

## PERANCANGAN SISTEM DETEKSI WAJAH UNTUK PRESENSI KEHADIRAN MENGGUNAKAN METODE LBPH ( Local Binary Pattern Histogram) BERBASIS ANDROID

*Implementation Identification of Face Recognition Using LBPH (Local Binary Pattern Histogram) Method For Attendance Presence Based Android*

Ahmad Fauzan<sup>1</sup>, Ledy Novamizanti, S.Si., M.T<sup>2</sup>, Yunendah Nur Fuadah, ST., MT.<sup>3</sup>

<sup>1,2,3</sup>Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

<sup>1</sup>[janpau@gmail.com](mailto:janpau@gmail.com), <sup>2</sup>[ledyaldn@telkomuniversity.ac.id](mailto:ledyaldn@telkomuniversity.ac.id), <sup>3</sup>[yunendah@gmail.com](mailto:yunendah@gmail.com)

### Abstrak

Setiap manusia di dunia ini pasti memiliki bentuk wajah yang tidak sama bahkan dengan kembar identik memiliki perbedaan dalam bentuk wajah. Salah satu cara manusia untuk mengenal setiap individu lain adalah dengan mengingat bentuk wajah individu tersebut dalam memori otak kita. Pada memori otak manusia akan memberikan sebuah gambaran wajah yang pernah kita kenal sebelumnya. Oleh karena itu, dibuatlah aplikasi deteksi wajah yang dapat mengenali wajah dan identitas yang ada dalam database berbasis android.

Dalam tugas akhir ini dibuat sebuah sistem deteksi wajah berbasis android yang dapat menerjemahkan citra latih dalam format jpg dan dapat dikenal oleh system dalam citra uji . Citra yang menjadi masukan berupa citra wajah yang terdiri dari 90 citra dimana terdapat 60 citra latih dan 30 citra uji. Tahapan *pre-processing* terdiri dari *scaling*, *rescale*, konversi RGB ke grayscale, normalisasi wajah, mengurangi efek noise

Setelah dilakukan *preprocessing*, selanjutnya tahap ekstraksi fitur menggunakan *Local Binary Pattern Histogram* (LBPH) yang bertujuan untuk mencari nilai yang dapat dikenal oleh system dalam citra hasil *preprocessing* lalu disimpan kedalam *database*. Lalu dilakukan klasifikasi menggunakan nilai histogram dengan *Neighbor*.

Berdasarkan hasil pengujian aplikasi deteksi wajah dapat dilakukan dengan menggunakan metode *LBPH* (Local Binary Pattern Histogram) pada parameter *Neighbors*=9, *Gridx*. *Gridy* = 8x8, *radius*=1, ukuran tresh hold *BW*=180, dan jarak 20-25cm mendapatkan akurasi sebesar 95.56 dan waktu komputasi 2.35 detik.

**Kata kunci:** LBPH,Citra, android, FAR,FRR.

### Abstract

Every human being in this world must have a face shape that is not the same even with identical twins have differences in the shape of the face. One of the human ways to get to know each other is to remember the shape of the face in our brain memory. On the memory of the human brain will provide a picture of a face we once knew before. Therefore, a face detection application that can recognize faces and identities that exist in the android-based database.

In this final project created an android based face detection system that can translate the image of training in jpg format and can be recognized by the system in the test image. The image into the form of a face image consisting of 90 images where there are 60 images of training and 30 test images. Pre-processing stages consist of registration, RGB, facial normalization, reduced noise effects

After preprocessing, the feature extraction stage uses *Local Binary Pattern Histogram* (LBPH) which aims to find the value that can be recognized by the system in the image of the preprocessing result and then stored into the database. Then do the classification using *K-Nearest Neighbor* method.

Based on the results of application testing can be done by using the *LBPH* (Local Binary Pattern Histogram) method in the *Neighbors* = 9, *Gridx* parameter. *Gridy* = 8x8, *radius* = 1, *BW* resistant tresh size = 180, and an accuracy of 20-25cm has an accuracy of 95.56 and a computing time of 2.35 seconds. Keywords: LBPH,Image, android, FAR,FRR

**Keywords:** LBPH, Image, android, FAR, FRR

## 1. Pendahuluan

Setiap manusia di dunia ini pasti memiliki bentuk wajah yang tidak sama bahkan dengan kembar identik memiliki perbedaan dalam bentuk wajah. Salah satu cara manusia untuk mengenal setiap individu lain adalah dengan mengingat bentuk wajah individu tersebut dalam memori otak kita. Pada memori otak manusia akan memberikan sebuah gambaran wajah yang pernah kita kenal sebelumnya. Pengenalan wajah merupakan suatu kemampuan yang digunakan oleh manusia dalam biometrik untuk membedakan manusia yang satu dengan lainnya[1] Seiring dengan semakin canggihnya teknologi saat ini, pengenalan wajahpun bisa dilakukan oleh sebuah sistem dengan *artificial intelligence* yang terhubung. Kemampuan untuk mengenal wajah tersebut memiliki kecerdasan buatan kemudian yang diimplementasikan dalam sebuah perangkat dengan *platform* android sehingga memiliki kemampuan sama seperti manusia.

Pada penelitian sebelumnya sudah ada sistem pengenalan wajah manusia menggunakan metode Hidden Markov Models (HMM) oleh Sepritahara mencapai tingkat akurasi pengenalan sebesar 84,28%, dengan database 70 gambar yang terdiri dari 10 individu dengan masing-masing individu memiliki 7 variasi ekspresi yang berbeda[4]. Pada penelitian selanjutnya juga sudah dilakukan deteksi wajah dengan penulis Esty dkk [5] membuat aplikasi yang mampu pendeteksian beberapa wajah dengan berbagai posisi pada foto menggunakan metode eigenface. Posisi pengujian yang dilakukan adalah tampak depan, atas, bawah, kanan dan kiri, ukuran, pencahayaan dan latar belakang. akurasi PCA yang diteliti oleh Dian Esti Pratiwi dan Agus Harjoko pada tahun 2013 dengan judul Implementasi Pengenalan Wajah Menggunakan PCA mempunyai akurasi sebesar 82,81%.

Eko dkk [6] melakukan pengujian terhadap metode chi square dan Local binary Pattern (LBP) dengan menggunakan satu wajah secara bergantian. Hasil dari penelitian mengatakan bahwa metode chi square lebih baik jika dibandingkan dengan metode PCA untuk fitur Local binary Pattern (LBP) yaitu tingkat pengenalan mencapai 92,4 %. Penelitian dilakukan oleh Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen dengan *Face Recognition with Local Binary Patterns* University of Oulu, Finland dan penelitian menghasilkan akurasi 95%[7].

Dalam penelitian ini yang berjudul "PERANCANGAN SISTEM DETEKSI WAJAH UNTUK PRESENSI KEHADIRAN MENGGUNAKAN METODE LBPH(local Binary Pattern Histogram) BERBASIS ANDROID" membahas tentang aplikasi android dimana masukkan berupa citra wajah hasil yang di *capture* dari kamera *handphone* dalam bentuk format jpg. Ada beberapa tahapan proses diantaranya yaitu tahap pertama yaitu *pre-processing* tujuannya agar menghasilkan citra yang lebih baik untuk mempermudah ke proses selanjutnya. Pada tahap *pre-processing* dilakukan Tahapan *pre-processing* terdiri dari registrasi, RGB grayscale, normalisasi wajah, mengurangi efek noise. Tahap kedua yaitu ekstraksi ciri menggunakan metode *Local Binary Pattern Histogram (LBPH)* dan Tahap ketiga yaitu klasifikasi menggunakan ekstraksi fitur yang mengelompokkan data baru berdasarkan jarak data baru itu ke beberapa data atau tetangga (*neighbor*) yang paling terdekat. [7]. Hasil yang diharapkan pada perancangan aplikasi ini yaitu dapat mendeteksi wajah berikut dengan identitas objek seperti NIM dan Nama objek

## 2. Dasar Teori

### 2.1 Citra

Citra (image) atau yang secara umum disebut gambar merupakan representasi spasial dari suatu objek yang sebenarnya dalam bidang dua dimensi yang biasanya ditulis dalam koordinat kartesian x-y, dan setiap koordinat merepresentasikan satu sinyal terkecil dari objek yang biasanya koordinat terkecil ini disebut sebagai pixel (picture element). Pixel (Picture Elements) adalah nilai tiap-tiap entri matriks pada bitmap. Rentang nilai-nilai pixel ini dipengaruhi oleh banyaknya warna yang dapat ditampilkan. Jika suatu bitmap dapat menampilkan 256 warna maka nilai-nilai pixel-nya dibatasi dari 0 hingga 255. 2.2 Citra Digital [5]

Citra digital dapat diperoleh dari penglihatan manusia terdiri dari dua komponen yaitu iluminasi dan reflektansi. Representasi citra secara matematis

dapat dituliskan sebagai berikut:

$$f(x,y) = i(x,y) \cdot r(x,y) \quad (2.1)$$

dimana:  $0 < i(x,y) < \infty$

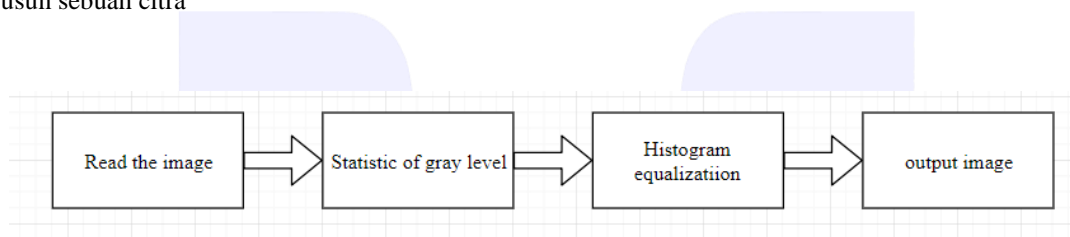
$r(x,y) < 1$

## 2.2 Teori Dasar Citra Digital

Citra digital [6] adalah suatu fungsi dua dimensi dari intensitas cahaya  $f(x,y)$  dimana  $x$  dan  $y$  menunjukkan koordinat spasial, nilai  $f$  pada koordinat  $(x,y)$  menyatakan tingkat keabuan dan informasi warna citra pada titik tersebut. Resolusi citra dapat menunjukkan seberapa detail suatu citra. Pengukuran resolusi citra dapat dilakukan dengan beberapa cara. Resolusi *pixel* dinyatakan dalam  $M \times N$  di mana  $M$  adalah jumlah kolom matriks atau lebar citra dan  $N$  adalah jumlah baris matriks atau tinggi citra. Resolusi *pixel* sekaligus menunjukkan jumlah *pixel* pembentuk suatu citra digital. Jumlah *pixel* per satuan panjang disebut resolusi spasial. Nilai intensitas suatu citra digital dibedakan ke dalam spektrum-spektrum dan jumlah spektrum yang digunakan disebut resolusi spektrum. Video sejatinya merupakan beberapa citra (yang juga disebut *frame*)

## 2.3 Histogram Equalization[10]

Histogram merupakan tampilan grafis yang merepresentasikan distribusi frekuensi dari masing-masing kategori pada suatu data tertentu. Pada citra digital, histogram dari sebuah citra menunjukkan distribusi frekuensi dari masing-masing nilai piksel atau derajat keabuan yaitu 0-255. Dengan kata lain, histogram citra dapat memberikan informasi tentang rentang nilai piksel dari sebuah citra dan nilai-nilai piksel mana yang dominan menyusun sebuah citra



Gambar 2.1 Histogram Equalization

## 2.4 Face Recognition System[11]

*Face recognition system* terdiri atas dua *processing* utama yaitu *face detection* dan *face recognition*. Mekanisme kerja *face detection* adalah dari citra yang diberikan, sistem akan mendeteksi apakah ada gambar wajah pada citra tersebut, jika memang terdeteksi gambar wajah maka kemudian sistem akan menentukan lokasi dan ukuran gambar wajah. Prinsip kerja *face recognition* adalah jika hasil *face detection* menyatakan terdapat gambar wajah, sistem akan membandingkannya dengan citra wajah yang sebelumnya telah disimpan di dalam *database*.

## 2.5 Local Binari Pattern Histograms (LBPH)

Local Binary Pattern Histogram (LBPH) adalah fitur untuk mengklasifikasi yang dikombinasikan dengan histogram dan merupakan teknik baru dari metode LBP untuk mengubah performa hasil pengenalan wajah. LBP pada umumnya didesain untuk pengenalan tekstur. LBPH adalah metode yang paling cocok untuk dilakukan pengenalan citra wajah untuk diimplementasikan pada perangkat bergerak android karena menggunakan penghitungan yang sederhana mengingat resource android yang terbatas [11].

## 2.6 Grayscale

Grayscale yaitu mengubah citra RGB menjadi citra keabuan yang digunakan untuk menyederhanakan model citra, sehingga waktu yang dibutuhkan untuk melakukan proses selanjutnya berjalan lebih cepat. Hal ini bisa dilakukan karena perhitungan yang terjadi hanya dilakukan pada satu layer saja, yaitu layer derajat keabuan.

## 2.7 Unified Modeling Language (UML)

UML merupakan bahasa standar untuk merancang dan mendokumentasikan perangkat lunak dengan cara yang berorientasi objek [14]

## 2.8 Android

Android adalah sistem operasi untuk perangkat mobile yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Android merupakan sebuah perangkat lunak untuk perangkat mobile, yang mana terdiri dari sebuah sistem operasi [15].

## 3. Perancangan Sistem

Dalam tugas akhir ini akan membahas tentang sistem deteksi wajah untuk presensi kehadiran menggunakan metode Local Binary Pattern Histogram (LBPH) berbasis android. Dengan memasukkan berupa foto wajah yang diambil melalui perangkat. Hasil dari tugas akhir ini merupakan sebuah aplikasi presensi kehadiran berbasis android

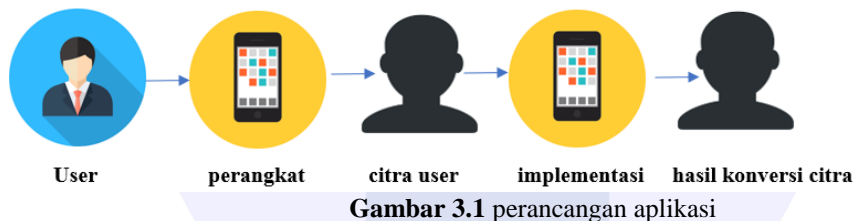
### 3.1 Analisa Kebutuhan Sistem

Pada tugas akhir ini, perangkat yang digunakan terdiri atas hardware (perangkat keras) dan software (perangkat lunak) dengan spesifikasi sebagai berikut:

1. Laptop dengan spesifikasi *Processor* Intel(R) Core i7~2.80 GHz, Memori 8 GB, dan ssd 120 GB.
2. *Handphone* dengan spesifikasi RAM 2 GB, Kamera 8 Megapixel, dan Android OS Lollipop (5.0).

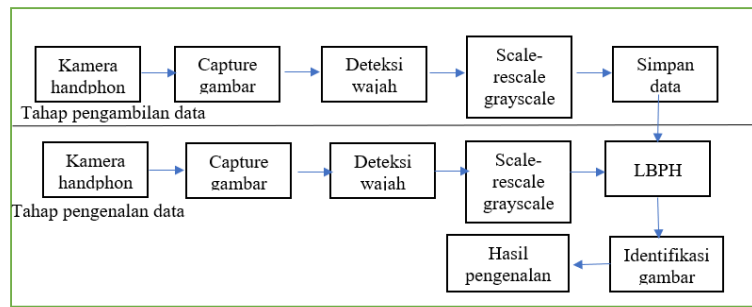
### 3.2 perancangan Aplikasi

Dalam tugas akhir ini dirancang sebuah sistem deteksi wajah. Gambaran perancangan aplikasi deteksi wajah seperti pada Gambar



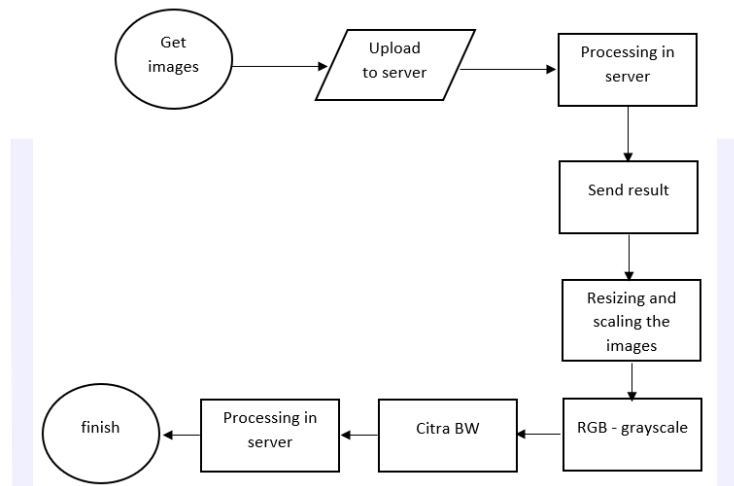
### 3.3 Perancangan sistem

dilakukan akuisisi citra dengan mengambil citra uji, lalu dilakukan tahap *pre-processing* terhadap citra uji kemudian dilakukan ekstraksi fitur dalam metode LBPH. Hasil dari ekstraksi dari ekstraksi fitur di klasifikasikan kedalam identitas objek yang sudah direkam saat citra uji berdasarkan data latih yang ada di *database*.



**Gambar 3.2** Gambaran Umum Sistem

### 3.4 Pre-Processing



**Gambar 3.3** Tahap Pre-processing

Proses pada tahap *pre-processing* pada Gambar 3.2 tahap pertama dalam proses ini yaitu Android get image, Setelah sistem mendapat masukan berupa image. Image Processing in Server. kemudian *resizing* dan *scaling* selanjutnya melakukan konversi RGB-grayscale dan Citra BW, lalu proses akhir adalah citra hasil pre-processing

### 3.5 Ekstraksi fitur

Ekstraksi fitur merupakan proses pengindeksan suatu database citra dengan isinya. Secara matematik, setiap ekstraksi fitur merupakan encode dari vektor  $n$  dimensi yang disebut dengan vektor fitur. Komponen vektor fitur dihitung dengan proses citra dan teknik analisis serta digunakan untuk membandingkan citra yang satu dengan citra yang lain. Ekstraksi fitur diklasifikasikan kedalam 3 jenis yaitu low level, middle level dan high level

### 3.6 Unified Modeling Language

Unified Modeling Language (UML) adalah metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek yang berfungsi untuk menederhanakan permasalahan yang kompleks agar lebih mudah dimengerti. Adapun jenis UML yang digunakan pada tugas akhir ini yaitu *use case diagram*, *activity diagram*, dan *sequence diagram*.

### 3.7 Tampilan Aplikasi

Tampilan aplikasi merupakan tampilan interface yang akan digunakan oleh user. Tampilan aplikasi dapat dilihat pada Gambar 3.3



Gambar 3.4 Tampilan Aplikasi

### 3.8 Parameter Pengujian

#### 1. Akurasi

Tingkat akurasi sistem.

Akurasi adalah kemampuan ketepatan sistem dalam menghitung sampel

$$Akurasi = \varphi = \frac{\text{jumlah citra yang total} - \text{jumlah citra yang salah}}{\text{jumlah citra yang salah}} \times 100\% \quad (3.1)$$

#### 2. Waktu komputasi

Waktu yang dibutuhkan untuk melakukan deteksi wajah menjadi dapat dikenal oleh sistem.

$$Waktu\ Komputasi = waktu\ selesai - waktu\ mulai \quad (3.2)$$

#### 3. False acceptance rate (FAR)

False acceptance rate (FAR) adalah kesalahan dalam mengenali identitas gambar masukan, baik itu kesalahan dalam mengenali identitas gambar masukan dari individu di luar database yang terdeteksi sebagai individu di dalam database, maupun kesalahan dalam mengenali identitas gambar masukan dari individu di dalam database yang dikenali sebagai individu lain.

$$FAR = \frac{\text{Banyak FAR}}{\text{jumlah percobaan}} \times 100\% \quad (3.3)$$

#### 4. False rejection rate (FRR)

False rejection rate (FRR) adalah kesalahan dalam menolak gambar masukan. Sebuah gambar masukan yang seharusnya dapat dikenali (identitasnya terdapat di dalam database) berubah menjadi tidak dikenali

$$FRR = \frac{\text{Banyak FRR}}{\text{jumlah percobaan}} \times 100\% \quad (3.4)$$

## 4 HASIL DAN PENGUJIAN

### 4.1 Pendahuluan

Pada bab ini berisi hasil dan pengujian sistem yang bertujuan untuk mengetahui hasil dari performansi sistem yang telah dibuat. Pengujian sistem ini dilakukan berdasarkan parameter yang didapatkan saat melakukan perancangan system

### 4.2 Pengujian terhadap hasil pre-processing

Pada tahap ini dilakukan pengujian citra wajah sebanyak 90 citra diantaranya 60 citra latih dan 30 merupakan citra yang akan diujikan

### 4.3 Pengujian Grid X dan Grid Y

**Tabel 4.1** Pengujian Grid X dan Grid Y

Ukuran GridX dan Grid Y	Akurasi (%)	Waktu Komputasi (detik)
2×2	64.44	1.45
4×4	71.11	1,53
8×8	95.56%	2.35

Dalam pengujian terhadap Grid X dan Grid Y, Grid X sebagai nilai jumlah sel dalam arah horizontal. Semakin banyak sel, semakin halus grid, semakin tinggi dimensi dari vektor fitur yang dihasilkan dan default value biasanya diatur ke 8

### 4.4 Pengaruh radius terhadap nilai akurasi

**Tabel 4.2** Pengaruh radius terhadap nilai akurasi

Radius	Akurasi (%)	Waktu Komputasi (detik)
1	95.56	2.35
2	72.22	3.48
3	57.78	4.51

Berdasarkan hasil pengujian yang ditunjukkan pada Tabel 4.2, menunjukan bahwa pengaruh radius pada angka 1 memiliki nilai akurasi yang paling tinggi adalah 95,56%. Semakin kecil ukuran window maka pengaruh median filter untuk mengurangi noise semakin sedikit sehingga hasil citra uji masih memiliki banyak noise. Sedangkan jika semakin besar maka pengaruh median filter untuk mengurangi noise semakin banyak sehingga informasi- informasi citra uji dianggap noise.



#### 4.5 Pengaruh threshold black and white nilai akurasi

threshold black and white merupakan proses konversi dari citra RGB normal menjadi citra black and white dengan citra yang diuji sebanyak 40 citra. Pengujian pengaruh nilai threshold black and white merupakan proses yang bertujuan untuk mencari nilai threshold yang akan dijadikan nilai biner. Pada pengujian tugas akhir ini, dilakukan pengujian terhadap nilai threshold yaitu 110, 150, 180, 210.

**Tabel 4.3** Pengaruh threshold black and white nilai akurasi

UkuranTresh hold BW	Akurasi (%)	Waktu Komputasi (detik)
110	26,54	1,45
150	43,58	1,52
180	95.56	2.35
210	95,56	2.92

#### 4.6 Pengujian terhadap Neighbors Pada LBPH

Pengujian terhadap *Neighbors* pada dilakukan untuk mengetahui performansi sistem. Pengujian ini dilakukan dengan cara mengubah nilai parameter N. Dimana nilai N yang digunakan parameter yaitu 1,3,6. *Neighbors* adalah Jumlah titik sampel untuk membangun pola biner Lokal yang melingkar. semakin banyak titik sampel yang dimasukkan, semakin lama sistem melakukan komputasi.

**Tabel 4.4** Hasil Pengujian Neighbors

Neighbors	Akurasi (%)	Waktu Komputasi (detik)
8	95.56	2.35
10	78.89	2.91
12	64.45	3.24

Berdasarkan hasil pengujian yang ditunjukkan pada Tabel 4.4, pengaruh parameter nilai N yang optimal adalah N=1 dengan nilai akurasi yang paling tinggi adalah 95.56%. Hal ini disebabkan karena cara kerja *Neighbors* mengambil keputusan berdasarkan tetangga terdekat. Semakin kecil nilai N, maka jarak tetangga yang diambil semakin dekat antara data yang ada di database dengan data yang diujikan sehingga hasilnya lebih akurat.

#### 4.7 Pengujian terhadap jarak

Pengujian terhadap jarak dilakukan untuk mengetahui sistem pada jarak antara objek dengan kamera yang paling optimal. Pengujian ini dilakukan dengan cara mengukur jarak antara jarak objek dengan kamera menggunakan sebuah penggaris. Jarak yang akan diujikan yaitu, 10 s.d 15 cm, 15 s.d 20 cm, dan 20 s.d 25 cm.

**Tabel 4.5** Hasil pengujian jarak pada pengambilan citra

Jarak (cm)	Akurasi (%)	Waktu Komputasi (detik)
10 s.d 15	81.11	3.40
15 s.d 20	87.78	2.84
20 s.d 25	95.56	2.35



Berdasarkan hasil pengujian yang ditunjukkan pada Tabel 4.5, menunjukan bahwa pengaruh jarak pengambilan objek yang paling optimal pada jarak 20 s.d 25 cm dengan akurasi 95.56% dan waktu komputasi 2.35 sekon.

#### 4.8 Hasil Pengujian terhadap default values

**Tabel 4.6** Hasil Pengujian default values

Gridx dan GridY	neighbors	radius	jarak	bw threshold	FAR	FRR	akurasi
8x8	8	1	20-25cm	180-210	3.33	1.11	95.56

Berdasarkan hasil pengujian aplikasi deteksi wajah dapat dilakukan dengan menggunakan metode *LBPH* (Local Binary Pattern Histogram) pada parameter Neighbors=9, Gridx. Gridy = 8x8, radius=1, ukuran tresh hold BW=180, dan jarak 20-25cm mendapatkan akurasi sebesar 95.56 dan waktu komputasi 2.35 detik.

### 5. Kesimpulan dan saran

#### 5.1 Kesimpulan

Berdasarkan hasil dan pengujian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Berdasarkan hasil pengujian aplikasi deteksi wajah dapat dilakukan dengan menggunakan metode *LBPH* (Local Binary Pattern Histogram) pada parameter Neighbors=9, Gridx. Gridy = 8x8, radius=1, ukuran tresh hold BW=180, dan jarak 20-25cm mendapatkan akurasi sebesar 95.56 dan waktu komputasi 2.35 detik.
2. Hasil pengujian menunjukan bahwa semakin kecil ukuran radius pada *LBPH* maka pengaruh untuk mengurangi noise semakin sedikit sehingga hasil citra uji masih memiliki banyak noise. Sedangkan jika semakin besar maka pengaruh dalam mengurangi noise semakin banyak sehingga informasi- informasi citra uji dianggap noise.
3. Berdasarkan hasil pengujian terhadap threshold, semakin kecil nilai threshold maka membuat informasi-informasi yang berada di citra wajah hilang. Sedangkan jika semakin besar nilai threshold maka kualitas citra wajah rusak menjadi warna hitam dan tidak bisa menggunakan ekstraksi fitur wajah.
4. Jarak pengambilan gambar yang paling optimal yaitu 20 sampai 25 cm. Apabila citra yang diambil semakin dekat maka citra akan *blur*. Jika pengambilan citra terlalu jauh maka citra yang tidak dapat terdeteksi.

#### 5.2 Saran

Berdasarkan hasil dan pengujian yang telah dilakukan, maka dapat diambil saran sebagai berikut:

1. Menggunakan Grid x, Grid Y, Neighbors, dan Thresh hold yang sudah ditentukan oleh *module face recognition* yang telah ditentukan oleh Philipp Wagner dalam *default values* yang sudah ada.
2. Menggunakan *platform* Android Studio karena lebih sinkron terhadap *processor* HP Android versi 5.
3. Tidak menggunakan *platform* ADT karena tidak *support* terhadap OpenCV dalam sinkronisasi *face recognition* menggunakan metode *LBPH* face recognition.

**DAFTAR PUSTAKA**

- [1] E. Wahyudi, Wirawan dan H. Kusuma., “Perbandingan Unjuk Kerja Pengenalan Wajah Berbasis Fitur Local Binary Pattern dengan Algoritma PCA dan Chi Square. Jurusan Teknik Elektro ITS, 2011
- [2] E. Vidyaningrum dan Prihandoko., “Human Face Detection by using eigenface method for various pose of human face. Fakultas Teknologi Industri, Universitas Gunadarma. 2009
- [3] D. Esti Pratiwi dan A. Harjoko., “Implementasi Pengenalan Wajah Menggunakan PCA (Principal Component Analysis). Yogyakarta: Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, 2013
- [4] H. Jeklin., “Perancangan Aplikasi Unlock Screen Dengan menggunakan Face Expression Pada Smartphone Berbasis Android. Tesis. Pascasarjana Universitas Bina Nusantara, 2014
- [5] T. Ahonen, Abdenour Hadid dan Matti Pietikainen. 2006. Face Description with Local Binary Pattern: Application to Face Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence,
- [6] P. Harini, Meika., "Pengenalan Pola Wajah Manusia Menggunakan Transformasi Wavelet dan LDA (Linear Discriminant Analysis)". Tugas Akhir. Sekolah Tinggi Teknologi Telkom, 2007
- [7] M. Leni., "Aplikasi Pendeteksi RAS Kucing Dengan Mendeteksi Wajah Kucing Dengan Metode Viola Jones Berbasis Android". Tesis. Pascasarjana Unikom Bandung, 2016
- [8] P. Dharma., "Pengolahan Citra Digital". Yogyakarta, 2010
- [9] K., Laila., "Perancangan Dan Implementasi Face Recognition System Menggunakan Eigenface Sebagai Fungsi Unlock Screen Pada Handset Android". Tugas Akhir. Institut teknologi Telkom 2012
- [10] R. Kruttsch dan D. Tenorio.. "Histogram Equalization", Freescale Semiconductor, Inc. 2011
- [11] T. Filareti(dkk).. Face Recognition - Face Detection, Global Approaches for, Feature Based Techniques, Problem and Considerations, Conclusions and Features Development, 2011
- [12] S. A. Wijaya, “Perbandingan Metode Pengenalan Wajah Secara Real Time Pada Perangkat Bergerak Berbasis Android”. 2014
- [13] N. S. H, Android Pemrograman Aplikasi Mobile Smartphone Dan Tablet PC berbasis Android, Bandung: Informatika, 2012.
- [14] Salton, Kelvin 2017. <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b/2018/04>, 25 April 2018

