandingkan teknik pengenalan yang konvensional seperti menggunakan kartu atau password. Pengenalan dengan menggunakan wajah tidak dapat digandakan, dicuri, atau terlupa^[3]. Wajah manusia dapat digunakan untuk melakukan identifikasi karena keunikan yang dimiliki berdasarkan parameter-parameter tertentu. Identitas seseorang digunakan untuk banyak keperluan seperti presensi, keamanan, analisis dan lain sebagainya. Keunikan dan pengukuran parameter yang berbeda membantu mengenali seseorang dan dapat memisahkan antara manusia dengan manusia lainnya^[4].

Berdasarkan pemaparan di atas, maka penelitian berfokus pada sistem pengenalan wajah dengan menerapkan metode Local Binary Pattern Histogram (LBPH) diakses melalui *web browser*. Metode LBPH merupakan kombinasi dari metode *Local Binary Pattern* (LBP) dengan *Histograms of Oriented Gradients* (HOG). LBP adalah operator tekstur yang sederhana akan tetapi sangat efisien yang memberi label *pixel* dan menganggap hasilnya sebagai angka biner, sedangkan *Histograms of Oriented Gradients* (HOG) akan meningkatkan kinerja pendeteksian berdasarkan *histogram*.

2. METODE PENELITIAN

Pada bagian ini, dibahas tahapan-tahapan penelitian yang digunakan untuk melakukan pengenalan wajah dengan metode *Local Binary Patterns Histograms* (LBPH) seperti pengumpulan data citra wajah, ekstraksi fitur dengan menggunakan LBPH dan melakukan pengujian. Tahapan pertama yang dilakukan adalah mengumpulkan data citra wajah. Pada penelitian ini banyaknya citra wajah yang dikumpulkan adalah 164 citra wajah. Data tersebut diperoleh dari empat (4) orang dengan tiap orang memiliki 40 citra latih. Data citra wajah tersebut selanjutnya dibagi menjadi dua bagian yaitu bagian pertama sebagai citra data latih dan bagian kedua sebagai citra data uji dengan komposisi citra data latih sebanyak 160 citra dan citra data uji sebanyak empat (4) citra. Untuk citra data uji tersebut diperoleh dengan cara mengekstrak video uji yang berasal dari empat (4) orang menjadi kumpulan citra wajah sebanyak empat (4) citra uji. Tahapan selanjutnya adalah melakukan ekstraksi fitur pada citra data latih dan citra data uji dengan menggunakan metode *Local Binary Patterns Histograms* (LBPH). Untuk melakukan ekstraksi fitur citra pada data latih dan data uji, maka citra data latih dan data uji diubah terlebih dahulu menjadi citra *grayscale*. Metode LBPH dilakukan untuk mendapatkan fitur histogram dari tiap citra wajah berdasarkan citra hasil *Local Binary Patterns* (LBP).

2.1. Rancangan Penelitian, Pengumpulan Data, Teknik Analisis Sistem, Teknik Perancangan Sistem.

2.1.1. Rancangan Penelitian

Dalam penelitian ini, digunakan desain penelitian deskriptif dengan mempelajari literatur-literatur untuk mengetahui cara merancang suatu aplikasi.

2.1.2. Pengumpulan Data

Teknik pengumpulan data yang digunakan adalah melakukan pengambilan data citra wajah dengan kamera. Kamera yang digunakan pada penelitian ini menggunakan kamera *smartphone*

2.1.3. Teknik Analisis Sistem

Teknik Analisis sistem yang digunakan untuk analisis sistem yaitu melakukan proses deteksi objek, penyimpanan ke dataset citra wajah, *training* wajah, identifikasi data pengenalan wajah dan penyimpanan data untuk keperluan presensi secara *real-time*. Deteksi objek dan penyimpanan dataset wajah menggunakan algoritma *haar cascade classifier*, *training* wajah dan disimpan di dalam file *.xml*, kemudian mengidentifikasi data berupa pengenalan wajah dengan metode LBPH dan menyimpan hasil data ke *database* MySQL.

2.1.4. Teknik Perancangan Sistem.

Teknik perancangan sistem yang akan digunakan dalam merancang website yaitu bahasa pemrograman *Python*, *Hypertext Markup Language* (HTML), *Javascript*, *Cascading Style Sheets* (CSS), serta XAMPP sebagai *database*. Perancangan *user interface* menggunakan *Pycharm*. Perancangan *database* menggunakan MySQL.

2.2. Landasan Teori

2.2.1. Perancangan Sistem

Perancangan sistem adalah merupakan perancangan yang melibatkan sistem tertentu, dalam arti sistem yang dirancang adalah sistem yang memang ingin dibuat dan dikendalikan^[6].

2.2.2. Pengenalan Wajah

Face Recognition (Pengenalan Wajah) adalah salah satu teknologi biometrik yang telah banyak diaplikasikan dalam sistem security selain pengenalan retina mata, pengenalan sidik jari dan iris mata. Pada aplikasinya sendiri pengenalan wajah menggunakan sebuah kamera untuk menangkap wajah seseorang kemudian dibandingkan dengan wajah yang sebelumnya yang telah disimpan di dalam database tertentu^[7].

2.2.3. Computer Vision

Computer Vision adalah ilmu pengetahuan dan teknologi dari suatu sistem komputer yang mampu memiliki kemampuan akuisisi, memproses, dan memahami (*understanding*) dari suatu image/streaming video^[10]. *Computer Vision* merupakan bidang pengetahuan yang berfokus pada bidang sistem kecerdasan buatan dan berhubungan dengan akuisisi dan pemrosesan image^[11].

2.2.4. Sampling

Sampling adalah proses mapping fungsi kontinu ke diskrit. Citra harus mengalami sampling dan kuantisasi agar dapat diproses dengan komputer yang bersifat diskrit (proses digitalisasi). Sampling menyatakan besarnya kotak-kotak yang disusun pada citra menyatakan besar kecilnya ukuran *pixel* (titik) pada citra, dan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam nilai tingkat keabuan (*grayscale*) sesuai dengan jumlah *bit biner* yang digunakan oleh mesin dengan kata lain kuantisasi pada citra menyatakan jumlah warna yang ada pada citra^[13].

2.2.5. Thresholding

Thresholding merupakan metode segmentasi yang paling mudah, sebagai contoh memisahkan area *image* suatu objek dengan area lainnya dengan cara variasi intensitas (dengan membuat image orisinal menjadi gray) di antara piksel objek tersebut dengan piksel background dengan nilai ambang (*threshold*) tertentu^[10]. *Thresholding* merupakan salah satu jenis metode segmentasi citra yang prosesnya berdasarkan perbedaan derajat keabuan citra. Pada proses ini dibutuhkan suatu nilai batas yang disebut nilai *threshold*. Nilai intensitas lebih atau sama dengan nilai *threshold* akan diubah menjadi putih (1) sedangkan nilai intensitas citra yang kurang dari nilai *threshold* akan diubah menjadi hitam (0). Sehingga hasil dari *output* dari *threshold* ini adalah berupa citra biner^[12].

2.2.6. Histogram Equalization

Histogram equalization merupakan representasi grafis dari distribusi intensitas dari image yang menguantifikasi jumlah piksel untuk tiap intensitas^[10]. *Histogram equalization* adalah salah satu teknik dalam meningkatkan kualitas citra dengan cara mendistribusikan tingkat piksel keabuan dalam sebuah histogram^[14].

2.2.7. Metode Local Binary Pattern Histogram (LBPH)

Local Binary Pattern Histogram (LBPH) merupakan sebuah fitur yang berfungsi untuk mengklasifikasi citra yang dikombinasikan dengan histogram dan LBPH salah satu teknik terbaru dari metode LBP yang digunakan untuk mengubah performa hasil pengenalan wajah. LBP pada umumnya didesain untuk pengenalan tekstur. Ide dasar *Local Binary Pattern* adalah untuk merangkum struktur lokal dalam gambar dengan membandingkan setiap piksel dengan lingkungannya. Kemudian menganggap hasilnya sebagai angka biner. Lalu mengubah angka biner menjadi angka desimal, dan angka desimal itu adalah nilai baru dari piksel tengah. Angka desimal ini disebut nilai *pixel* LBPH dan kisarannya 0-255^[5]. Secara umum kerangka kerja dari metode LBPH merupakan proses ekstraksi ciri fitur berdasarkan tekstur dengan membagi citra ke dalam beberapa area lokal dan mengekstraksi seluruh area lokal untuk mendapatkan pola biner lokal. Teknik dasar pada LBPH adalah melakukan analisis ketetanggaan terhadap setiap *pixel* pada sebuah citra, sebagai contoh sebuah gambar direpresentasikan oleh nilai matrik. LBPH mendeteksi struktur mikro seperti tepi, garis, bintik dan area datar yang dapat diperkirakan dengan histogram.

Koordinat dari titik pusat *pixel* adalah (x_c, y_c) , koordinat dari tetangga P (x_p, y_p) pada tepi lingkaran dengan jari-jari R ditentukan menggunakan persamaan *neighborhood* di bawah ini:

$$x_p = x_c + R \cos\left(\frac{2\pi p}{P}\right), y_p = y_c - R \sin\left(\frac{2\pi p}{P}\right)$$

Keterangan:

x_p, y_p : titik koordinat ketetanggaan x dan y

x_c, y_c : titik pusat dari x dan titik pusat dari y

R : *Radius*

P : *Sampling point*

Berikutnya tahapan *thresholding* merupakan proses semua nilai piksel tetangga pada setiap pola dikomparasi dengan nilai di titik pusat atau nilai *threshold*. Kemudian diubah menjadi nilai biner (0 dan 1). Jika nilai ketetanggaan lebih kecil dari nilai titik pusat maka bernilai 0, sedangkan bernilai 1 jika nilai lebih besar atau sama dengan dari nilai titik pusat. Langkah ini untuk membantu dalam mendapatkan informasi tentang perbedaan dari biner lokal pada setiap bagian *pixel*. Kemudian langkah selanjutnya sejumlah nilai biner yang diperoleh dari langkah *thresholding* akan dikonversi ke nilai desimal untuk karakteristik dari struktur pola. Tahapan mendapatkan nilai LBP, nilai biner yang telah didapatkan, dikalikan dengan nilai bobot. Kemudian hasil dari perkalian nilai bobot dan nilai biner tersebut dijumlahkan untuk mendapatkan nilai *pixel* LBP.

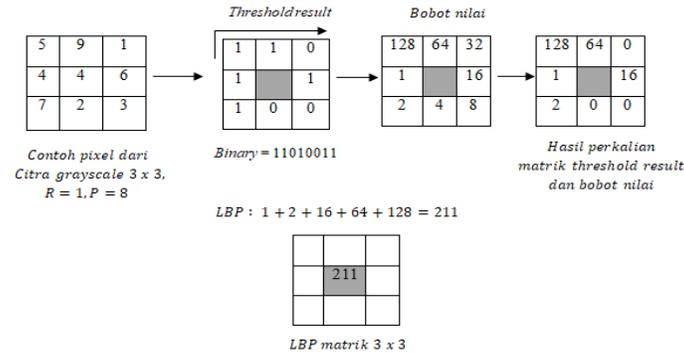
Nilai intensitas dari titik pusat *pixel* adalah g_c dan g_p nilai intensitas dari *pixel* tetangga dengan indeks P adalah dengan $p = 0, 1, 2, \dots, P - 1$, nilai LBP untuk *pixel* (x_c, y_c) dapat ditentukan menggunakan persamaan berikut.

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(G_i - G_c) 2^p$$

dengan fungsi *threshold* $s(x)$:

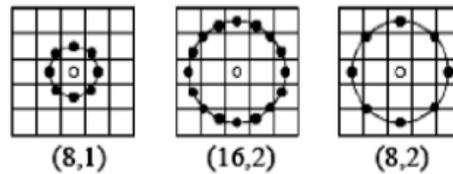
$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

$LBP_{p,R}(x_c, y_c)$ adalah hasil dari operator LBP, x_c, y_c adalah koordinat piksel tengah pada sel yang berukuran 3×3 , G_c adalah nilai *pixel* tengah pada citra *grayscale* dan G_i adalah nilai dari *pixel* tetangga pada citra *grayscale*. Gambar 1 merupakan contoh ilustrasi proses operator LBP.



Gambar 1. Ilustrasi Proses Operator LBP

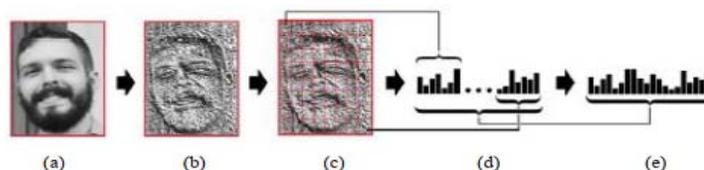
Pada aplikasi analisis tekstur diharapkan untuk memiliki fitur yang invarian agar kuat untuk rotasi gambar input untuk mengetahui tingkat akurasi terbaik dari ekstraksi fitur menggunakan LBP yaitu dapat memperluas jumlah radius dan tetangga. Seperti pada gambar 2 contoh varian rotasi seperti nilai radius = 2 dan nilai *Sampling Point* P = 16. Prosedur matrik LBP dapat diperluas dengan menggunakan jumlah radius dan tetangga yang berbeda, atau biasanya disebut dengan *LBP circular*.



Gambar 2. Varian Pola Ketetangaan LBP

Setelah mendapatkan hasil citra biner lokal, selanjutnya dilakukan ekstraksi histogram dari nilai LBP dengan menggunakan metode *Histograms of Oriented Gradients (HOG)*. Teknik mengekstrak fitur untuk metode ini yaitu dari distribusi lokal dari intensitas *gradient* tiap *pixel* yang terdapat pada sebuah objek citra. Dalam metode *Histogram of Oriented Gradients* pada pengenalan wajah manusia, ukuran sel berupa kumpulan atau gabungan *pixel* dan blok berupa kumpulan atau gabungan sel beserta jumlah *orientation bin* yang merupakan tempat untuk menampung hasil arah dan besar *gradient* akan mempengaruhi hasil keluaran fitur vektor yang dihasilkan dan juga akurasi yang didapat. Pada metode HOG terdapat empat (4) langkah yaitu:

- Citra hasil LBP dibagi menjadi daerah-daerah kecil yang disebut dengan grid.
- Membuat histogram dari tiap daerah tersebut.
- Oleh karena citra hasil LBP merupakan citra *grayscale* maka setiap histogram dari setiap grid hanya akan berisi 256 posisi (0-255) yang mewakili setiap intensitas *pixel*.
- Histogram dari tiap daerah (grid) digabung menjadi satu histogram besar. Pada penelitian ini menggunakan grid berukuran 8×8 , maka total memiliki $8 \times 8 \times 256 = 16,384$ posisi dalam histogram akhir. Histogram terakhir merupakan fitur yang mewakili citra asli dan diasumsikan sebagai sebuah vektor. Pada gambar di bawah merupakan contoh ilustrasi LBPH, (a) Citra asli, (b) Citra hasil operator LBP, (c) Pembagian daerah grid pada citra LBP, (d) Histogram tiap daerah grid, (e) Gabungan dari histogram tiap daerah^{[8][9]}.

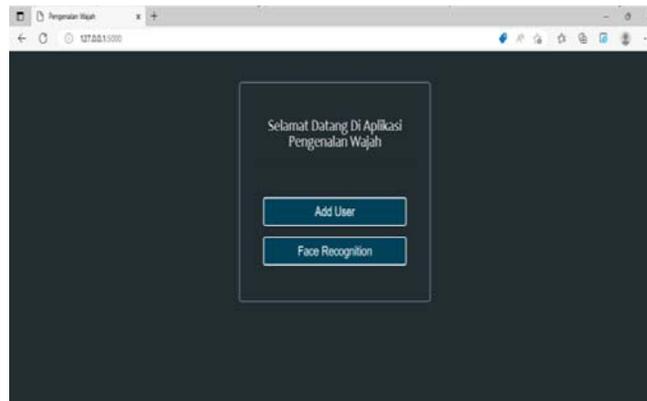


Gambar 3. Ilustrasi proses LBPH

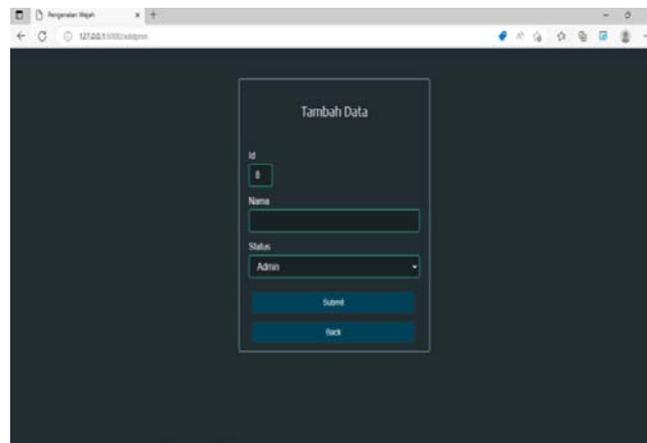
3. HASIL DAN PEMBAHASAN

3.1. Pengambilan citra data latih

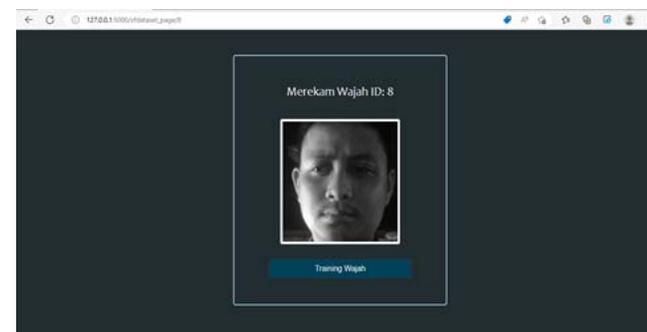
Deteksi objek dalam penelitian ini menggunakan algoritma *Haar Cascade Classifier* untuk melakukan deteksi wajah dan menyimpan sampel wajah ke dataset. Algoritma *Haar Cascade Classifier* atau yang dikenal dengan nama lain *Haar-Like Features* merupakan *rectangular features* (fungsi persegi), yang memberikan indikasi secara spesifik pada sebuah citra wajah. Prinsip *Haar-like features* adalah mengenali obyek berdasarkan nilai sederhana dari fitur tetapi bukan merupakan nilai piksel dari image obyek tersebut. Algoritma ini memiliki kelebihan yaitu komputasinya sangat cepat, karena hanya bergantung pada jumlah piksel dalam persegi bukan setiap nilai piksel dari sebuah *image*. Algoritma *Haar Cascade Classifier* adalah salah satu metode untuk deteksi wajah dengan berdasarkan kepada klasifikasi gambar berdasarkan nilai fitur dalam sebuah citra. Salah satu manfaat penggunaan fitur secara langsung adalah fitur dapat digunakan untuk mengkodekan pengetahuan dengan data latih. Selain itu penggunaan fitur beralasan bahwa operasi fitur dianggap lebih cepat daripada klasifikasi berbasis *piksel*.



Gambar 4. Tampilan Halaman Menu



Gambar 5. Tampilan Halaman Add User



Gambar 6. Pengambilan Dataset Citra Wajah

Ketika wajah sudah terdeteksi objek wajah di-*capture* selama beberapa detik atau sesuai dengan banyak citra latih, setelah itu data wajah akan tersimpan di folder dataset. Ketika citra wajah sudah tersimpan maka, program akan di-*pause* sampai kemudian mengeklik tombol "Training Wajah" untuk melakukan pelatihan terhadap citra di folder dataset. Data *training* menggunakan metode LBPH. Metode LBPH dapat mencari tekstur dari sebuah gambar kemudian mengambil setiap *pixel* dari gambar tersebut, sehingga kerapatan antar *pixel* dapat mengoptimalkan prediksi gambar saat

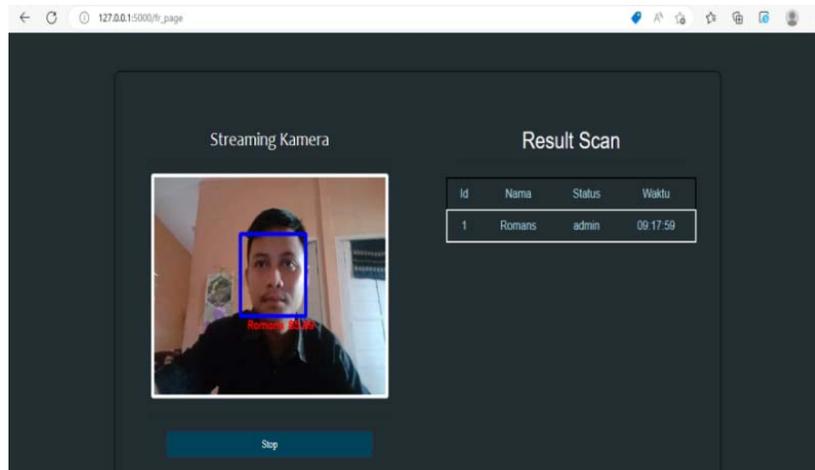
dilakukan identifikasi wajah. Pada metode LBPH, citra yang terdapat di folder dataset akan dibagi dibagi menjadi beberapa bagian, kemudian dilakukan proses ekstraksi fitur untuk mendapatkan nilai histogram dengan mengubah setiap *pixel* dari citra menjadi *thresholding* dengan ukuran 3x3 dengan diambil satu *pixel* tengah untuk menjadi pembanding. Jika nilai tetangga lebih besar atau sama besar dari *pixel* yang menjadi pembanding dalam nilai *grayscale* maka akan diubah nilainya menjadi 1 dan sebaliknya jika nilai tetangga lebih kecil dari pembanding dalam nilai *grayscale* maka akan diubah nilai *pixel*-nya menjadi 0. Setelah diubah menjadi nilai biner maka akan menghasilkan deret biner yang kemudian jika diubah menjadi nilai desimal akan menghasilkan nilai baru untuk nilai tengah *pixel* selanjutnya, deret biner tersebut yang disebut dengan kode LBP. Hasil dari deret biner ini disimpan didalam file *.xml* yang akan digunakan untuk proses pengenalan wajah.



Gambar 7. Dataset Citra Wajah

3.2. Hasil Pengenalan Wajah dan Penyimpanan Data Presensi

Selanjutnya akan dilakukan pengenalan wajah. Seperti pada data *training*, pengenalan wajah menggunakan metode LBPH untuk mendapatkan nilai histogram citra uji, Informasi di data *training* yang sudah tersimpan di file *.xml* akan dibandingkan dengan data *real-time* citra uji. Ketika data cocok atau memiliki dengan data yang tersimpan di file *.xml*, maka identitas dari data citra uji tersebut akan ditampilkan di layar berupa nama pemilik wajah. Data identitas yang sudah teridentifikasi akan disimpan di database MySQL yang digunakan untuk data presensi.



Gambar 8. Hasil Wajah Berhasil Dikenali



Gambar 9. Citra Uji

Hasil dari penelitian menggunakan beberapa parameter pengujian yaitu pengujian berdasarkan jumlah citra

latih, pada pengujian ini bertujuan untuk mengetahui seberapa berhasil sistem dalam mengenali wajah yang telah di-*training* berdasarkan banyak citra wajah tiap orang pada data *training*. Jumlah percobaan sebanyak tujuh (7) kali tiap pengujian. Pengujian dilakukan pada jumlah wajah data *training* tiap citra uji, yaitu 10 citra latih, 20 citra latih, 30 citra latih, dan 40 citra latih dari tiap citra uji. Parameter pengujian berikutnya yaitu pengenalan wajah berdasarkan jarak. Pada pengujian ini bertujuan untuk mengetahui seberapa berhasil sistem dalam mengenali citra wajah berdasarkan jarak antara kamera dengan objek wajah. Pengujian dilakukan dengan cara wajah menghadap ke kamera secara tegak lurus dengan jarak yang berbeda-beda yaitu 20 cm, 50 cm, 100 cm, dan 120 cm. Kemudian program di-*running* dan dilakukan pengenalan objek berdasarkan jarak yang telah ditentukan pada tiap citra uji. Setiap *running* program diberikan waktu jeda pengenalan selama 6 detik untuk sistem mendeteksi dan mengenali citra wajah. Pengujian terakhir yaitu mengenali citra wajah berdasarkan resolusi kamera. Pengujian dilakukan dengan kamera *smartphone* dengan resolusi kamera 8 MP, 16 MP dan 64 MP. Pengujian dilakukan dengan cara wajah menghadap ke kamera secara tegak lurus. Seperti pada pengujian sebelumnya program di-*running* dan dilakukan pengenalan objek berdasarkan jarak yang telah ditentukan, pada tiap citra uji.

Tabel 1. Hasil Pengujian Berdasarkan Banyak Citra Latih

No	Hasil Pengenalan wajah user	Jumlah citra latih	Jumlah percobaan	Kesimpulan
1	 ID : 3	10 citra latih	7 kali	5 dikenali
		20 citra latih	7 kali	5 dikenali
		30 citra latih	7 kali	7 dikenali
		40 citra latih	7 kali	7dikenali
2	 ID : 4	10 citra latih	7 kali	4 dikenali
		20 citra latih	7 kali	5 dikenali
		30 citra latih	7 kali	7 dikenali
		40 citra latih	7 kali	7 dikenali
3	 ID : 5	10 citra latih	7 kali	4 dikenali
		20 citra latih	7 kali	5 dikenali
		30 citra latih	7 kali	6 dikenali
		40 citra latih	7 kali	7 dikenali
4	 ID : 6	10 citra latih	7 kali	5 dikenali
		20 citra latih	7 kali	6 dikenali
		30 citra latih	7 kali	6 dikenali
		40 citra latih	7 kali	7 dikenali

Tabel 2. Hasil Pengujian Berdasarkan Jarak Objek Wajah

No	Hasil Pengenalan	Jarak pengujian	Jumlah percobaan	Kesimpulan
1		20 cm	5 kali	5 dikenali
		50 cm	5 kali	4 dikenali

	 Heti Permasi 59,97 ID : 3	100 cm	5 kali	3 dikenali
		120 cm	5 kali	3dikenali
2	 Romana 78,13 ID : 4	20 cm	5 kali	4 dikenali
		50 cm	5 kali	5 dikenali
		100 cm	5 kali	3 dikenali
		120 cm	5 kali	3 dikenali
3	 Rico Patrio 80,3 ID : 5	20 cm	5 kali	4 dikenali
		50 cm	5 kali	5 dikenali
		100 cm	5 kali	3 dikenali
		120 cm	5 kali	3 dikenali
4	 ID : 6	20 cm	5 kali	5 dikenali
		50 cm	5 kali	4 dikenali
		100 cm	5 kali	3 dikenali
		120 cm	5 kali	3 dikenali

Tabel 3. Hasil Pengujian Dengan Kamera 8 MP

No	Hasil Pengenalan wajah user	Jarak Pengujian	Jumlah Percobaan	Spesifikasi Kamera	Kesimpulan
1	 ID: 3	120 cm	5 kali	8 MP, f/1.8, Resolusi video 1080p@30fps	3 kali berhasil dikenali
2	 ID : 4	120 cm	5 kali	8 MP, f/1.8, Resolusi video 1080p@30fps	3 kali berhasil dikenali
3	 ID: 6	120 cm	5 kali	8 MP, f/1.8, Resolusi video 1080p@30fps	4 kali berhasil dikenali

Tabel 4. Hasil Pengujian dengan kamera 16 MP

No	Hasil Pengenalan wajah user	Jarak Pengujian	Jumlah percobaan	Spesifikasi Kamera	Kesimpulan
1	 ID : 4	120 cm	5 kali	16 MP (f/2.4), Resolusi video 1080p@30fps	4 kali berhasil dikenali
2	 ID : 3	120 cm	5 kali	16 MP (f/2.4), Resolusi video 1080p@30fps	5 kali berhasil dikenali
3	 ID : 5	120 cm	5 kali	16 MP (f/2.4), Resolusi video 1080p@30fps	3 kali berhasil dikenali

Tabel 5. Hasil Pengujian Dengan Kamera 64 MP

No	Hasil Pengenalan wajah user	Jarak Pengujian	Jumlah percobaan	Spesifikasi Kamera	Kesimpulan
1	 ID : 4	120 cm	5 kali	64 MP (f/1.9), 26mm (wide), Resolusi video 4K	4 kali berhasil dikenali
2	 ID : 3	120 cm	5 kali	64 MP (f/1.9), 26mm (wide), Resolusi video 4K	5 kali berhasil dikenali
3	 ID : 5	120 cm	5 kali	64 MP (f/1.9), 26mm (wide), Resolusi video 4K	4 kali berhasil dikenali

4. KESIMPULAN

Berdasarkan uraian di atas, bahwa pengenalan wajah dengan metode LBPH berhasil dilakukan menggunakan bahasa pemrograman *Python*. Tingkat keberhasilan dan keakuratan pengenalan wajah dipengaruhi oleh beberapa parameter yaitu banyak citra latih wajah, jarak objek wajah dan resolusi kamera. Kesimpulan yang penulis ambil dari uraian di atas adalah sebagai berikut:

- a. Proses pengenalan wajah pada sistem pengenalan wajah dengan metode LBPH menghasilkan tingkat akurasi pengenalan yang cukup baik.
- b. Berdasarkan hasil dari pengujian yang telah dilakukan, pada pengujian pertama penulis menyimpulkan bahwa semakin sedikit citra latih kemungkinan objek wajah dikenali akan lebih sulit. Pengujian kedua penulis menyimpulkan bahwa jarak di atas 120 cm akan semakin sulit dikenali, pengujian ketiga, pada pengujian ketiga peneliti menggunakan kamera *smartphone*, disimpulkan bahwa semakin tinggi resolusi suatu kamera yang digunakan maka didapatkan hasil yang lebih baik, karena dengan resolusi kamera yang tinggi pengolahan menjadi lebih stabil sehingga akan mendukung pengenalan objek yang lebih akurat dan cepat.
- c. Sistem pengenalan wajah yang diakses melalui halaman *browser* dengan metode LBPH berhasil mengenali wajah.

5. SARAN

Setelah menerapkan metode *Local Binary Pattern Histogram* (LBPH) untuk melakukan pelatihan dan pengenalan wajah, penulis menyadari bahwa aplikasi pengenalan wajah masih belum sempurna dan masih banyak kekurangan, untuk itu penulis memberikan beberapa saran agar aplikasi pengenalan wajah menggunakan *Local Binary Pattern Histogram* (LBPH) dapat dikembangkan lebih lanjut, beberapa hal yang perlu diperhatikan:

- a. Penggunaan kamera yang lebih baik dapat meningkatkan kemampuan sistem dalam mendeteksi dan mengenali wajah.
- b. Untuk meningkatkan keberhasilan pengenalan wajah, citra latih sebaiknya memiliki tingkat kecerahan dan ekspresi yang berbeda-beda. Kualitas citra latih juga dapat mempengaruhi tingkat keberhasilan pengenalan wajah.
- c. Penelitian ini masih sangat sederhana, sehingga diperlukan penelitian lebih lanjut untuk dapat lebih bermanfaat.

UCAPAN TERIMA KASIH

Dalam penelitian ini, peneliti telah banyak mendapatkan bantuan berupa bimbingan, petunjuk, saran maupun dorongan moril dari berbagai pihak. Maka pada kesempatan ini peneliti mengucapkan terima kasih yang sebesar-besarnya atas bantuan dan dorongan selama penelitian hingga selesainya penelitian ini kepada seluruh civitas akademika Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak, kepada keluarga, beserta teman terkasih.

DAFTAR PUSTAKA

- [1] Kokasih, Rifki dan Christian Daomara. (Oktober 2021). "*Pengenalan Wajah Dengan Menggunakan Metode Local Binary Pattern Histogram (LBPH)*." Jurnal Media Informatika Budidarma. Vol. 5, no. 4: hal. 1258-1264.
- [2] Septyanto, Moh. Wahyu, Herry Sofyan, Herlina Jayadianti, Oliver Samuel S., dan Dessyanto Boedi P. (Oktober 2019). "*Aplikasi Presensi Pengenalan Wajah Dengan Menggunakan Algoritma Haar Cascade Classifier*". Telematika. Vol.16, no. 2: hal. 87-96.
- [3] Al-Aidid, Sayyed dan Daniel S. Pamungkas. (April 2018). "Sistem Pengenalan Wajah dengan Algoritma Haar Cascade dan Local Binary Pattern Histogram." Jurnal Rekayasa Elektrika. Vol. 14, no. 1: hal 62-67.
- [4] Setia Buana, I Komang. (Juli 2021). "Penerapan Pengenalan Wajah Untuk Aplikasi Absensi dengan Metode Viola Jones dan Algoritma LBPH." Jurnal Media Informatika Budidarma. Vol. 5. no. 3: hal. 1008.
- [5] Ramadini, Febrin Ludia dan Emy Haryatmi. (Maret 2022). "*Penggunaan Metode Haar Cascade Classifier dan LBPH Untuk Pengenalan Secara Realtime*." Jurnal Nasional Informatika dan Teknologi Jaringan. Vol. 6. no.2: hal. 1.
- [6] Pane, Syafrial Fachri, Mochamad Zamzam dan Muhamad Diar Fadillah. (2020). *Membangun Aplikasi Peminjaman Jurnal Menggunakan Aplikasi Oracle Apex Online*. Kreatif Industri Nusantara. Bandung.
- [7] Suryansah, Alwan, Roni Habibi, dan Rolly Maulana Awangga. (2020). *Penggunaan Face Recognition Untuk Akses Ruangan*. Kreatif Industri Nusantara. Bandung.

-
- [8] Singh, Abhishek Pratap, SunilKumar S Manvi, Pratik Nimbale, dan Gopal Krishna Shyam. (Mei 2019). “*Face Recognition System Based on LBPH Algorithm*”. International Journal of Engineering and Advanced Technology (IJEAT). Vol. 8. no. 5: hal. 1-5.
- [9] Bhagekar, Shashank, Saroj Jamdhade, Akash Gutti, Renuka Kamble, dan Prof. Amruta Vidwat. (2018). “*Face Recognition Using Local Binary Pattern Histogram (LBPH) Technique.*” International Journal Of Advanced Computing And Electronics Technology (IJAGET). Vol. 5. no.1: hal.3.
- [10] Budiharto, Widodo, Jarot S. Suroso, Alexander Agung Santoso Gunawan, dan Andry Chowanda. (2021). *Desain dan Pemrograman Drone Cerdas*. Penerbit ANDI. Yogyakarta.
- [11] Putro, Hanson Prihantoro, Tri Wahyu Widyarningsih, Inti Englishtina, Eko Nursanty, Robet, Efitra, Sepriano dan Herman Demas. (2023). *Development Of Artificial Intelligence Applications (Studi Kasus dan Implementasi AI Menggunakan Berbagai Bahasa Pemrograman)*. PT. Sonpedia Publishing Indonesia. Kota Jambi.
- [12] Muttaqin, Yuswardi, Elly Warni, Wahyudin S, Radhinka Bagaskara, Agus Ambarwari, Muhammad Rizal H, Imam Ekowicaksono, Defiariany, Zelvi Gustiana, Marzuki Sinambela, dan Ilham Djufri. (2023). *Konsep Dasar Kecerdasan Buatan*. Yayasan Kita Menulis. Indonesia.
- [13] Nafi'iyah, Nur dan Siti Mujilawati. (2018). *Buku Ajar Citra Binarisasi Dan Encancement*. CV Budi Utama. Yogyakarta.
- [14] Leonardy, Joshu. (2022). *Peran Fisikawan Indonesia Dalam Pengembangan Ilmu Pendidikan, Sains, dan Teknologi sebagai Upaya Membangun Sumber Data Manusia yang Berkualitas di Era Society 5.0*. Uwais Inspirasi Indonesia. Ponorogo, Indonesia.