

**IMPLEMENTASI METODE FIRE DAN IMAGE PROCESSING
CITRA MATA UNTUK MENDETEKSI POLA IRIS PADA
PROSES AUTENTIKASI SMARTPHONE**

SKRIPSI

**OLEH
AHMAD ZIDAN NUR HAKIM
NIM. 18610070**



**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

**IMPLEMENTASI METODE FIRE DAN IMAGE PROCESSING
CITRA MATA UNTUK MENDETEKSI POLA IRIS PADA
PROSES AUTENTIKASI SMARTPHONE**

SKRIPSI

**Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Matematika (S.Mat)**

**Oleh
Ahmad Zidan Nur Hakim
NIM. 18610070**

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

**IMPLEMENTASI METODE FIRE DAN IMAGE PROCESSING
CITRA MATA UNTUK MENDETEKSI POLA IRIS PADA
PROSES AUTENTIKASI SMARTPHONE**

SKRIPSI

**Oleh
Ahmad Zidan Nur Hakim
NIM. 18610070**

Telah Disetujui untuk Diuji
Tanggal 16 Juni 2022

Dosen Pembimbing I



Hisyam Fahmi, M.Kom
NIP. 19890727 201903 1 018

Dosen Pembimbing II



Erna Herawati, M.Pd
NIDT. 19760723 20180201 2 222

Mengetahui,
Ketua Program Studi Matematika



Dr. Elly Susanti, M.Sc
NIP. 19741129 200012 2 005

**IMPLEMENTASI METODE FIRE DAN IMAGE PROCESSING
CITRA MATA UNTUK MENDETEKSI POLA IRIS PADA
PROSES AUTENTIKASI SMARTPHONE**

SKRIPSI

Oleh
Ahmad Zidan Nur Hakim
NIM. 18610070


Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal 21 Juni 2022

Ketua Penguji : Muhammad Khudzaifah, M.Si

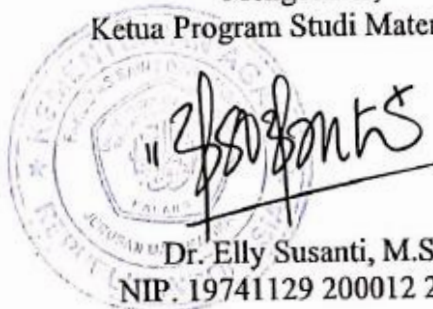
Anggota Penguji 1 : Mohammad Nafie Jauhari, M.Si

Anggota Penguji 2 : Hisyam Fahmi, M.Kom

Anggota Penguji 3 : Erna Herawati, M.Pd



Mengetahui,
Ketua Program Studi Matematika



Dr. Elly Susanti, M.Sc
NIP. 19741129 200012 2 005

PERNYATAN KEASLIAN TULISAN

Saya yang bertandatangan di bawah ini:

Nama : Ahmad Zidan Nur Hakim

NIM : 18610070

Program Studi : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Implementasi Metode FIRE dan *Image Processing* Citra
Mata untuk Mendeteksi Pola Iris pada Proses Autentikasi
Smartphone

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan dan pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan atau daftar rujukan. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 13 Juni 2022
Yang membuat pernyataan,



Ahmad Zidan Nur Hakim
NIM. 18610070

MOTO DAN PERSEMBAHAN

“Pemenang adalah orang-orang biasa dengan hati yang luar biasa”

Skripsi ini dipersembahkan kepada:

Orang tua penulis bapak Bambang Irawan, M.Pd dan ibu Nur Holisah Hamim, S.Pt,
yang selalu mencurahkan segala bentuk cinta kasihnya tanpa meminta balasan,
serta adik tersayang Kharisma Nur Maulidia yang selalu memberi dukungan dan
doa-doa terbaik kepada penulis.

KATA PENGANTAR

Dengan memanjatkan rasa syukur kehadirat Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga penulis dapat menyelesaikan pengerjaan tugas akhir yang berjudul “Implementasi Metode FIRE dan *Image Processing* Citra Mata untuk Mendeteksi Pola Iris pada Proses Autentikasi *Smartphone*”, sebagai salah satu syarat untuk menyelesaikan Program Sarjana (S1) serta mendapatkan gelar sarjana matematika di Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.

Dalam proses penyusunan skripsi ini, penulis menerima banyak bimbingan, masukan dan arahan dari berbagai pihak. Oleh karena itu, melalui halaman ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Dr. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Elly Susanti, S.Pd., M.Sc, selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim.
4. Hisyam Fahmi, M.Kom, selaku dosen pembimbing I yang telah memberikan banyak ilmu serta arahan kepada penulis.
5. Erna Herawati, M.Pd, selaku dosen pembimbing II yang telah memberikan banyak ilmu serta arahan integrasi agama kepada penulis.
6. Dewi Ismiarti, M.Si, selaku dosen yang mengarahkan penulis untuk menghasilkan skripsi dengan format yang baik dan benar.
7. Juhari, M.Si, selaku dosen wali yang selalu membimbing agar penulis memiliki banyak ide serta gambaran terkait topik penelitian yang bermanfaat bagi sesama.
8. Seluruh civitas akademika Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
9. Kedua orang tua saya, bapak Bambang Irawan, M.Pd, ibu Nur Holisah Hamim, S.Pt dan adik perempuan saya Kharisma Nur Maulidia serta seluruh keluarga.

10. Seluruh mahasiswa angkatan 2018 yang selalu memberikan dukungan terbaik bagi penulis.
11. Serta ucapan terima kasih kepada beberapa tempat yang telah memberikan fasilitas pada penulis untuk menyelesaikan penelitian ini diantaranya yaitu KOPI STUDIO 24 Blimbing, STARBUCKS Coffee, NAKOA Soekarno Hatta, AADK Coffee, dan McDonald's Mt Haryono.

Semoga skripsi yang berjudul “Implementasi Metode FIRE dan *Image Processing* Citra Mata untuk Mendeteksi Pola Iris pada Proses Autentikasi *Smartphone*” yang telah penulis buat akan memberikan banyak manfaat bagi penulis sendiri serta pembaca. Aamiin

Malang, 10 Maret 2022

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGANTAR	ii
PERSETUJUAN.....	iii
HALAMAN PENGESAHAN	iv
PERNYATAN KEASLIAN TULISAN	v
MOTO DAN PERSEMBAHAN	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
DAFTAR SIMBOL	xiii
DAFTAR LAMPIRAN	xiv
ABSTRAK	xv
ABSTRACT	xvi
مستخلص البحث.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat Penelitian.....	4
1.5 Batasan Masalah	4
BAB II KAJIAN TEORI	6
2.1 Teori Pendukung	6
2.1.1 Citra Digital	6
2.1.2 Pengolahan Citra Digital (<i>Digital Image Processing</i>).....	7
2.1.3 Deteksi Tepi.....	8
2.1.4 FIRE (<i>Fast Iris Recognition</i>).....	10
2.1.5 Iris Mata.....	12
2.1.6 Segmentasi Iris	13
2.1.7 Operator <i>Integro-Differential</i>	14
2.1.8 Transformasi Hough	21
2.1.9 Fitur <i>Encoding</i>	23
2.1.10 Algoritma <i>Hamming Distance</i>	26
2.1.11 <i>Weight Euclidean Distance</i>	27
2.1.12 Normalisasi <i>Template Iris</i>	28
2.1.13 Cross-Validation.....	30
2.1.14 Autentikasi <i>Smartphone</i>	30
2.2 Kajian Integrasi Topik Dengan Al-Quran/Hadits.....	31
2.3 Kajian Topik Dengan Teori Pendukung.....	32
BAB III METODE PENELITIAN	35
3.1 Jenis Penelitian	35
3.2 Data dan Sumber Data.....	35
3.3 Teknik Pengumpulan Data	35

3.4 Instrumen Penelitian	36
3.5 Teknik Analisis Data	37
3.5.1 Tahap <i>Preprocessing</i>	39
3.5.2 Tahapan Segmentasi Iris Mata	39
3.5.3 Tahapan Perbandingan (<i>Comparison</i>)	40
3.5.4 Tahapan Evaluasi.....	41
BAB IV HASIL DAN PEMBAHASAN	42
4.1 Tahap <i>Preprocessing</i>	42
4.2 Iris Segmentation	43
4.2.1 Mendeteksi dan <i>Mapping</i> Refleksi Iris Mata	43
4.2.2 Menghapus Refleksi	47
4.2.3 Menentukan Pupil.....	49
4.2.4 Menentukan Batas Iris	55
4.2.5 Mendeteksi Bulu Mata.....	58
4.2.6 Normalisasi Iris Mata	60
4.2.7 Fitur <i>Encoding</i>	63
4.3 Perbandingan Iris	66
4.4 Evaluasi	69
BAB V PENUTUP.....	73
5.1 Kesimpulan.....	73
5.2 Saran untuk Penelitian Lanjutan.....	73
DAFTAR PUSTAKA	75
LAMPIRAN.....	77
RIWAYAT HIDUP	79

DAFTAR TABEL

Tabel 2.1 Keterangan Rumus <i>Integro-Differential</i>	14
Tabel 2.2 Keterangan <i>Hamming Distance</i>	26
Tabel 2.3 Keterangan Rumus <i>Euclidean Distance</i>	28
Tabel 3.1 Proses Segmentasi Iris	40
Tabel 4.1 Hasil <i>Hamming Distance</i> dan <i>Euclidean Distance</i>	68
Tabel 4.2 Hasil Pengujian <i>Template</i> Iris	69

DAFTAR GAMBAR

Gambar 2.1	Posisi Koordinat Citra Digital	8
Gambar 2.2	Tepi Curam.....	9
Gambar 2.3	Citra Dengan Tepi Curam	9
Gambar 2.4	Tepi Landai	9
Gambar 2.5	Citra Dengan Tepi Landai	10
Gambar 2.6	Tepi Yang Memiliki Noise.....	10
Gambar 2.7	Anatomi Mata Manusia	13
Gambar 2.8	<i>Euclidean Distance</i>	27
Gambar 2.9	<i>K-Fold Cross-Validation</i>	30
Gambar 3.1	Diagram Alur <i>Input</i> Citra Acuan Pada Database	38
Gambar 3.2	Diagram Alur Proses <i>Iris Recognition</i>	38
Gambar 4.1	Citra <i>Grayscale</i>	42
Gambar 4.2	<i>Threshold Mapping</i> Refleksi.....	45
Gambar 4.3	Peta Refleksi.....	46
Gambar 4.4	Titik-Titik Refleksi dan Batas Titik Refleksi	47
Gambar 4.5	Deteksi Pupil Asli dan Palsu	51
Gambar 4.6	Deteksi Pupil	52
Gambar 4.7	Pupil Asli.....	54
Gambar 4.8	Pupil Iris Mata Terdeteksi	54
Gambar 4.9	Mendeteksi Lingkaran Iris Dengan Algoritma <i>Starburst</i>	55
Gambar 4.10	Proses Mencari Iris	56
Gambar 4.11	Mencari Posisi Iris Pada Piksel Sekitar	57
Gambar 4.12	Mencari Iris Dengan Kontur Area Sekitar	57
Gambar 4.13	Tahapan Mencari Iris Mata	58
Gambar 4.14	Mendeteksi Bulu Mata	60
Gambar 4.15	Proses <i>Unwrapping</i> Iris Mata.....	61
Gambar 4.16	Perbedaan Citra Iris Yang Telah Bersih Dari Bulu Mata.....	62
Gambar 4.17	Peningkatan Kualitas Citra <i>Template</i> Iris.....	63
Gambar 4.18	Representasi Fasor.....	63
Gambar 4.19	Representasi Citra <i>Template</i> Iris	64
Gambar 4.20	Nilai Biner <i>Template</i> Iris.....	65
Gambar 4.21	Total Waktu Segmentasi	66
Gambar 4.22	Perbandingan Dua <i>Template</i> Tanpa Pergeseran.....	68
Gambar 4.23	Hasil Perbandingan <i>Template</i> Iris	69

DAFTAR SIMBOL

r	: Jari-jari lingkaran iris
$G_{\sigma}(r)$: Fungsi Gaussian <i>Smoothing</i> dengan skala σ
$I(x, y)$: Intensitas piksel citra iris
ds	: Batas kontur lingkaran
β_s	: Nilai intensitas rata-rata citra iris
$ \mathcal{E}f ^p$: Citra yang dibentuk dengan menaikkan nilai dari setiap titik
ψ_i	: Lambang potongan isi
ψ_p	: Lambang potongan pengisi
d_{SSD}	: Jarak SSD
$E_i^{\vec{d}}(x, y)$: Istilah <i>energy regional</i> untuk area tertentu
$D_i^{\vec{d}}$: Nilai intensitas tingkat keabuan dari piksel (x, y)
T	: Nilai ambang batas (<i>threshold</i>)
C_p	: Lingkaran calon pupil yang terdeteksi
H	: Ukuran dispersi didalam calon pupil C_p
$I'_{x,y}$: Intensitas citra dalam calon pupil pada koordinat (x, y)
XOR atau \oplus	: Operator <i>XOR</i> mendeteksi perbedaan antara setiap pasang bit
d_{ij}	: Jarak <i>Euclidean</i> entri ke- i dan entri ke- j matriks citra mata

DAFTAR LAMPIRAN

Lampiran 1 : Proses Segmentasi Citra Iris Mata

Lampiran 2 : Proses Perbandingan Antar *Template* Iris

ABSTRAK

Hakim, Ahmad Zidan Nur. 2022. **Implementasi Metode FIRE dan *Image Processing* Citra Mata untuk Mendeteksi Pola Iris pada Proses Autentikasi *Smartphone***. Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing (1): Hisyam Fahmi, M.Kom, Pembimbing (2): Erna Herawati, M.Pd.

Kata Kunci : Citra Digital, *Iris Recognition*, FIRE, *Smartphone*

Perkembangan teknologi sensor gambar pada *smartphone* yang terus meningkat menyebabkan penyebaran sistem pengenalan biometrik pada perangkat baru semakin cepat. *Smartphone* memiliki keunggulan utama yaitu *portabel*, proses komputasi yang canggih, dan dilengkapi dengan kamera resolusi tinggi. Hal tersebut membuat *smartphone* dapat memiliki fasilitas autentikasi dengan aman dan akurat yang dapat dilakukan kapan saja dan di mana saja. Pandemi COVID-19 mengharuskan seluruh masyarakat menggunakan masker, sehingga sistem pengenalan wajah atau Face Id dianggap dapat membahayakan pengguna *smartphone* karena mengharuskan membuka masker. Dengan adanya kendala tersebut, pada penelitian ini mengembangkan sistem biometrik *Iris Recognition* dengan menggunakan metode *Image Processing* dan dikombinasikan dengan metode *Fast Iris Recognition* (FIRE). Alasan penggunaan dua metode tersebut yaitu adanya hasil yang lebih baik dalam aspek akurasi dan kecepatan pada proses komputasi dengan perangkat *hardware* yang memiliki spesifikasi rendah. Uji coba dilakukan menggunakan dataset CASIA V1 yang berjumlah 756 citra iris. Hasil evaluasi dengan menggunakan *k-fold cross validation* menghasilkan nilai akurasi 97,33% dan rata-rata waktu 0,67 detik.

ABSTRACT

Hakim, Ahmad Zidan Nur. 2022. **The Implementation of FIRE Method and Image Processing of Iris Image's to Detect Iris Pattern on Smartphone's Authentication Program.** Thesis. Mathematic Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisor (1): Hisyam Fahmi, M.Kom, Supervisor (2): Erna Herawati, M.Pd.

Keywords: Digital Image, Iris Recognition, FIRE, Smartphone

The development of image sensor technology on smartphone that increases continuously cause the spread of biometric authentication system on new devices faster. Smartphone has the main excellence, it is portable which means an advanced computation process, and equipped with high resolution camera. That matters made smartphone able to have a safe and accurate authentication facility that can be used everywhere and every time. The Covid-19 pandemic has obliged every people to use mask, so that the facial recognition system of Face ID is considered to be dangerous for the smartphone's user because they have to open the mask before using smartphone. Due to that obstacle, this research is developing biometric system for Iris Recognition by using Image Processing method and combined with another method called Fast Iris Recognition (FIRE). The background of using those two method is the better the result on accuracy and velocity on computation process with hardware that has low specifications. The test has done by using dataset CASIA V1 that has 756 amounts of iris images. The evaluation using k-fold cross validation was resulting accuracy value on 97.33% and the average times on 0.67 seconds.

مستخلص البحث

حاكم، أحمد زيدان نور. 2022. تطبيق طريقة *FIRE* ومعالجة صورة صور العين لكشف نمط قزحية العين في عملية المصادقة في الهاتف الذكي. البحث الجامعي. قسم الرياضيات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف (1): هشام فهمي، الماجستير، المشرف (2): أيرنا هيراواتي، الماجستير .

الكلمات المفتاحية: تجترا الرقمي، التعرف على القزحية، *FIRE*، الهاتف الذكي

تطور التكنولوجيا مستشعر الصرة على الهاتف الذكي بتطور وسيع يسبب إلى انتشار نظام التعرف على المقياس الحيوي على الجهاز الجديد بشكل أسرع. والهاتف الذكي بالمزايا الرئيسية لكونها محمولة والحوسبة المتقدمة ومجهزة بكاميرة عالية الدقة. ويجعل الهاتف الذكي الحصول على تسهيل مصادقة آمنة ودقيقة يمكن إجراؤها في أي وقت وفي أي مكان. ويتطلب وباء كوفيد-19 من جميع الأشخاص ارتداء القناع، لذلك يعتبر نظام التعرف على الوجه أو *Face ID* محطراً على الهاتف الذكي لأنه يتطلب فتح قناع. بالنظر إلى هذه المشكلة، سيطور الباحث نظام القياسة الحيوية للتعرف على قزحية العين بطريقة معالجة الصورة، ودجه بالتعرف السريع على القزحية (*FIRE*). وسبب استخدام هاتين الطريقتين هو أن هناك حاصل أفضل من حيث الدقة والسرعة في عملية الحوسبة بالجهاز ذو المواصفات المنخفضة. وإجراء التجربة باستخدام مجموعة بيانات *CASIA VI* ، والتي تتكون من 756 صورة قزحية. ونتيجة التقييم باستخدام التحقق المتقاطع *k-fold* بقيمة الدقة 97.33% ومتوسط وقت 0.67 دقيقة.

BAB I PENDAHULUAN

1.1 Latar Belakang

Kebutuhan penggunaan, penyimpanan, dan layanan data yang aman serta digabungkan dengan perkembangan teknologi sensor gambar pada *smartphone* yang terus meningkat menyebabkan penyebaran sistem pengenalan biometrik pada perangkat baru semakin cepat. *Smartphone* memiliki keunggulan utama yaitu *portabel*, proses komputasi yang canggih, dan dilengkapi dengan kamera resolusi tinggi. Hal tersebut membuat *smartphone* dapat memiliki fasilitas autentikasi dengan aman dan akurat yang dapat dilakukan kapan saja dan di mana saja.

(Wardani, 2020) menyatakan bahwa autentikasi menggunakan Face Id atau pengenalan wajah memang lebih mudah daripada penggunaan pin atau *password* bagi pengguna *smartphone* ketika belum terjadi pandemi COVID-19. Namun, hal tersebut berbeda semenjak adanya pandemi COVID-19 yang mengharuskan masyarakat untuk menggunakan masker ketika berada diluar rumah. Oleh karena itu, penggunaan *Face Id* atau pengenalan wajah untuk melakukan autentikasi *smartphone* dianggap tidak efektif dan tidak direkomendasikan. Saat ini sistem biometrik *smartphone* pada umumnya menggunakan *fingerprint* dan *face recognition*.

Bagi *smartphone* keluaran terbaru, pemindai sidik jari (*fingerprint*) sudah banyak tidak dipergunakan dengan alasan mudah untuk di duplikasi. Namun, dengan menerapkan teknologi *face recognition* di masa pandemi seperti ini juga tidak menjadi sebuah solusi untuk membuka kunci *smartphone*.

Berkaitan dengan hal tersebut, pada penelitian ini akan dikaji sistem autentikasi *iris recognition* yang memiliki tingkat keamanan tinggi serta memiliki aspek fleksibilitas pada pengguna *smartphone*. *Iris recognition* hadir dengan berbagai keunggulan di mana dapat diterapkan pada seluruh *smartphone* tanpa membuat pengguna membuka masker terlebih dahulu. Karena sistem ini mengambil data mata dengan bantuan sensor inframerah sehingga resiko pengguna *smartphone* untuk tertular COVID-19 semakin menurun. Namun, ada sejumlah masalah terkait dengan pengenalan iris pada *smartphone*, seperti citra di luar fokus (*blur*), pantulan cahaya, dan citra iris beresolusi rendah yang bisa dikatakan citra tersebut memiliki *noise*.

Dengan adanya kendala tersebut, pada penelitian ini akan menggunakan metode *image processing* yang dikombinasikan dengan metode *Fast Iris Recognition* (FIRE). Alasan penggunaan metode tersebut yaitu adanya hasil yang lebih baik dalam aspek akurasi dan kecepatan pada proses komputasi dengan perangkat *hardware* yang memiliki spesifikasi rendah. Oleh karena itu, kedua metode tersebut akan mudah diimplementasikan pada *smartphone* yang memiliki teknologi dengan kemampuan yang terbatas dibandingkan dengan *Personal Computer* (PC) atau Laptop.

Ajaran agama Islam telah mengajarkan tentang pentingnya mengembangkan ilmu pengetahuan pada setiap individu. Dalam Al-Quran dijelaskan bahwa manusia memang terlahir dalam kondisi yang tidak mengetahui apapun, yaitu terdapat di dalam surat an-Nahl ayat 78 yang berbunyi:

وَاللَّهُ أَخْرَجَكُمْ مِنْ بُطُونِ أُمَّهَاتِكُمْ لَا تَعْلَمُونَ شَيْئًا وَجَعَلَ لَكُمُ السَّمْعَ وَالْأَبْصَارَ وَالْأَفْئِدَةَ ۗ
لَعَلَّكُمْ تَشْكُرُونَ

“Dan Allah mengeluarkan kamu dari perut ibumu dalam keadaan tidak mengetahui sesuatu pun, dan Dia memberimu pendengaran, penglihatan, dan hati nurani, agar kamu bersyukur” (QS. An-Nahl:78).

Penjelasan ayat tersebut dalam Kitab Tafsir Al-Misbah (M. Quraish Shihab, 2002) yakni, pada dasarnya manusia terlahir dalam kondisi tidak mengetahui apapun. Sehingga Allah SWT memberikan potensi dasar berupa indra pendengaran, penglihatan, dan hati atau akal untuk memperoleh pengetahuan tersebut. Hal ini menandakan bahwa, pada penelitian kali ini secara tidak langsung akan mengembangkan ilmu pengetahuan semakin luas.

Pada penelitian sebelumnya, metode FIRE merupakan sebuah pendekatan baru untuk proses pengenalan iris mata pada citra mata. Salah satu keunggulan metode FIRE yaitu dapat mendeteksi iris pada citra mata yang memiliki *noise* dengan waktu singkat dan akurasi yang baik. FIRE telah diuji pada database besar, yaitu MICHE I DB yang tersusun dari kumpulan citra iris yang didapatkan oleh berbagai *smartphone* berbeda. FIRE pertama kali dipresentasikan pada evaluasi MICHE II yang diadakan pada tahun 2016 (Galdi, 2017).

Pada penelitian ini, peneliti menggabungkan metode *image processing* dengan FIRE untuk mendeteksi iris pada citra mata. *Image Processing* mampu mengolah gambar (citra) yang ditransformasikan menjadi citra lain. Dengan melakukan penggabungan metode *Image Processing* dan metode FIRE diharapkan menghasilkan proses pendeteksi iris mata yang lebih akurat, serta total waktu proses komputasi yang semakin cepat. Data citra yang diproses akan melalui ekstraksi warna, tekstur dan sistem akan mencari pola iris pada citra tersebut. Oleh karena itu, penelitian ini diharapkan dapat menghasilkan pola iris mata yang akurat serta dapat memangkas waktu proses komputasinya.

1.2 Rumusan Masalah

Berdasarkan penjelasan pada latar belakang, rumusan masalah yang diangkat dalam penelitian ini yaitu, bagaimana hasil akurasi dan efisiensi waktu penggunaan metode FIRE dan *Image Processing* dalam pengenalan iris mata menggunakan *library* OpenCV dan IrisSeg (*Iris Segmentation*)?

1.3 Tujuan Penelitian

Tujuan penelitian ini yaitu mengetahui hasil akurasi dan efisiensi waktu dari implementasi metode FIRE dan *Image Processing* dalam pengenalan iris mata pada proses autentikasi *smartphone*.

1.4 Manfaat Penelitian

Penelitian ini diharapkan dapat memberi manfaat sebagai berikut:

1. Memudahkan pengguna *smartphone* dalam proses autentikasi tanpa membuka masker.
2. Mempersingkat waktu komputasi saat proses autentikasi *smartphone* dengan menggunakan sistem biometrik *iris recognition*.

1.5 Batasan Masalah

Berdasarkan latar belakang tersebut, maka batasan masalah dalam penelitian ini adalah :

1. Penelitian ini berfokus pada pengembangan sistem biometrik iris mata yang disimulasikan menggunakan *library* OpenCV, Keras, dan Iris Segmentation pada bahasa pemrograman *python*.

2. Data yang diolah diperoleh dari dataset CASIA V1 dengan jumlah 756 citra mata, di mana setiap citra tersebut memiliki dimensi piksel 320x280 dan warna *grayscale*. Serta format file *.jpg*, atau *.jpeg*
3. Batasan penelitian ini menggunakan metode FIRE dan *image processing* yang dibantu dengan algoritma *integro-differential*. Dengan melalui ekstraksi fitur iris dengan menggunakan transformasi Hough dan *integro-differential* lalu mencari iris yang serupa dengan menggunakan Filter Log-Gabor, algoritma *Hamming Distance*, dan *Euclidean Distance*.

BAB II KAJIAN TEORI

2.1 Teori Pendukung

2.1.1 Citra Digital

Menurut KBBI, citra memiliki arti gambar, di mana citra terdiri dari dua jenis, yaitu citra kontinu dan citra diskrit. Citra kontinu adalah citra yang dihasilkan dari sistem optik yang menerima sinyal analog. Seperti contoh, citra kontinu didapatkan dari kamera analog dan mata manusia. Sedangkan citra diskrit merupakan citra yang didapatkan dari proses digitalisasi citra kontinu. Citra diskrit juga biasa disebut sebagai citra digital. Citra digital merupakan gambar dua dimensi yang dapat dilihat serta ditampilkan pada layar monitor *desktop* atau komputer dengan himpunan berhingga nilai digital atau biasa disebut piksel. Piksel adalah singkatan dari *picture element*, dimana tiap piksel memiliki nilai yang dapat menunjukkan suatu intensitas warna. Proses digitalisasi citra analog terdiri dari, penerokan dan kuantisasi. Definisi penerokan adalah pembagian citra pada tiap elemen suatu piksel, sedangkan kuantisasi merupakan pemberian suatu nilai intensitas warna pada tiap piksel, dengan nilai yang berupa bilangan bulat (Awcock, 1996).

Jumlah nilai yang dapat digunakan dalam proses kuantisasi suatu citra bergantung pada setiap kedalaman piksel yang dimiliki, yakni berupa banyaknya bit yang dapat digunakan untuk menampilkan intensitas warna pada piksel. Nilai kedalaman pada tiap piksel dapat disebut juga kedalaman warna. Dimana citra digital yang memiliki kedalaman warna piksel n-bit dapat disebut citra n-bit.

Dengan berdasarkan jenisnya, suatu citra digital dapat dibagi menjadi tiga yaitu citra biner (monokrom), citra *grayscale*, citra berwarna (Sutoyo, 2009).

2.1.2 Pengolahan Citra Digital (*Digital Image Processing*)

Pengolahan citra digital atau *Digital Image processing* merupakan suatu metode yang dapat digunakan untuk memproses atau memanipulasi citra digital 2 dimensi. Citra yang baik merupakan sebuah citra yang memiliki kualitas tinggi, dimana citra tersebut memiliki sejumlah informasi yang detail dan jelas sesuai dengan gambar asli. Namun, citra dapat mengalami penurunan kualitas seperti cacat, terlalu kontras, kurang tajam, dan lain sebagainya. Sehingga, pengolahan citra digital dapat dikatakan sebagai segala operasi untuk memperbaiki, menganalisa, atau mengubah suatu gambar (Gonzalez, 2002).

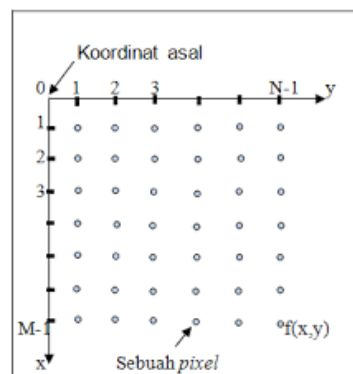
Pengolahan citra digital dipergunakan untuk membantu mengamati serta menganalisis suatu objek tanpa adanya hubungan atau kontak langsung dengan objek yang sedang diamati, sehingga dapat dipergunakan sebagai alat bantu untuk mengevaluasi mutu suatu produk tanpa merusak produk itu sendiri (Suhandy, 2007).

Pada umumnya, objektivitas dari pengolahan citra digital adalah mentransformasi atau menganalisis suatu gambar sehingga informasi baru tentang gambar dibuat lebih jelas. Ada empat klasifikasi dasar dalam pengolahan citra yaitu *point*, *area*, geometrik, dan *frame* (Gonzalez, 2002). Proses digitalisasi sebuah citra merupakan proses mempresentasikan citra yang berawal dari fungsi kontinu menjadi nilai diskrit. Setelah proses digitalisasi, maka citra digital yang telah terbentuk memiliki dimensi ukuran tinggi x lebar.

Agar dapat diproses dan diolah dengan komputer, maka citra direpresentasikan secara numerik dengan nilai-nilai diskrit. Citra yang direpresentasikan dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi. Sehingga dari proses tersebut, akan dihasilkan sebuah citra digital. Secara umum, citra digital memiliki bentuk empat persegi panjang yang dimensi ukurannya dinyatakan sebagai tinggi (M) * lebar (N). Citra digital yang memiliki ukuran dimensi $N \times M$ dapat dinyatakan dengan matriks yang memiliki ukuran N baris dan M kolom sebagai berikut:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(1, M - 1) \\ f(1,0) & f(1,1) & \dots & f(1, M - 1) \\ \dots & \dots & \dots & \dots \\ f(N - 1,0) & f(N - 1,1) & \dots & f(N - 1, M - 1) \end{bmatrix}$$

Posisi koordinat citra digital $N \times M$ direpresentasikan sebagai berikut:



Gambar 2.1 Posisi Koordinat Citra Digital

Diketahui dari Gambar 2.1, terdapat sebanyak $N \times M$ posisi diskrit. Setiap posisi diskrit tersebut memiliki unsur gambar atau piksel.

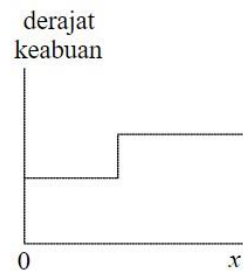
2.1.3 Deteksi Tepi

Deteksi Tepi (*Edge Detection*) dapat didefinisikan sebagai suatu batas diantara dua wilayah. Kedua wilayah tersebut memiliki nilai intensitas yang berbeda. Sehingga deteksi tepi merupakan sebuah proses untuk mendeteksi garis

tepi yang membatasi antar dua wilayah citra yang bersifat homogen dan memiliki tingkat nilai kecerahan yang berbeda.

Terdapat tiga jenis *edge* yang terdapat pada suatu citra digital, yaitu:

1. Tepi curam, karena memiliki nilai perubahan intensitas yang sangat tajam, nilai tersebut berkisar 90° .

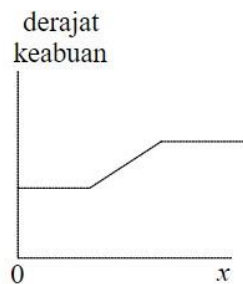


Gambar 2.2 Tepi Curam

$$\begin{bmatrix} 4 & 4 & 4 & 8 & 8 & 8 & 8 & 8 \\ 4 & 4 & 4 & 8 & 8 & 8 & 8 & 8 \\ 4 & 4 & 4 & 8 & 8 & 8 & 8 & 8 \\ 4 & 4 & 4 & 8 & 8 & 8 & 8 & 8 \\ 4 & 4 & 4 & 8 & 8 & 8 & 8 & 8 \end{bmatrix}$$

Gambar 2.3 Citra Dengan Tepi Curam

2. Tepi landai, karena di mana tepi lebar dengan sudut arah kecil.

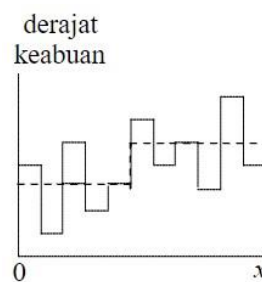


Gambar 2.4 Tepi Landai

$$\begin{bmatrix} 4 & 4 & 5 & 6 & 7 & 8 & 8 & 8 \\ 4 & 4 & 5 & 6 & 7 & 8 & 8 & 8 \\ 4 & 4 & 5 & 6 & 7 & 8 & 8 & 8 \\ 4 & 4 & 5 & 6 & 7 & 8 & 8 & 8 \\ 4 & 4 & 5 & 6 & 7 & 8 & 8 & 8 \end{bmatrix}$$

Gambar 2.5 Citra Dengan Tepi Landai

3. Tepi yang memiliki *noise*, biasanya dilakukan oleh operator *image enhancement* yang berfungsi untuk menghaluskan citra tersebut.



Gambar 2.6 Tepi Yang Memiliki *Noise*

2.1.4 FIRE (*Fast Iris Recognition*)

FIRE adalah singkatan dari algoritma *Fast Iris Recognition* yang dirancang khusus untuk pengenalan iris pada *smartphone* dengan syarat harus terdapat cahaya pada irisnya. Ini didasarkan pada kombinasi tiga pengklasifikasi dalam mengeksplorasi informasi warna dan informasi pada tekstur iris. Dengan waktu komputasi yang terbatas membuat FIRE sangat cocok untuk verifikasi pengguna yang cepat dan tepat pada perangkat seluler (Galdi, 2017).

Algoritma ini didasarkan pada kombinasi tiga ekstraktor fitur, masing-masing menggambarkan karakteristik iris yang berbeda, yaitu deskriptor warna pada iris, mendeskripsi tekstur iris, dan mendeskripsikan titik warna iris (selanjutnya disebut "*cluster*"). Fitur utama dari metode yang diusulkan adalah:

1. Deskriptor warna pada iris.

Informasi warna dalam citra iris sangat diperlukan, karena dapat dimanfaatkan untuk meningkatkan kinerja algoritma untuk mengenali jarak pada setiap warna tiap pikselnya. Dalam hal tersebut, jarak yang di uji yaitu jarak euclidean, jarak chi-kuadrat, jarak kecocokan, jarak Kolmogorov-Smirnov dan perpotongan histogram.

2. Mendeskripsi tekstur iris.

Deskripsi tekstur didasarkan pada perhitungan dimensi Minkowski–Bouligand, dan juga dikenal sebagai dimensi penghitungan kotak (*square*). Citra iris pertama kali diproyeksikan di CIELAB ruang warna pada tahun 1976, di mana L^* adalah menyatakan tingkat nilai kecerahan dan A^* dan B^* menyatakan untuk 4 warna unik penglihatan manusia yakni, merah, hijau, biru dan kuning.

Hanya dimensi warna yang memiliki informasi kecerahan lebih atau berbeda diantara citra iris yang berbeda dapat diproses lebih lanjut, misalnya kondisi pencahayaan citra berubah yang diperoleh dari pengambilan citra dari pengguna yang sama.

Untuk deskriptor tekstur, peneliti menguji ruang dimensi warna yang berbeda dan jumlah saluran warna yang berbeda untuk diproses dengan teknik penghitungan kotak pada setiap piksel. Dalam versi algoritma sebelumnya, peneliti memilih saluran a^* dan b^* dari $L^*A^*B^*$ ruang dimensi warna. Di mungkin untuk mengamati konfigurasi berbeda yang diperoleh dengan menggabungkan $L^*A^*B^*$ dan saluran RGB.

3. Mendeskripsikan titik warna pada iris.

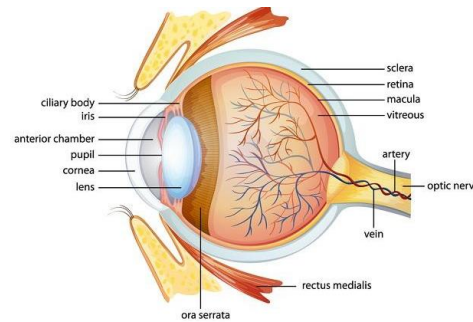
Penggunaan istilah "*cluster*" berfungsi untuk menunjukkan bintik-bintik warna kecil yang menjadi ciri beberapa iris manusia. Ketika dua iris serupa dalam hal warna (atau tampak serupa karena gambar dalam skala abu-abu), manusia memanfaatkan titik-titik kecil ini untuk menentukan apakah gambar yang diamati termasuk dalam iris yang sama atau tidak.

Pada versi sebelumnya, FIRE memiliki tiga metrik *cluster* yang telah digabungkan untuk mendapatkan jarak *cluster*, yaitu: jarak horizontal *centroid*, perbedaan eksentrisitas, dan perbedaan orientasi. Di sini peneliti menguji tiga metrik tersebut secara tunggal dan juga dua metrik tambahan, yaitu jarak vertikal *centroid* dan jarak euclidean *centroid*.

Hasilnya menunjukkan bahwa jarak *centroid* euclidean melakukan kinerja yang terbaik diikuti oleh jarak *centroid* vertikal dan horizontal. Kinerja perbedaan eksentrisitas dan orientasi, sangat buruk. Untuk alasan ini, peneliti memilih jarak *centroid* euclidean.

2.1.5 Iris Mata

Iris atau selaput pelangi pada mata dapat dijadikan sebagai basis sistem biometrik. Dimana setiap iris memiliki tekstur yang amat rinci dan unik untuk setiap orang serta tetap stabil dalam waktu bertahun-tahun (Masek, 2003). Sesuai Gambar 2.7, letak iris terlindung di belakang kornea sehingga membuat iris terlindung dari berbagai kerusakan atau perubahan dari luar. Bagian mata ini tidak dapat diubah melalui pembedahan tanpa menimbulkan kerusakan pada penglihatan.



Gambar 2.7 Anatomi Mata Manusia

Berbagai keuntungan dari pemakaian iris untuk sistem identifikasi adalah:

1. Iris pada mata berada dibagian yang terlindung dari lingkungan, jadi keberadaannya akan sangat aman.
2. Iris mata tidak mungkin berubah, bahkan dengan melakukan operasi, iris akan tetap terjaga. Namun jikalau operasi tersebut membuat luka pada akan menyebabkan cacat pada mata.
3. Iris memiliki reaksi terhadap cahaya, sehingga memungkinkan dilakukan pengujian secara alami terhadap kemungkinan adanya penipuan serta penggunaan lensa mata palsu.

2.1.6 Segmentasi Iris

Segmentasi iris atau *Iris Segmentation* merupakan proses yang mengacu pada penentuan lingkaran iris bagian luar beserta batas lingkaran iris yang biasa tertutupi oleh kelopak mata bahkan tertutupi oleh bulu mata (Paulín-Martínez et al., 2019). Peran Segmentasi Iris pada penelitian sangat penting, karena jika proses segmentasi gagal mengakibatkan berkurangnya efektifitas atau iris tidak dapat terbaca. Dua jenis metode segmentasi yang digunakan, yaitu *Transformasi Hough* dan *Integro-Differential*.

2.1.7 Operator *Integro-Differential*

Operator *Integro-differential* adalah sebuah operator untuk mencari iris mata oleh John Daugman. Operator ini dapat berjalan karena berdasar atas perbedaan intensitas cahaya pada citra mata (Putra, 2010). (Daugman, 2004) menyarankan *Integro-Differential Operator* (IDO) dan *Hough Transform* digunakan untuk mendeteksi kontur atau garis tepi pupil dan iris, dengan mengasumsikan garis tepi pupil dan iris berupa lingkaran dan dapat dicari dengan melalui proses citra iris yang telah dihaluskan (*smoothing*) menggunakan Gaussian Filter karena dapat mengurangi jumlah *noise* pada suatu citra iris tersebut. Parameter yang digunakan adalah nilai maksimum integral derivative dari titik pusat dan jari-jari dari garis tepi lingkaran iris.

John Daugman menggunakan operator *integro-differential* tidak hanya untuk menemukan langar iris dan daerah pupil, namun juga mencari lengkungan kelopak mata atas dan bawah. Operator *integro-differential* didefinisikan sebagai berikut:

$$\max(r, x_0, y_0) \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{(r, x_0, y_0)} \frac{I(x, y)}{2\pi r} ds \right| \quad (2.1)$$

Keterangan:

Tabel 2.1 Keterangan Rumus *Integro-Differential*

No.	Simbol	Keterangan
1.	$I(x, y)$	Intensitas Pixel Iris Pada Citra Mata
2.	r	Jari-Jari Lingkaran Iris
3.	ds	Batas Kontur Lingkaran dari r, x_0, y_0
4.	(x_0, y_0)	Titik Koordinat Pusat

5.	$G_{\sigma}(r)$	Fungsi Gaussian <i>Smoothing</i> dengan skala σ
----	-----------------	---

Operator *integro-differential* mencari jalur melingkar di mana ada perubahan maksimum dalam nilai piksel, dengan memvariasikan jari-jari dan pusat x dan y dari kontur melingkar. Operator diterapkan secara iteratif dengan jumlah penghalusan yang semakin berkurang untuk mencapai lokalisasi yang tepat. Kelopak mata dilokalisasi dengan cara yang sama, dengan jalur integrasi kontur berubah dari melingkar menjadi busur.

$G_{\sigma}(r)$ adalah fungsi *Gaussian Smoothing* dengan nilai skala spasial, secara matematis dijelaskan dengan ekspresi berikut:

$$G_{\sigma}(r) = \left(\frac{1}{2\pi r} \right) e^{-\frac{(r-r_0)^2}{2\sigma^2}} \quad (2.2)$$

Citra iris mata umumnya memiliki pantulan atau refleksi spekular yang muncul sebagai titik terang dan dikelilingi oleh nilai gradien yang besar. Area refleksi yang kuat dapat sangat mudah untuk dideteksi dengan menggunakan pendekatan *threshold* sederhana. Pada proses mencari iris, kecerahan citra dihitung dan nilai ambang batas (*threshold*) ditentukan berdasarkan kecerahan citra.

Citra iris $I(x, y)$, dimana (x, y) menunjukkan lokasi piksel area dengan intensitas tertinggi dapat dideteksi dengan ekspresi berikut:

$$I(x, y) > \beta_s \quad (2.3)$$

$I(x, y)$ menyatakan intensitas suatu piksel citra iris dan β_s nilai intensitas citra iris yaitu:

$$\beta_s = \frac{\sum_{i=1}^N I(x, y)}{N} \quad (2.4)$$

N menyatakan jumlah total piksel pada citra iris yang dihitung dengan menggunakan operasi perkalian ukuran baris dengan ukuran kolom. Citra iris selanjutnya akan difilter menggunakan filter *multi-scale edge-weighted highpass* adaptif untuk mendeteksi tepi (*edges*) yang bervariasi dimana tepi tersebut merepresentasikan batas refleksi. Filter *highpass* dapat direpresentasikan sebagai berikut:

$$f(x, y) = |\mathcal{E}f(x, y)|^p \cdot Hf(x, y) \quad (2.5)$$

Dimana $|\mathcal{E}f|^p$ adalah citra yang dibentuk dengan menaikkan nilai dari setiap titik pada $Lf(x, y)$ pada citra $\mathcal{E}f$ ke pangkat p . Citra $|\mathcal{E}f|^p$ memiliki nilai bobot filter *highpass* Hf *pointwise* sesuai dengan tepi (*edges*) pada \mathcal{E} . Rincian lebih lanjut terdapat pada (Nowak & Baraniuk, n.d.). Citra yang telah dihaluskan (*smoothed*) $f(x, y)$ selanjutnya akan dikomposisi dengan menerapkan *discrete wavelet transform* (DWT). Pada skala 2^j , diperoleh tiga detail citra $W_{2^j}^h$, $W_{2^j}^v$ dan $W_{2^j}^d$ serta diperkirakan ukuran sebuah citra tersebut dengan A_{2^j} . Superskrip h , v , dan d masing-masing merepresentasikan koefisien detail horizontal, vertikal dan diagonal. Tepi di area refleksi digambarkan pada skala 2^j dengan menghitung nilai maksimum dari koefisien sebagai berikut:

$$D_{2^j} f(x, y) = \sqrt{|W_{2^j}^h f(x, y)|^2 + |W_{2^j}^v f(x, y)|^2 + |W_{2^j}^d f(x, y)|^2} \quad (2.6)$$

Citra yang terfilter *edge-weighted highpass* dihitung dengan memanipulasi titik dari citra terfilter Hf dengan $|D_{2^j} f|^p$ yang menghasilkan filter *highpass* $P + 1^{\text{st}}$ -order dan diterapkan terhadap tepi pada skala 2^j . Analisis multiskala menghasilkan satu set filter *highpass* yang memiliki nilai tepi sehingga dapat

diimplementasikan ke tepi (*edges*) pada skala yang akan ditentukan. Oleh karena itu, peta refleksi $R(x, y)$ sebagai berikut:

$$R(x, y) = \sum_{p=1}^P \sum_{j=1}^J |D_{2^j} f|^p Hf \quad (2.7)$$

Dari persamaan (2.7) dapat merepresentasikan batas-batas titik refleksi.

Setelah pendeteksian refleksi dilakukan modifikasi adaptif pada citra iris secara *in-painting* untuk mengisi titik refleksi. Diberikan citra iris $I(x, y)$ dengan peta area titik refleksi yang dilambangkan dengan Ω sedangkan batas titik refleksi atau kontur dilambangkan $\partial\Omega$. Sebutan “lubang” atau “wilayah isi” untuk merujuk kepada titik refleksi dan “wilayah target” merujuk pada sisa citra utuh. Dalam menghitung kesamaan antara wilayah isi dan wilayah target, maka potongan citra harus dipilih pada satu waktu yang bersama. Potongan tersebut dilambangkan ψ_i yang merupakan sub wilayah berbentuk persegi dari citra yang berpusat pada piksel i . Pada umumnya, ukuran ψ_i memiliki dimensi sedikit lebih besar dari elemen tekstur terbesar yang terdeteksi.

Ukuran potongan yang digunakan berukuran 3×3 karena lebih efektif diterapkan pada citra memiliki tekstur tinggi. Potongan pengisi dilambangkan ψ_p memiliki informasi dari bagian wilayah pengisi (lubang) dan bagian dari wilayah target. Setiap potongan wilayah pengisi ψ_{pi} dipilih dengan melakukan pencarian dari wilayah target. Ketika N -potongan yang paling mirip ditemukan, maka potongan ψ_{pi} digabungkan secara linier dengan potongan ψ_{qi} untuk memprediksi wilayah ψ_p yang tidak diketahui atau terdeteksi. Untuk membatasi area pencarian pada batas refleksi, maka titik prioritas diberikan pada potongan tersebut dimana didasarkan oleh jarak. Priotitas untuk urutan pengisian wilayah iris $P(p)$

didasarkan atas aturan konfiden (*confidence*) $C(p)$ dan aturan data $D(p)$ sehingga didapatkan persamaan (2.8):

$$P(p) = C(p) * D(p) \quad (2.8)$$

Dimana

$$C(p) = \frac{\sum_{q \in \psi_p \cap \bar{\Omega}} C(q)}{|\psi_p|}, \quad D(p) = \frac{|\nabla I_p^\perp \cdot n_p|}{\alpha} \quad (2.9)$$

$|\psi_p|$ merupakan daerah atau area dari ψ_p , dan α adalah faktor normalisasi yang mana $\alpha = 255$ untuk citra iris *grayscale*, n_p merupakan vektor satuan ortogonal terhadap aturan $\partial\Omega$ dari titik p .

$$\hat{P}(p) = C(p) * \exp\left(\frac{D(p)}{2\sigma^2}\right) \quad (2.10)$$

Eksponensial kuadrat dari aturan data $D(p)$ digunakan untuk meningkatkan pengaruh data pada persamaan (2.10) sehingga meningkatkan nilai kesamaan atau kecocokan antara ψ_p dan ψ_{qi} untuk lebih mendekati pada batas. Pemilihan potongan ψ_{qi} dari potongan K-potongan yang serupa didapatkan dengan cara meminimalkan jumlah kuadrat jarak SSD yang mana terdapat pada persamaan (2.11).

$$\psi_{qi} = \arg \min_{\psi_{q \in \bar{\Omega}}} d_{SSD}(P\psi_p, P\psi_q) \quad (2.11)$$

Dimana d_{SSD} merupakan jarak SSD, sedangkan $\{P\}$ mengekstrak area refleksi yang telah diketahui oleh ψ_p (piksel yang telah diketahui dan terisi pada bagian potongan refleksi).

Setelah menghilangkan refleksi, hal yang dilakukan untuk membuat *template* iris agar bersih dari *noise*, maka dilakukan pembersihan iris dari bulu mata. Bulu mata dapat merusak keutuhan iris pada citra iris, mengakibatkan dapat

menghambat akurasi pengenalan jika tidak terdeteksi, sehingga diperlukan untuk menghilangkan bagian iris yang tertumpuk oleh bulu mata. Pada umumnya bulumata yang terdapat pada citra iris memiliki intensitas tinggi berwarna gelap dibandingkan intensitas yang dimiliki oleh iris.

Misalkan $I(N, M)$ merupakan citra iris asli yang dibagi menjadi blok-blok dimana setiap blok merupakan sub-citra dari citra iris aslinya. Untuk setiap blok $n \times m$ akan dihitung menggunakan *Discrete Wavelet Transform* (DWT), yang menggunakan teknik *multiscale frequency decomposition*. Frekuensi tinggi *sub-bands* dari koefisien wavelet dilambangkan dengan $D_i = \{HL, LH, HH\}$ yang mampu menangkap variasi frekuensi tinggi dalam arah horizontal, vertikal dan diagonal. Wavelet Haar dengan satu tingkat resolusi dipilih untuk menguraikan atau mendekomposisi setiap blok menjadi empat frekuensi *bands* dengan ukuran $\frac{n_1}{2} \times \frac{m_1}{2}$. Akar kuadrat orde kedua dari koefisien wavelet di setiap koefisien D_i dihitung seperti yang didefinisikan oleh (Pundlik et al., 2008).

$$f(k) = \left(\frac{1}{2} \sum_{i=0}^{\frac{x_l}{2}} \sum_{j=0}^{\frac{y_l}{2}} W_{k,i,j}^2 \right)^{\frac{1}{2}} \quad (2.12)$$

Dimana $k \in D_i$ dan W_k merepresentasikan matriks koefisien wavelet dari blok dalam himpunan D_i . Deskriptor tekstur yang merepresentasikan *maximum energy* balok (energi lokal) ditentukan sebagai berikut:

$$E_n = \max f(k), \quad \forall k \in D_i \quad (2.13)$$

$f(k)$ adalah energi dengan frekuensi tertinggi dari *sub-bands*. Sebuah vektor energi 1-D digunakan dalam proses pengelompokan *block-wise* untuk mencari wilayah yang sesuai atau *Region of Interest* (ROI). Klasifikasi blok

menjadi daerah kandidat bulu mata atau *candidate eyelash region* (CES) dan daerah non-bulu mata atau *non-eyelash* (NES) dicapai dengan menggunakan pendekatan klasifikasi Bayesian. Pengklasifikasi Bayes adalah pengklasifikasi probabilistik sederhana berdasarkan penerapan teorema Bayes (dari statistik Bayesian). Tiga blok didefinisikan sebagai berikut yaitu blok B_H frekuensi tinggi, blok B_L frekuensi rendah dan blok B_M frekuensi menengah. Blok frekuensi tinggi mencakup wilayah dalam citra dengan frekuensi tinggi (tepi tajam) seperti blok bulu mata, dan blok tepi pupil. Blok frekuensi rendah (tanpa tepi sama sekali atau tepi yang sangat halus) mencakup blok seperti blok tepi limbik dan area halus lainnya pada citra iris, sedangkan blok frekuensi menengah mencakup blok dengan tepi lemah seperti blok bulu mata yang dapat dipisahkan.

Block of Interest (BOI) merupakan jumlah dari blok-blok frekuensi menengah B_M dan frekuensi tinggi B_H . Semua blok akan dicari nilainya dengan *k-means classifier* untuk mengelompokkan E_n menjadi tiga kelas, selanjutnya klasifikasi akhir menggunakan *bayes classifier* sebagai berikut:

$$P(x_n|E) = \frac{P(E^k|x_n)P(x_n)}{P(E)} \quad (2.14)$$

$P(E^k|x_n)$ menunjukkan probabilitas distribusi dari komponen fitur ekstraksi E^k yang dihasilkan dari pengelompokan x_n . Misalkan $E_i^{\vec{d}}(x, y)$ adalah istilah *energy regional* untuk area tertentu pada iris dan $D_i^{\vec{d}}$ merupakan nilai intensitas tingkat keabuan dari piksel (x, y) . Dimana i merupakan tingkat resolusi, \vec{d} merupakan simbol arah atau *direction* yang merepresentasikan horizontal, vertikal, dan arah diagonal dari koefisien detail transformasi wavelet dan k

merepresentasikan area (*region*). *Energy regional* dapat dihitung dengan persamaan (2.15) berikut:

$$E_i^{\vec{d}}(x, y) = \sum_{n' \in L, m' \in K} [D_i^{\vec{d}}(n + n', m + m')]^2 \quad (2.15)$$

Dengan $\vec{d} = 1, 2, 3$. Selisih dari rasio calon bulu mata dan tidak calon bulu mata yang terdeteksi akan melalui evaluasi dengan persamaan (2.16) berikut:

$$RDR = \frac{(\mu E_{i,k1}^{\vec{d}} - \mu E_{i,k2}^{\vec{d}})}{\mu E_{i,k1}^{\vec{d}}} \quad (2.16)$$

Standar deviasi σ_i dari area calon tidak bulu mata akan tetap dihitung.

Selanjutnya setiap piksel pada iris jika memiliki rata-rata $\mu E_i^{\vec{d}}$ (energi dalam arah horizontal, vertikal, dan diagonal) lebih besar dari salah satu dari delapan nilai dari piksel tetangga (*neighbours*) dan lebih kecil dari T , maka piksel tersebut dianggap sebagai piksel bulu mata. T merupakan ambang batas (*threshold*) yang ditentukan sebagai berikut:

$$T = \mu E_{i,k1}^{\vec{d}} - 2.5\sigma_i \quad (2.17)$$

2.1.8 Transformasi Hough

Algoritma *Hough Transform* digunakan untuk mendeteksi objek geometris sederhana baik berupa garis, lingkaran, serta elips yang ada pada citra digital mata. Pada penelitian ini, transformasi hough dengan model *graph cut based active contour* (GBAC) yang membantu dalam proses ekstraksi citra karena dapat mendeteksi lingkaran pusat pupil dan lingkaran iris dengan baik. Lingkaran iris terdapat dua batas dimana batas dalam disebut batas pupil dan batas luar disebut batas iris atau skelera. GBAC mengadopsi pendekatan hierarki dengan 2 tingkat dan 2 langkah, dimana akan melakukan segmentasi pupil terlebih dahulu sebelum

melakukan segmentasi pada batas limbik (*limbic boundaries*) iris mata (Ukpai, n.d. 2015). Dengan nilai keuntungan dapat mendapatkan hasil lebih cepat dengan proses pencocokan dengan pola serta mampu memberikan toleransi terhadap *noise* yang dimiliki oleh citra (Putra, 2010). Kemudian, lingkaran transformasi Hough digunakan untuk mencari dan mendeteksi lingkaran dalam citra iris yang telah diproses sebelumnya. Lingkaran yang terdeteksi disebut calon pupil C_p . Skema evaluasi dikembangkan untuk menemukan calon pupil C_{pi} yang mana sesuai dengan pupil sebenarnya. Segmentasi akhir dari batas pupil selanjutnya didapatkan dengan menyempurnakan kontur awal pupil yang dihasilkan menggunakan model kontur aktif berbasis potongan grafik (GCBAC).

Citra iris sebelum diproses harus melewati proses modifikasi menjadi citra yang berukuran $(n \times n)$ yang berfungsi untuk mengurangi *noise*. Lingkaran transformasi Hough kemudian diterapkan untuk mencari calon pupil C_p pada citra iris. Jarak Euclidian (*Euclidean Distance*) dari semua C_p menuju titik pusat citra iris kemudian dihitung. C_p yang memiliki jarak Euclidean terendah atau terdekat ke pusat citra iris mata kemudian dipilih sebagai pupil.

Mengukur entropi calon pupil C_p dapat membantu untuk membedakan perbedaan calon pupil asli dari bentuk bentuk calon pupil palsu. Nilai intensitas yang disesuaikan I^{C_p} dari calon pupil C_{pi} dihitung dengan mengonversi nilai intensitas pada citra dengan matriks dengan ukuran 5×5 yang digunakan untuk menghaluskan citra dan mengurangi *noise*. Hough *transform* digunakan untuk mendeteksi calon pupil di dalam citra. Calon pupil asli dipilih berdasarkan jarak dari pusat citra serta variasi intensitas (entropi) dalam lingkaran calon pupil. Hal ini dapat dinyatakan secara matematis sebagai berikut:

$$I^{c_p}(x, y) = \frac{I'_{x,y}}{1 + 3d_{(x,y)}^2} + H \quad (2.18)$$

Dimana d^2 merupakan nilai jarak Euclidean antara calon pupil dan titik pusat citra iris. H merupakan ukuran dispersi didalam calon pupil C_p yang dihitung menggunakan entropi Shannon yang didefinisikan sebagai berikut:

$$H = \sum_{i=1}^m P_i \log \frac{1}{P_i} = - \sum_{i=1}^m P_i \log P_i \quad (2.19)$$

I' adalah nilai *Intensity Adjustment* (intensitas yang disesuaikan), dimana n merupakan nilai yang dipilih berdasarkan 5% dari resolusi citra. $I'_{x,y}$ adalah intensitas citra dalam calon pupil pada koordinat (x, y) . Calon pupil dengan intensitas penyesuaian (*Intensity Adjustment*) tertinggi dipilih sebagai calon pupil asli.

$$I' = (255 - 1) * 1_{n \times n} \quad (2.20)$$

Transformasi Hough menggunakan parameter lingkaran pusat koordinat x_0^2 dan y_0^2 serta radius r_0^2 akan dihitung untuk evaluasi pada ruang Hough. Berikut persamaan (2.21) lingkaran tersebut.

$$x_0^2 + y_0^2 - r_0^2 = 0 \quad (2.21)$$

2.1.9 Fitur *Encoding*

Pada tahap fitur *encoding*, dalam proses pengenalan iris akan dilakukan ekstraksi fitur iris dan pembuatan *template* untuk perbandingan antara citra iris yang berbeda. Untuk mencapai hasil identifikasi terbaik, algoritma *encoding* harus mengekstrak informasi yang paling diskriminatif dari iris (Ukpai, n.d. 2015). Prosedur pengkodean ini, yang melibatkan demodulasi dan kuantisasi, didefinisikan oleh Daugman sebagai berikut:

$$\begin{aligned}
& h_{\{R_e, I_m\}} \tag{2.22} \\
& = \text{sgn}_{\{R_e, I_m\}} \int_{\rho} \int_{\phi} I(\rho, \phi) e^{-i\omega(\theta_0 - \phi)} e^{-(r_0 - \rho)^2 / \alpha^2} e^{-(\theta_0 - \phi)^2 / \beta^2} \rho d\rho d\phi
\end{aligned}$$

Daugman menggunakan filter gabor 2D untuk mengekstraksi informasi *local* frekuensi. Koefisien bilangan kompleks yang merepresentasikan *phasor* (fasor) dalam bidang kompleks dihitung, kemudian sesuai dengan lokasi, sudut vektor akan dikuantisasi ke salah satu dari empat segmen lingkaran yang memberikan dua bit ke dalam *template* iris. Dimana $I(\rho, \phi)$ merupakan citra iris dalam koordinat polar (kutub), (r_0, θ_0) merupakan koordinat titik dimana *encoding* diterapkan, α dan β adalah parameter ukuran dari multiskala wavelet 2D, serta ω merupakan frekuensi wavelet. Nilai yang dihasilkan $h_{\{R_e, I_m\}}$ sesuai dengan bagian nyata dan imajiner terkuantisasi dari sebuah fasor. Setelah diaplikasikan kesetiap titik piksel iris yang dinormalisasi, maka diperoleh 2048-bit informasi.

Filter Gabor digunakan pada tahap ekstraksi ciri yang bertujuan untuk mendapatkan berbagai informasi dari tekstur yang dimiliki oleh suatu citra digital. Dengan tujuan mengekstraksi suatu fitur atau ciri, filter gabor memiliki keterbatasan. Sehingga banyak disarankan oleh (Field, 1987), filter log-gabor dapat mengekstraksi suatu citra lebih baik dilihat dari skala linier karena menggunakan fungsi gaussian yang dapat dilihat pada skala logaritmik. Log-gabor dapat dikatakan pengembangan dari filter gabor. Karena, statistik yang didapatkan setelah mengekstraksi fitur pada citra digital, menunjukkan filter log-gabor dapat mendapatkan informasi dengan komponen frekuensi tinggi. Dikarenakan filter gabor kurang untuk mendapatkan informasi dengan komponen

frekuensi tinggi, maka filter log-gabor adalah pilihan yang lebih baik (Neha Kak, 2010).

Pada penelitian ini, filter log-gabor dapat mengekstrak fitur dari iris dan berfungsi untuk membuat model biometrik pada citra tersebut. Berikut merupakan rumus yang digunakan:

Rumus 1 dimensi:

$$G(f) = \exp\left(\frac{-(\log(f/f_0))^2}{2(\log(\sigma/f_0))^2}\right) \quad (2.23)$$

Keterangan:

f_0 = Pusat Frekuensi Filter Gabor

σ = Lebar Filter Gabor

Rumus 2 dimensi:

$$G(f, \theta) = \exp\left(\frac{-(\log(f/f_0))^2}{2(\log(\sigma/f_0))^2}\right) \times \exp\left(\frac{-(\theta - \theta_0)}{2\sigma_\theta^2}\right) \quad (2.24)$$

Keterangan:

f_0 = Pusat Frekuensi Filter Log-Gabor

σ = Lebar Filter Gabor

f = Frekuensi

θ = Sudut Orientasi

σ/f_0 = Lebar Filter Terhadap Pusat Frekuensi

θ_0 = Sudut Awal

σ_θ = Lebar Sudut Filter Log-Gabor

2.1.10 Algoritma *Hamming Distance*

Algoritma *hamming distance* merupakan sebuah algoritma yang dapat menghitung jumlah perbedaan dua buah bilangan biner yang mempunyai panjang yang sama (Murti, 2005). Pada penelitian ini *Hamming distance* dapat memberikan informasi berapa banyak bit yang sama antara dua pola bit. Sehingga, dari perbedaan antara kedua pola tersebut dapat memberikan informasi bahwa iris pada berbeda memiliki perbedaan atau tidak. *Hamming distance* menggunakan operator *Boolean Exclusive – OR (XOR)* yang digunakan untuk membandingkan dua pola iris yang berbeda. Operator *XOR* tersebut, mendeteksi perbedaan antara setiap pasang bit. Rumus yang digunakan sebagai berikut:

$$HD = \frac{1}{N} \sum_{j=1}^N X_j(XOR)Y_j \quad (2.25)$$

Keterangan:

Tabel 2.2 Keterangan *Hamming Distance*

<i>HD</i>	Jumlah bit yang berbeda dari dua <i>string</i>
<i>N</i>	Jumlah biner yang ditampilkan pada <i>string</i>
X_j, Y_j	Bentuk representasi untuk dibandingkan
<i>XOR</i>	\oplus (<i>Exclusive-OR</i>)

Secara teori, dua model iris yang dihasilkan dari iris yang sama akan memiliki jarak hamming 0. Namun, dalam penerapannya tidak selamanya terjadi. Beberapa faktor yang menyebabkan gagalnya algoritma jarak hamming tidak

mendapatkan hasil yang maksimal yakni, Normalisasi yang tidak sempurna, *Noise* pada citra yang tidak terdeteksi.

2.1.11 *Weight Euclidean Distance*

Weight Euclidean Distance adalah sebuah metode yang digunakan untuk memperhitungkan jarak dari dua buah titik yang terdapat dalam *Euclidean Space* (Viriyavisuthisakul, 2015). Metode ini mempelajari terkait hubungan antara sudut dan jarak, serta biasanya diterapkan pada dua dimensi dan tiga dimensi. *Euclidean Distance* merupakan jarak yang paling umum digunakan untuk data numerik, untuk dua titik data x dan data y dalam ruang d -dimensi (Gan, Ma, & Wu, 2007).

Dalam bentuk umum *Euclidean Distance* (d) diperoleh sebagai berikut:

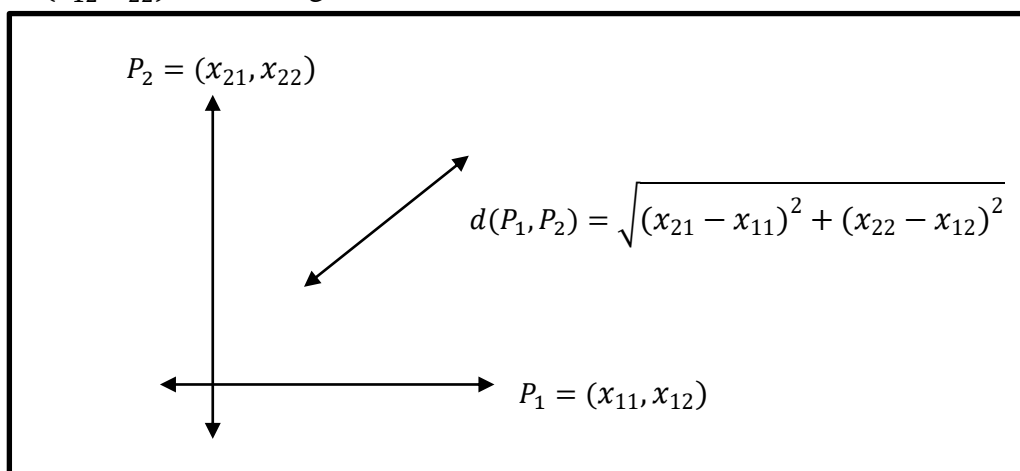
$$x = (x_1, x_2, \dots, x_n)$$

$$y = (y_1, y_2, \dots, y_n)$$

$$d(x, y) = \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + \dots + (y_n - x_n)^2} \quad (2.26)$$

Misalkan pada citra dua dimensi memiliki dua titik koordinat (x_{11}, x_{21})

dan (x_{12}, x_{22}) sesuai dengan Gambar 2.8.



Gambar 2.8 *Euclidean Distance*

Maka *Euclidean Distance* (d) di antara kedua titik tersebut adalah:

$$d(P_1, P_2) = \sqrt{(x_{21} - x_{11})^2 + (x_{22} - x_{12})^2} \quad (2.27)$$

Berikut bentuk lain dari persamaan (2.7):

$$d_{ij} = \sqrt{\sum_{k=1}^m (x_{ik} - x_{jk})^2} \quad (2.28)$$

Keterangan:

Tabel 2.3 Keterangan Rumus *Euclidean Distance*

d_{ij}	Jarak <i>Euclidean</i> entri ke- i dan entri ke- j matriks citra mata
m	Banyaknya parameter atau peubah yang digunakan
x_{ik}	Entri ke- i matriks citra pada parameter ke- k
x_{jk}	Entri ke- j matriks citra pada parameter ke- k

Dengan proses yang menyerupai *Hamming Distance*, *Weighted Euclidian Distance* (WED) membantu menyempurnakan hasil perbandingan iris. Menurut (Ukpai, n.d. 2015) persamaan (2.29) digunakan dalam membandingkan *template* iris yaitu:

$$WED(k) = \sum_{i=1}^M \frac{(f_i - f_i^{(k)})^2}{(\vartheta_i^{(k)})^2} \quad (2.29)$$

Di mana k adalah iris yang memiliki beberapa *template* berukuran M dalam database, f_i merupakan bit ke- i pada *template* yang tidak teridentifikasi, $f_i^{(k)}$ dan $\vartheta_i^{(k)}$ adalah bit ke- i dalam *template database* dan standar deviasi untuk iris k .

2.1.12 Normalisasi *Template Iris*

Normalisasi diperlukan untuk mendapatkan dua set fitur yang konsisten dari citra mata dua dimensi yang berbeda dari orang yang sama. Daugman

menggunakan *Rubber Sheet* untuk menormalisasi citra iris. Dari proses segmentasi area iris yang terdeteksi tidak memiliki bentuk lingkaran atau *elips* yang sempurna, serta memiliki bentuk yang berbeda-beda pada setiap citra, maka digunakan strategi mencari lingkaran yang tepat oleh (Park & Savvides, 2015) untuk mengkompensasi bentuk serta ukuran dari iris.

Dalam proses mengubah area iris yang terdeteksi menjadi bentuk persegi Panjang, menentukan jari-jari dan koordinat titik pusat iris yang sesuai sangat diperlukan. Dengan menentukan jari-jari R pada enam titik yang dipilih dari kontur iris yang telah diekstraksi yaitu pada sudut $[-30^\circ, 30^\circ, 0^\circ, 150^\circ, 180^\circ, 210^\circ]$ dengan terhubung sumbu horizontal dan perhitungan jarak rata-rata dari pusat pupil.

Normalisasi menghasilkan array 2-D dengan dimensi horizontal yang sesuai dengan resolusi sudut, dan dimensi vertikal yang sesuai dengan resolusi radial. Mengingat $I(x, y)$ sebagai citra terlokalisasi, representasi kutub dari bentuk $I(r, \theta)$ dapat diperoleh sebagai berikut:

$$r = \sqrt{(x - x_i)^2 + (y - y_i)^2}, \quad 0 \leq r \leq r_{max} \quad (2.30)$$

$$\theta = \tan^{-1} \left(\frac{y - y_i}{x - x_i} \right) \quad (2.31)$$

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \quad (2.32)$$

$$x(r, \theta) = (1 - r)x_p(\theta) + rx_i(\theta) \quad (2.33)$$

$$y(r, \theta) = (1 - r)y_p(\theta) + ry_i(\theta) \quad (2.34)$$

Dimana r dan θ didefinisikan koordinat pusat (x_i, y_i) . Nilai koordinat tengah yang diperoleh selama pemasangan lingkaran digunakan sebagai titik pusat untuk proses *unwrapping* citra iris.

2.1.13 Cross-Validation

Cross-Validation merupakan salah satu metode statistik yang dapat digunakan untuk proses evaluasi dan proses membandingkan algoritma menjadi dua bagian, yaitu bagian proses melatih model (*training*) dan bagian validasi model (*testing*) (Crammer, 2002), di mana kedua bagian tersebut harus bersilangan satu sama lain, dengan tujuan agar tidak terdapat penumpukan atau tumpang tindih pada data testing. Dapat disimpulkan bahwa, *cross-validation* merupakan sebuah teknik yang digunakan untuk memvalidasi model agar mengetahui hasil statistik analisis data. Salah satu teknik *cross-validation* adalah *k-fold cross-validation*, di mana K adalah bagian set data dengan ukuran yang sama (Juniati, 2017).

K-Fold Cross-Validation

Percobaan 1	Test	Train	Train	Train	Train
Percobaan 2	Train	Test	Train	Train	Train
Percobaan 3	Train	Train	Test	Train	Train
Percobaan 4	Train	Train	Train	Test	Train
Percobaan 5	Train	Train	Train	Train	Test

Gambar 2.9 K-Fold Cross-Validation

2.1.14 Autentikasi Smartphone

Autentikasi adalah suatu proses yang menjadi tindakan atau pembuktian (validasi) terhadap identitas pengguna ketika ingin memasuki dan mengakses sistem penting tertentu. Menurut (Shafique, 2017) dalam penelitian tentang teknik

pengamanan modern pada *smartphone*, teknik autentikasi *smartphone* yang paling umum terdapat pada perangkat *smartphone* ialah menggunakan PIN (numerik), *password* (*alphanumeric*), pola, dan sidik jari. Pada proses autentikasi tersebut bisa dijadikan sebagai penanda atau verifikasi tentang apakah seseorang tersebut diperkenankan masuk ke sistem atau tidak. Dengan demikian pengguna *smartphone* dapat melakukan aktivitas penting secara aman dan nyaman.

2.2 Kajian Integrasi Topik Dengan Al-Quran/Hadits

Ajaran agama Islam telah mengajarkan tentang pentingnya mengembangkan ilmu pengetahuan pada setiap individu manusia. Dalam Al-Quran dijelaskan bahwa manusia memang terlahir dalam kondisi yang tidak mengetahui apapun, yaitu terdapat di dalam surat An-Nahl ayat 78 yang berbunyi:

وَاللَّهُ أَخْرَجَكُم مِّن بُطُونِ أُمَّهَاتِكُمْ لَا تَعْلَمُونَ شَيْئًا وَجَعَلَ لَكُمُ السَّمْعَ وَالْأَبْصَارَ وَالْأَفْئِدَةَ ۗ لَعَلَّكُمْ تَشْكُرُونَ

“Dan Allah mengeluarkan kamu dari perut ibumu dalam keadaan tidak mengetahui sesuatu pun, dan Dia memberimu pendengaran, penglihatan, dan hati nurani, agar kamu bersyukur” (QS. An-Nahl:78).

Penjelasan ayat tersebut dalam Kitab Tafsir Al-Misbah (M. Quraish Shihab, 2002) yakni, pada dasarnya manusia terlahir dalam kondisi tidak mengetahui apapun. Sehingga Allah SWT memberikan potensi dasar berupa indra pendengaran, penglihatan, dan hati atau akal untuk memperoleh pengetahuan tersebut. Hal ini menandakan bahwa, pada penelitian kali ini secara tidak langsung akan mengembangkan ilmu pengetahuan semakin luas.

Berkaitan dengan sistem yang dikembangkan pada penelitian ini, sistem biometrik *iris recognition* mampu membantu pemilik *smartphone* pada proses autentikasi dalam hal efisiensi waktu, mengurangi resiko terkena COVID-19, serta

meningkatkan keamanan data. Dalam proses mencari iris pemilik *smartphone*, dilakukan tahapan ekstraksi fitur pada citra mata dan perbandingan atau *comparation* untuk mencari iris yang sama. Melalui kedua tahapan tersebut, sistem akan mengenali iris pemilik *smartphone* dikarenakan memiliki informasi terkait kebenaran pemilik *smartphone*. Terdapat penjelasan tentang perintah untuk menjaga kebenaran pada surat Al-Baqarah ayat 42, yaitu:

وَلَا تَلْبِسُوا الْحَقَّ بِالْبَاطِلِ وَتَكْتُمُوا الْحَقَّ وَأَنْتُمْ تَعْلَمُونَ

“Dan janganlah kamu campuradukkan kebenaran dengan kebatilan dan (janganlah) kamu sembunyikan kebenaran, sedangkan kamu mengetahuinya.” (QS. Al-Baqarah:42).

Sistem biometrik *iris recognition* dapat mendeteksi iris pada pemilik *smartphone*. Sesuai dengan kedua ayat yang tertulis, penelitian ini selain mengembangkan ilmu pengetahuan semakin luas, juga bermanfaat untuk menjaga keamanan data *smartphone* melalui verifikasi iris mata, karena dari proses *iris recognition* didapatkan informasi terkait kebenaran pemilik *smartphone*.

2.3 Kajian Topik Dengan Teori Pendukung

Penelitian ini dilakukan untuk mendapatkan solusi untuk pengguna *smartphone* di era pandemi COVID-19. Sesuai dengan latar belakang sebelumnya, penelitian ini diharapkan dapat membuat solusi agar pengguna *smartphone* tidak perlu membuka masker saat proses autentikasi *smartphone* dengan menggunakan sistem *iris recognition*. Sehingga secara tidak langsung akan berdampak mengurangi resiko penularan COVID-19.

Proses autentikasi *smartphone* menggunakan iris mata dapat dilakukan dengan adanya objek mata yang tertangkap oleh sensor. Objek mata yang disimpan

dalam bentuk citra digital akan diproses untuk mencari bagian mana pada citra iris tersebut yang terdeteksi iris dan pupil. Citra digital didapatkan dari proses digitalisasi dari citra kontinu. Citra digital merupakan gambar dua dimensi yang dapat dilihat serta ditampilkan pada layar monitor *desktop* atau komputer dengan himpunan berhingga nilai digital atau biasa disebut piksel. Piksel adalah singkatan dari *picture element*, di mana tiap piksel memiliki nilai yang dapat menunjukkan suatu intensitas warna.

Dalam proses mencari iris dan pupil pada citra mata yang dimiliki, pertama citra tersebut diproses menggunakan dasar metode pengolahan citra digital. Pengolahan citra digital atau *Digital Image processing* merupakan suatu metode yang dapat digunakan untuk memproses atau memanipulasi citra digital 2 dimensi. Citra yang baik merupakan sebuah citra yang memiliki kualitas tinggi, di mana citra tersebut memiliki sejumlah informasi yang detail dan jelas sesuai dengan gambar asli. Pada umumnya, objektivitas dari pengolahan citra digital adalah mentransformasi atau menganalisis suatu gambar sehingga informasi baru tentang gambar dibuat lebih jelas. Proses digitalisasi sebuah citra merupakan proses mempresentasikan citra yang berawal dari fungsi kontinu menjadi nilai diskrit. Setelah proses digitalisasi, maka citra digital yang telah terbentuk memiliki dimensi ukuran tinggi x lebar. Sehingga dari metode ini didapatkan informasi citra yang berupa matriks berdimensi $M \times N$.

Informasi citra yang telah didapatkan akan diproses untuk mencari bagian pupil dan iris mata. Pada proses ini menggunakan metode FIRE yang merupakan singkatan dari algoritma *Fast Iris Recognition*, algoritma yang dirancang khusus untuk pengenalan iris pada *smartphone* dengan syarat harus terdapat cahaya cukup

pada citra iris yang diproses. Ini didasarkan pada kombinasi tiga pengklasifikasi dalam mengeksploitasi informasi warna dan informasi pada tekstur iris. Dengan waktu komputasi yang terbatas membuat FIRE sangat cocok untuk verifikasi pengguna yang cepat dan tepat pada perangkat seluler. Tiga proses klasifikasi dalam mencari iris dan pupil yang sama pada citra mata terdapat salah satu algoritma yang sangat efisien, yaitu menggunakan *Euclidean Distance*.

Dalam proses mencari kesamaan antara citra iris yang dimiliki akan melewati dua proses, yaitu proses ekstraksi fitur pada citra mata dan proses pencocokan antar iris yang dimiliki. Proses ekstraksi tersebut, peneliti menggunakan Operator *Integro-Differential* dan Transformasi Hough.

Peneliti memutuskan untuk menggabungkan antara proses ekstraksi fitur citra mata menggunakan Operator *Integro-Differential* dan Transformasi Hough lalu masuk kedalam proses pencocokan iris dengan bantuan algoritma *Euclidean Distance* dan dibantu dengan *Hamming Distance* serta Filter Log-Gabor. Sehingga diperoleh iris mata yang cocok yang didapatkan dari dua atau lebih citra. Dengan begitu peneliti mendapatkan hasil akurasi yang tinggi dan efisiensi waktu proses komputasi pada *smartphone* yang semakin meningkat.

BAB III METODE PENELITIAN

3.1 Jenis Penelitian

Jenis penelitian yang digunakan dalam penelitian ini adalah dengan menggunakan metode eksperimen. Dengan menggunakan metode eksperimen, penelitian ini diartikan sebagai pendekatan penelitian secara kuantitatif yang paling penuh, artinya dapat memenuhi semua persyaratan dalam proses menguji hubungan sebab dan akibat. Serta alasan peneliti menggunakan metode eksperimen ini, dikarenakan sejalan dengan rumusan masalah yaitu untuk mengetahui hasil implementasi dari metode FIRE dan *Image Processing* dibantu dengan operator *integro-differential* dalam pengenalan iris mata pada proses autentikasi *smartphone*.

3.2 Data dan Sumber Data

Penelitian ini, menggunakan data yang berupa sekumpulan data citra digital yang berjumlah 756 citra mata dari dataset CASIA V1 yang memiliki ukuran dimensi 320x280 piksel dengan warna *grayscale*, serta dengan format file **.jpg* atau **.jpeg*.

3.3 Teknik Pengumpulan Data

Data yang digunakan pada penelitian ini adalah dataset iris CASIA V1, di mana dataset didapatkan dari *website* resmi penyedia dataset tersebut. Untuk mengunduh dataset CASIA V1 dapat menuju *website* di <http://biometrics.idealtest.org/> namun dengan syarat telah mempunyai akun di

website tersebut. Sehingga dataset CASIA V1 dapat digunakan oleh peneliti, tanpa adanya pelanggaran penggunaan dataset karena bersifat gratis.

3.4 Instrumen Penelitian

Mengacu pada jenis penelitian ini yaitu menggunakan metode eksperimen, maka instrumen penelitian sebagai alat peneliti utama adalah bahasa pemrograman *python*, karena mampu memberikan visualisasi data untuk menyajikan data dalam bentuk gambar (*plot*), grafik, diagram, dan sebagainya. Peneliti menggunakan *library* OpenCV dan IrisSeg (*Iris Segmentation*), dimana kedua *library* utama tersebut dapat digunakan untuk menyelesaikan tugas untuk mencari iris mata pada proses *iris recognition*. Oleh karena itu, peneliti memilih perangkat yang dapat mendukung dalam menyelesaikan penelitian ini. Perangkat tersebut terbagi menjadi dua jenis, yaitu:

1. Perangkat Lunak (*Software*)

Daftar perangkat lunak yang digunakan peneliti antara lain:

- a) Sistem operasi yang digunakan peneliti adalah Microsoft Windows 11 *Home Single Language*.
- b) *Jupyter Notebook* sebagai media menulis dan menjalankan *script*.
- c) *Google Colabs*, memiliki fungsi serupa seperti *Jupyter Notebook*, dimana dapat digunakan sebagai media menulis, menjalankan, dan menyimpan *script* program melalui *Google Drive*.

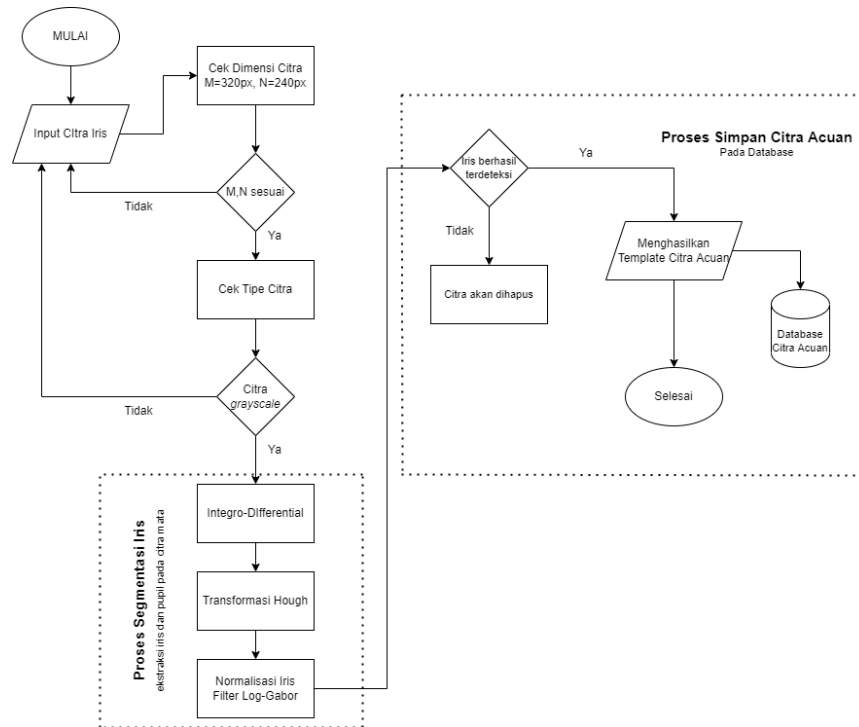
2. Perangkat Keras (*Hardware*)

Beberapa perangkat keras yang dipakai peneliti untuk menjalankan perangkat lunak di atas antara lain:

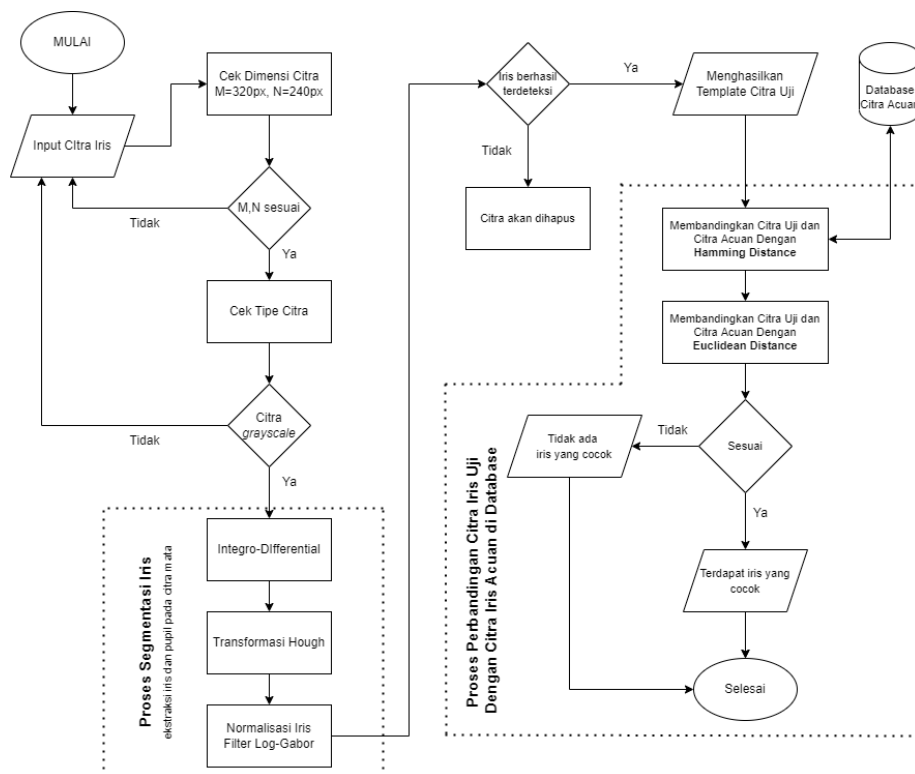
- a) *Google Colabs*, memiliki fungsi serupa seperti *Jupyter Notebook*, dimana dapat digunakan sebagai media menulis, menjalankan, dan menyimpan *script* program melalui *Google Drive*.
- b) RAM (*Random Access Memory*) 8GB + 4GB (*extend*) DDR4.
- c) VGA AMD Radeon Vega 8 Graphics 2GB.
- d) VGA NVIDIA GeForce GTX 1050 4GB.
- e) SSD M.2 NVME 256 GB EVO.
- f) HDD (*Hardisk*) dengan ukuran 1 Terabyte (1000 GB).
- g) Layer monitor 15,6"
- h) Server Google Colabs (2xCPU (Prosesor) Intel Xeon 2,2 GHz, RAM 13GB, HDD 100 GB, GPU (VGA) 64GB).

3.5 Teknik Analisis Data

Dalam melakukan proses mencari iris mata, diperlukan melewati beberapa tahapan terlebih dahulu. Tahapan tersebut tergambar pada diagram alur (*flowchart*) yang dapat dilihat pada Gambar 3.2. Proses diawali dengan tahapan *preprocessing*, selanjutnya citra iris mata yang sesuai dengan ketentuan *preprocessing* masuk ketahapan ekstraksi fitur, lalu dari proses tersebut selanjutnya akan masuk ke tahapan perbandingan atau *comparation* dan tahapan evaluasi. Sehingga, dari beberapa tahapan tersebut peneliti mendapatkan hasil dari prosesnya.



Gambar 3.1 Diagram Alur *Input Citra Acuan Pada Database*



Gambar 3.2 Diagram Alur Proses *Iris Recognition*

3.5.1 Tahap *Preprocessing*

Pada tahapan pengolahan awal atau *preprocessing* penelitian ini akan diawali dengan proses verifikasi citra iris terlebih dahulu. Citra iris yang digunakan didapatkan dari dataset CASIA V1. Dimana, citra dataset tersebut merupakan citra *grayscale* dengan format **jpg* atau **jpeg* serta memiliki dimensi 320 x 280 piksel. Terdapat 756 citra iris dalam dataset tersebut, yang dapat digunakan untuk data latih dan data uji.

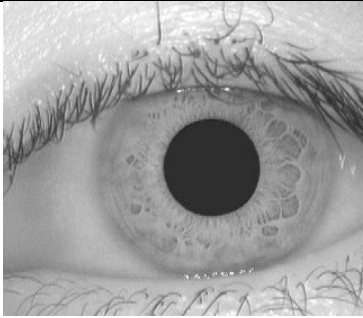
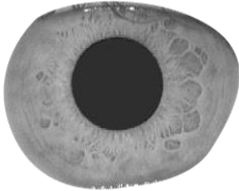
Dalam proses pencarian iris mata agar didapatkan informasi terkait iris dan pupil, maka akan melewati beberapa tahapan terlebih dahulu, yakni:

1. Memastikan dimensi citra iris, sesuai dengan dimensi yang telah ditetapkan yaitu, $M=320$ piksel dan $N=280$ piksel.
2. Memastikan citra iris memiliki warna *grayscale*.

3.5.2 Tahapan Segmentasi Iris Mata

Pada tahapan segmentasi iris mata, terdapat proses ekstraksi fitur iris yang bertujuan untuk mendeteksi iris mata yang terdapat pada citra. Proses mengekstrak pupil dan iris menggunakan operator *integro-differential* (persamaan 2.1) dan transformasi Hough (persamaan 2.21). Citra mata yang telah diproses dan didapatkan bagian iris mata, citra tersebut akan dipotong atau *cropping* untuk diambil bagian lingkaran irisnya saja.

Tabel 3.1 Proses Segmentasi Iris

Citra Mata	
Citra Hasil Proses Segmentasi	

Setelah citra hasil segmentasi telah didapatkan, maka proses normalisasi dilakukan, serta akan dilakukan verifikasi dan klasifikasi citra iris oleh Filter Log-Gabor (persamaan 2.24). Di mana pada tahap ini, citra yang telah dipotong akan diubah menjadi persegi panjang, agar lebih mudah diproses pada tahapan perbandingan.

3.5.3 Tahapan Perbandingan (*Comparison*)

Tahap perbandingan merupakan tahapan lanjutan dari tahapan segmentasi iris mata, di mana iris telah ditemukan dan citra telah diklasifikasikan. Proses perbandingan iris mata bertujuan untuk mencari iris mata yang sama pada setiap citra yang berbeda, dengan menggunakan algoritma *Hamming Distance* (persamaan 2.5), dan *Euclidean Distance* (persamaan 2.8). Peneliti menggunakan kedua algoritma tersebut karena mampu memaksimalkan proses perbandingan iris pada tiap citra yang berbeda. *Hamming Distance*, dapat menghitung jumlah

perbedaan dua buah bilangan biner yang mempunyai panjang yang sama. Sedangkan *Euclidean Distance* merupakan algoritma yang mampu memperhitungkan jarak dari dua buah titik yang terdapat dalam *Euclidean Space*. Sehingga dengan dilakukan komparasi antara kedua algoritma tersebut dapat meningkatkan hasil iris mata yang sama pada citra yang berbeda.

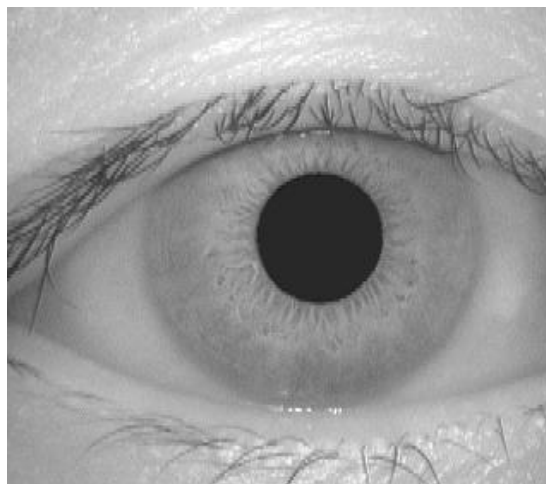
3.5.4 Tahapan Evaluasi

Proses evaluasi merupakan tahapan terakhir dari proses *iris recognition* pada penelitian ini. Dimana data yang digunakan merupakan data asli mata manusia yang didapatkan dari dataset CASIA V1, dengan jumlah 756 citra mata. Dengan menggunakan metode *K-Fold Cross-Validation*, data yang diolah berupa data *training* dan data *testing*, yang diambil secara acak sebagai input untuk pengujian. Hasil yang didapatkan dari pengujian ini akan digunakan oleh peneliti untuk mengetahui efektivitas sistem melalui parameter akurasi.

BAB IV HASIL DAN PEMBAHASAN

4.1 Tahap *Preprocessing*

Tahap *preprocessing* dalam penelitian ini merupakan sebuah tahapan untuk menentukan data citra iris yang akan diproses memenuhi syarat sistem atau tidak, sehingga dapat dikatakan jika pada tahap *preprocessing* tidak memenuhi syarat, maka sistem tidak akan dapat berjalan. Citra yang digunakan memiliki format warna *grayscale*, sehingga sistem tidak perlu mengubah citra menjadi *grayscale* untuk diproses pada tahapan segmentasi selanjutnya yang tentunya akan membuat kinerja sistem *Iris Recognition* akan lebih berat dan tidak efisien. Namun, bila citra iris yang diinput berupa citra iris RGB, maka diperlukan untuk mengubah menjadi citra iris *grayscale*. Citra iris diekstraksi menjadi tiga nilai dalam setiap pikselnya yaitu, nilai *Red* (merah), nilai *Green* (hijau), dan nilai *Blue* (biru). Nilai merah pada suatu piksel dikalikan dengan 0.2989, nilai hijau dikalikan dengan 0.5870, dan nilai biru dikalikan dengan 0.114. Selanjutnya, ketiga nilai hasil akan dijumlahkan dan menghasilkan citra *grayscale* dengan rentang 0-255.



Gambar 4.1 Citra *Grayscale*

Selain mengutamakan format warna *grayscale*, penelitian ini juga memberikan batasan untuk menggunakan citra yang berdimensi 320x280 piksel. Bila menggunakan citra dengan resolusi tinggi akan sangat membebani kinerja *hardware*.

4.2 Iris Segmentation

Tahap segmentasi citra iris menggunakan penggabungan dari dua metode yaitu metode *Image Processing* dimana menggunakan algoritma daugman yang digabungkan dengan *Euclidean Distance* yang dimiliki oleh metode FIRE. Dalam proses segmentasi, akan melewati tahapan mendeteksi dan menghapus refleksi, mendeteksi pupil, mendeteksi iris, serta melakukan proses menghilangkan bulu mata, normalisasi dan *encoding*.

4.2.1 Mendeteksi dan *Mapping* Refleksi Iris Mata

(Daugman, 2004) menyarankan *Integro-Differential Operator* (IDO) dan *Hough Transform* digunakan untuk mendeteksi kontur atau garis tepi pupil dan iris, dengan mengasumsikan garis tepi pupil dan iris berupa lingkaran serta dapat dicari dengan melalui proses citra iris yang telah dihaluskan (*smoothing*) menggunakan Filter Gaussian karena dapat mengurangi jumlah *noise* pada suatu citra iris tersebut. Parameter yang digunakan adalah nilai maksimum turunan integral dari titik pusat dan jari-jari dari garis tepi lingkaran iris, sesuai pada persamaan (2.1).

Simbol $*$ menunjukkan konvolusi (penghitungan yang dihasilkan dari penggabungan dua deret hasil perhitungan dari titik pusat dan jari-jari), $I(x, y)$ memiliki nilai intensitas piksel dalam koordinat okular citra iris, nilai jari-jari

lingkar iris di-*input* pada r , dengan titik tengah (x_0, y_0) . $G_\sigma(r)$ fungsi *Gaussian Smoothing* dengan nilai skala spasial, digunakan untuk menghaluskan citra iris dengan persamaan (2.2).

Setelah citra iris dihaluskan dengan menggunakan fungsi *Gaussian Smoothing* maka selanjutnya merupakan deteksi pantulan atau refleksi. Refleksi spekular muncul sebagai titik terang dan dikelilingi oleh nilai gradien yang besar. Pada proses mencari iris, kecerahan citra dihitung dan nilai ambang batas (*threshold*) ditentukan berdasarkan kecerahan citra. Proses *threshold* digunakan untuk mendeteksi area yang memiliki nilai intensitas yang tinggi. Kemudian, titik refleksi atau pantulan dideteksi dalam area yang memiliki intensitas tinggi dengan mencari tepi (*edge*) tertutup yang dapat mewakili refleksi dalam area pada citra iris tersebut. Hal ini digunakan untuk memastikan bahwa refleksi yang terdeteksi tersebut tergolong refleksi kuat atau lemah.

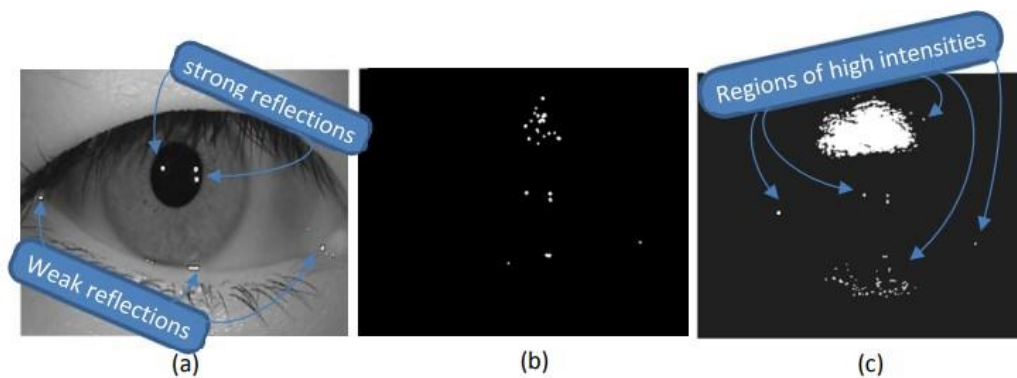
Sebuah citra iris $I(x, y)$, di mana (x, y) menunjukkan lokasi piksel area dengan intensitas tertinggi dapat dideteksi dengan persamaan (2.3) yakni sebagai berikut:

$$I(x, y) > \beta_s \quad (4.1)$$

Sesuai persamaan 2.4, $I(x, y)$ merupakan intensitas suatu piksel citra iris dan β_s nilai intensitas citra iris yaitu:

$$\beta_s = \frac{\sum_{i=1}^N I(x, y)}{N} \quad (4.2)$$

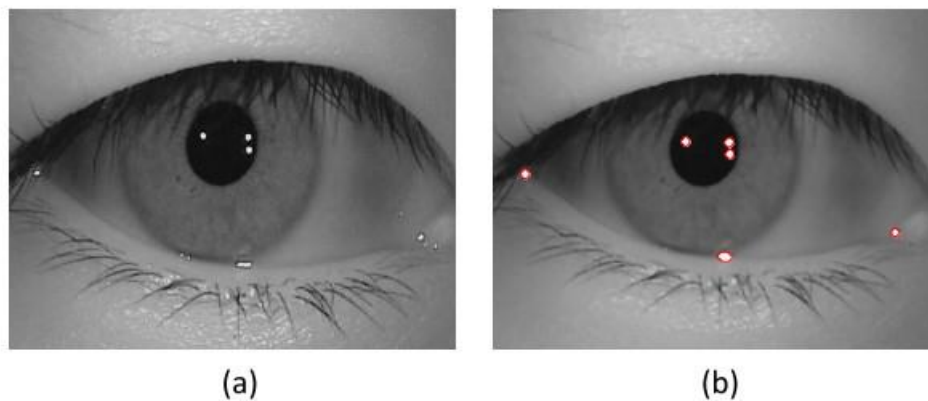
N merupakan jumlah total piksel pada citra iris yang dihitung dengan menggunakan operasi perkalian ukuran baris dengan ukuran kolom.



Gambar 4.2 Threshold Mapping Refleksi

Gambar 4.2(b) menunjukkan hasil *threshold* pada citra iris dengan intensitas paling terang 5%. Gambar 4.2(c) adalah hasil dari proses *threshold* adaptif. Gambar 4.2(b) menunjukkan bahwa metode *threshold* dengan menggunakan intensitas paling terang 5% gagal mendeteksi refleksi lemah disisi kiri pada citra mata tersebut. Sehingga dibutuhkan metode *threshold* adaptif untuk mendeteksi refleksi lemah, seperti pada persamaan (4.3) dan (4.4). Gambar 4.2(c) menunjukkan area yang memiliki rona (HUE) saturasi sebagian besar terdeteksi, hal ini dikarenakan penerapan *threshold* adaptif sehingga titik refleksi telah dipetakan (*mapping*).

Setelah area dengan intensitas tertinggi telah ditandai seperti yang ditunjukkan pada Gambar 4.2(c) digunakan filter *multi-scale edge-weighted highpass* untuk mendeteksi kontur tertutup (*closed contours*) yang mana merepresentasikan batas titik refleksi. Kontur tertutup dipetakan sebagai titik refleksi yang merepresentasikan batas titik refleksi pada citra iris yang dapat ditunjukkan pada Gambar 4.3.



Gambar 4.3 Peta Refleksi

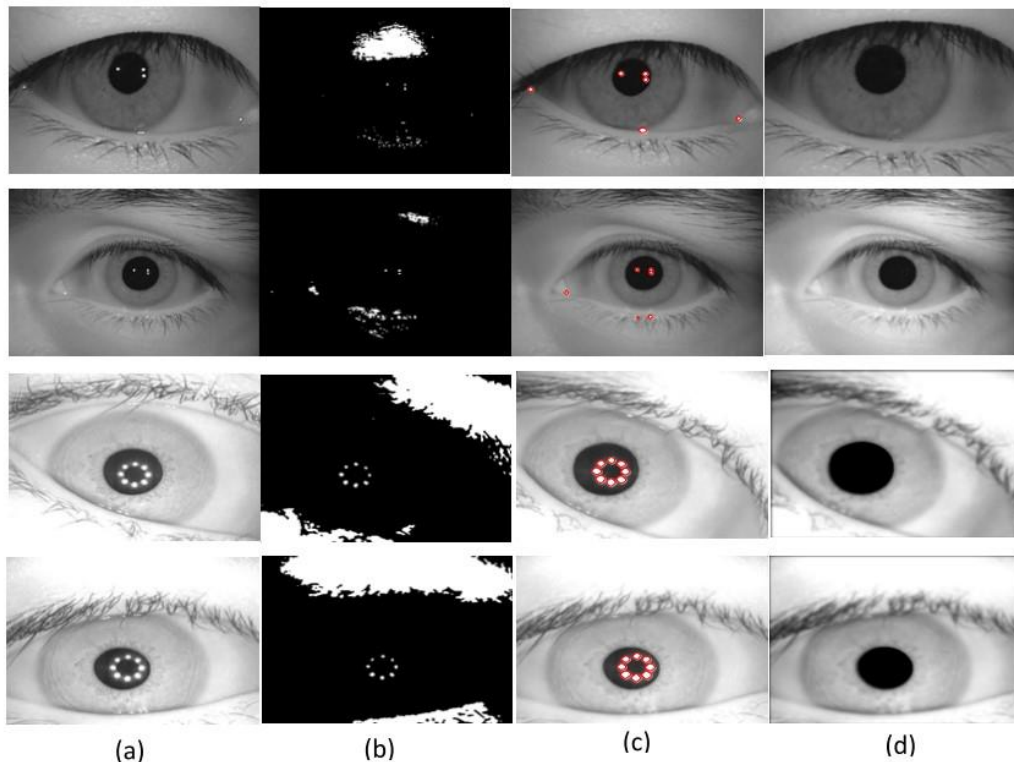
Gambar 4.3(a) citra iris *original*, Gambar 4.3(b) citra iris dengan peta refleksi. Pada penelitian ini menggunakan citra iris mata dari dataset CASIA yang menunjukkan nilai intensitas refleksi pada citra iris lebih besar dibanding nilai kecerahan citra iris itu sendiri. Area dengan intensitas tertinggi mencakup titik refleksi dan bagian kelopak mata yang memiliki saturasi rona (HUE) terlihat pada Gambar 4.2(c). Gambar 4.3(b) menggambarkan tepi yang membentuk kontur tertutup yang merepresentasikan refleksi dan ditandai dengan peta refleksi.

Citra iris selanjutnya akan difilter menggunakan filter *multi-scale edge-weighted highpass* adaptif untuk mendeteksi tepi (*edges*) yang bervariasi dimana tepi tersebut merepresentasikan batas refleksi. Filter *highpass* sesuai dengan persamaan (2.5).

Citra yang telah dihaluskan (*smoothed*) $f(x,y)$ selanjutnya akan dikomposisi dengan menerapkan *discrete wavelet transform* (DWT). Analisis multiskala menghasilkan satu set filter *highpass* yang memiliki nilai tepi sehingga dapat diimplementasikan ke tepi (*edges*) pada skala yang akan ditentukan. Oleh karena itu, sesuai persamaan (2.7) peta refleksi $R(x,y)$ sebagai berikut:

$$R(x, y) = \sum_{p=1}^P \sum_{j=1}^J |D_{2^j} f|^p Hf \quad (4.3)$$

Persamaan (4.3) dapat merepresentasikan batas-batas titik refleksi. Peta (*mapping*) area refleksi dapat dilihat pada Gambar 4.4(b), sedangkan Gambar 4.4(c) batas-batas titik refleksi. Setelah titik refleksi terdeteksi, pemrosesan lanjut akan dilakukan untuk menghilangkan refleksi.



Gambar 4.4 Titik-Titik Refleksi dan Batas Titik Refleksi

4.2.2 Menghapus Refleksi

Setelah pendeteksian refleksi dilakukan modifikasi adaptif pada citra iris secara *in-painting* untuk mengisi titik refleksi. Modifikasi yang dilakukan telah banyak digunakan dalam proses *computer vision* untuk memulihkan citra *original* yang memiliki beberapa cacat (Criminisi et al., 2004; Hasegawa et al., 2013). Proses *in-painting* citra digunakan untuk mengisi area cacat yang dianggap sebagai “lubang” dalam sebuah citra tersebut dengan mencari informasi kemiripan

dengan nilai terbaik dari daerah yang tidak cacat lalu menyalinnya ke area yang cacat.

Kelemahan yang dimiliki dari proses *in-painting* adalah rekonstruksi tepi (*edge*) yang salah dan memakan cukup waktu sehingga diperlukan modifikasi untuk model *in-painting* untuk lebih mudah dalam mendeteksi tepi dan membatasi area terhadap batas lubang titik refleksi untuk meningkatkan kecepatan algoritma sehingga dapat menekan waktu yang digunakan pada proses menghapus refleksi pada citra iris mata (Criminisi et al., 2004).

Diberikan citra iris $I(x,y)$ dengan peta area titik refleksi yang dilambangkan dengan Ω sedangkan batas titik refleksi atau kontur dilambangkan $\partial\Omega$. Peneliti menggunakan sebutan “lubang” atau “wilayah isi” untuk merujuk kepada titik refleksi dan “wilayah target” merujuk pada sisa citra utuh.

Untuk menghitung kesamaan antara wilayah isi dan wilayah target, maka potongan citra harus dipilih pada satu waktu yang bersama. Potongan tersebut dilambangkan ψ_i yang merupakan *sub-wilayah* berbentuk persegi dari citra yang berpusat pada piksel i . Pada umumnya, ukuran ψ_i memiliki dimensi sedikit lebih besar dari elemen tekstur terbesar yang terdeteksi. Ukuran potongan paling umum berukuran 9×9 pada citra pada umumnya, namun pada penelitian ini ukuran potongan yang digunakan berukuran 3×3 karena lebih efektif diterapkan pada citra memiliki tekstur tinggi seperti citra iris yang digunakan pada penelitian ini. Potongan pengisi dilambangkan ψ_p memiliki informasi dari bagian wilayah pengisi (lubang) dan bagian dari wilayah target.

Setiap potongan wilayah pengisi ψ_{pi} dipilih dengan melakukan pencarian dari wilayah target. Ketika N -potongan yang paling mirip ditemukan, maka

potongan ψ_{pi} digabungkan secara linier dengan potongan ψ_{qi} untuk memprediksi wilayah ψ_p yang tidak diketahui atau terdeteksi. Untuk membatasi area pencarian pada batas refleksi, maka titik prioritas diberikan pada potongan tersebut dimana didasarkan oleh jarak.

Prioritas untuk urutan pengisian wilayah iris $P(p)$ didasarkan atas aturan konfiden (*confidence*) $C(p)$ dan aturan data $D(p)$ sesuai dengan persamaan (2.8).

Dimana,

$$C(p) = \frac{\sum_{q \in \psi_p \cap \bar{\Omega}} C(q)}{\psi_p}, \quad D(p) = \frac{|\nabla I_p^\perp \cdot n_p|}{\alpha} \quad (4.4)$$

Dalam peningkatan rekontruksi tepi, data dioptimalkan secara non-linier dengan memodifikasi persamaan (2.8) sehingga menjadi persamaan (4.5).

$$\hat{P}(p) = C(p) * \exp\left(\frac{D(p)}{2\sigma^2}\right) \quad (4.5)$$

Hasil dari metode *in-painting* terdapat pada Gambar 4.4(d).

4.2.3 Menentukan Pupil

Pada penelitian ini, transformasi hough dengan model *graph cut based active contour* (GBAC) yang membantu dalam proses ekstraksi citra karena dapat mendeteksi lingkaran pusat pupil dan lingkaran iris dengan baik. Lingkaran iris terdapat dua batas dimana batas dalam disebut batas pupil dan batas luar disebut batas iris atau skelera. GBAC mengadopsi pendekatan dengan 2 tingkat dan 2 langkah, dimana akan melakukan segmentasi pupil terlebih dahulu sebelum melakukan segmentasi pada batas limbik (*limbic boundaries*) iris mata. Setiap tingkat proses segmentasi dilakukan dalam dua langkah yaitu dengan pendekatan batas awal terlebih dahulu sebelum melakukan penentuan batas akhir iris untuk proses segmentasi. Dengan pendekatan tersebut digunakan untuk melacak dan

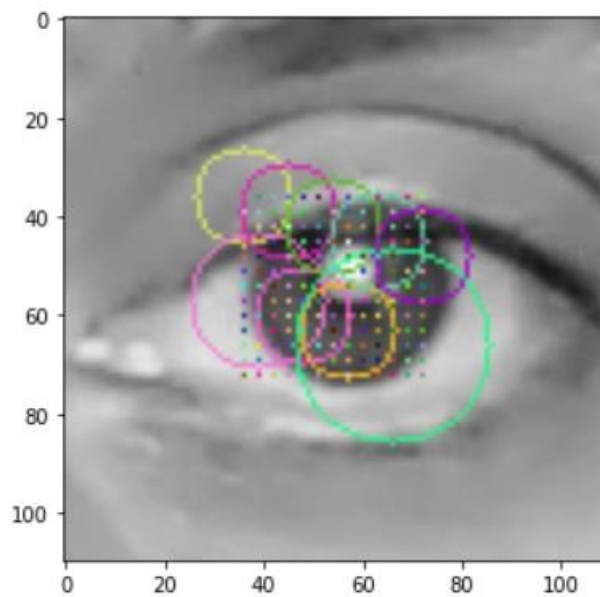
menangani kesalahan yang mungkin muncul pada tingkatan atau level manapun, sehingga dapat dipastikan algoritma tersebut selesai diproses pada tingkat itu dan tidak membawa kesalahan ke tingkat berikutnya yang dapat menyebabkan kegagalan keseluruhan algoritma.

Langkah pertama algoritma memulai mendekati batas pupil dengan pendekatan secara kasar dengan menggunakan metode yang didasarkan pada lingkaran transformasi Hough. Kemudian, batas akhir pupil disegmentasi dengan menyempurnakan kontur pada perkiraan awal dengan berdasarkan model GCBAC. Setelah segmentasi batas pupil, proses yang sama diulangi lagi untuk mensegmentasikan batas limbuk iris kecuali bahwa pendekatan awal batas limbuk dilakukan dengan menggunakan metode berbasis energi. Estimasi kasar dari batas limbuk diperoleh dengan menggunakan algoritma *starburst* berbasis energi yang telah dimodifikasi. Langkah pertama dari proses segmentasi iris adalah prasyarat untuk inisialisasi kontur. Tujuan utamanya adalah untuk menginisialisasi kontur aktif yang cukup dekat dengan batas sebenarnya dari pupil atau iris sehingga dapat meningkatkan akurasi segmentasi dan mengurangi waktu komputasi.

Pupil merupakan bagian gelap dari citra mata yang dikelilingi oleh iris dengan warna relatif lebih terang. Hal tersebut mengakibatkan terdapat kontras yang besar antara pupil dan iris. Lingkaran transformasi Hough digunakan untuk mencari dan mendeteksi lingkaran dalam citra iris yang telah diproses sebelumnya. Lingkaran yang terdeteksi disebut calon pupil C_p . Skema evaluasi dikembangkan untuk menemukan calon pupil C_{pi} yang mana sesuai dengan pupil sebenarnya. Segmentasi akhir dari batas pupil selanjutnya didapatkan dengan

menyempurnakan kontur awal pupil yang dihasilkan menggunakan model kontur aktif berbasis potongan grafik (GCBAC).

Bentuk pupil umumnya mendekati bentuk lingkaran, namun terkadang dapat berubah karena terhalang sebagian bagian pupil dikarenakan oleh kelopak mata atau bulu mata. Dari proses yang dilakukan oleh transformasi hough, citra iris menghasilkan banyak calon pupil seperti yang ditunjukkan pada Gambar 4.5 dan 4.6(c).



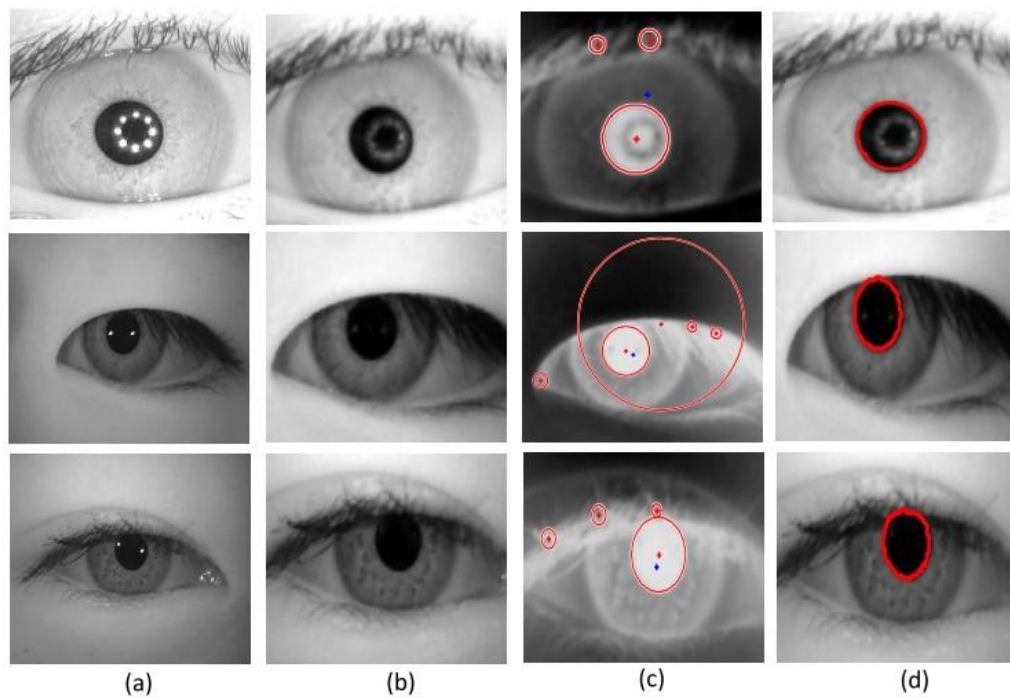
Gambar 4.5 Deteksi Pupil Asli dan Palsu

Calon pupil pada Gambar 4.5 menunjukkan pupil asli namun juga banyak calon pupil palsu yang biasanya berasal dari daerah bulu mata. Untuk memilih pupil yang asli dari calon pupil, maka ada proses memilih dan menghilangkan pupil palsu, dengan membuat beberapa asumsi berdasarkan beberapa fitur yang berbeda pada pupil yaitu:

- a. Pupil memiliki bentuk lingkaran dengan intensitas warna paling gelap atau pupil juga dapat dikatakan sebagai bagian lingkaran yang terdekat dari titik pusat citra iris.

- b. Calon pupil yang asli memiliki variasi intensitas yang paling halus.

Berdasarkan asumsi tersebut, peneliti melakukan segmentasi kasar terhadap pupil menggunakan metode serupa seperti pada penelitian (Feng et al., 2006). Seluruh citra pada dataset dicari bentuk lingkaran seperti pupil dengan menggunakan transformasi hough, setelah itu lingkaran yang memiliki jarak terpendek ke pusat citra iris akan dipilih sebagai pupil asli. Gambar 4.6(c) menunjukkan beberapa contoh citra iris dengan banyak lingkaran calon pupil yang terdeteksi. Penanda biru yang ditunjukkan pada Gambar 4.6(c) menunjukkan bagian tengah dari citra iris, sedangkan penanda tanda bintang merah menunjukkan bagian tengah dari lingkaran pupil asli.



Gambar 4.6 Deteksi Pupil

Citra iris sebelum diproses harus melewati proses modifikasi menjadi citra yang berukuran $(n \times n)$ yang berfungsi untuk mengurangi *noise*. Transformasi

Hough menggunakan parameter lingkaran pusat koordinat x_0^2 dan y_0^2 serta radius r_0^2 akan dihitung untuk evaluasi pada ruang Hough, sesuai dengan persamaan (2.21).

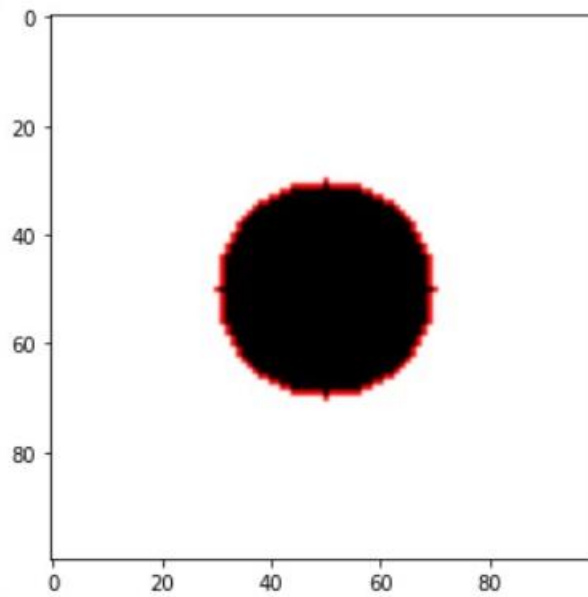
Lingkaran transformasi Hough kemudian diterapkan untuk mencari calon pupil C_p pada citra iris. Jarak Euclidian (*Euclidean Distance*) dari semua C_p menuju titik pusat citra iris kemudian dihitung. C_p yang memiliki jarak Euclidean terendah atau terdekat ke pusat citra iris mata kemudian dipilih sebagai pupil.

Dengan mempertimbangkan beberapa citra iris yang tidak ideal, rotasi mata dapat menyebabkan iris terletak lebih dekat ke batas citra iris daripada titik pusat citra. Oleh karena itu, untuk mengimbangi hal tersebut dan untuk menghindari kesalahan dalam memilih calon pupil palsu yang lebih dekat ke tengah citra sebagai pupil, peneliti menyempurnakan calon pupil dengan syarat tambahan.

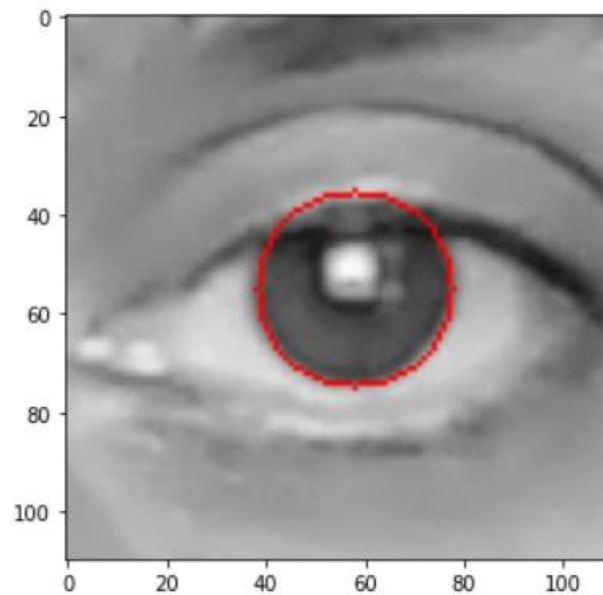
Mengukur entropi calon pupil C_p dapat membantu untuk membedakan perbedaan calon pupil asli dari bentuk bentuk calon pupil palsu. Peneliti menganalisis, calon pupil palsu yang terdeteksi rata-rata terletak didalam daerah bulu mata yang gelap yang mengakibatkan calon pupil palsu tersebut memiliki nilai entropi yang lebih tinggi (variasi intensitas lebih tinggi) dibandingkan dengan calon pupil asli.

Nilai intensitas yang disesuaikan I^{c_p} dari calon pupil C_{pi} dihitung dengan mengonversi nilai intensitas pada citra dengan matriks dengan ukuran 5×5 yang digunakan untuk menghaluskan citra dan mengurangi *noise*. Transformasi Hough digunakan untuk mendeteksi calon pupil di dalam citra. Persamaan (2.18) calon pupil asli dipilih berdasarkan jarak dari pusat citra serta variasi intensitas (entropi) dalam lingkaran calon pupil.

Gambar 4.7 menghasilkan pupil asli atau pupil sejati dengan menghitung nilai jarak Euclidean antara calon pupil dan titik pusat citra iris pada d^2 , ukuran dispersi H didalam calon pupil C_p dan menilai nilai intensitas tertinggi sesuai dengan persamaan (2.20) maka pupil akan terdeteksi. Hasil dari proses ini seperti yang ditunjukkan pada Gambar 4.6(d) dan 4.8.



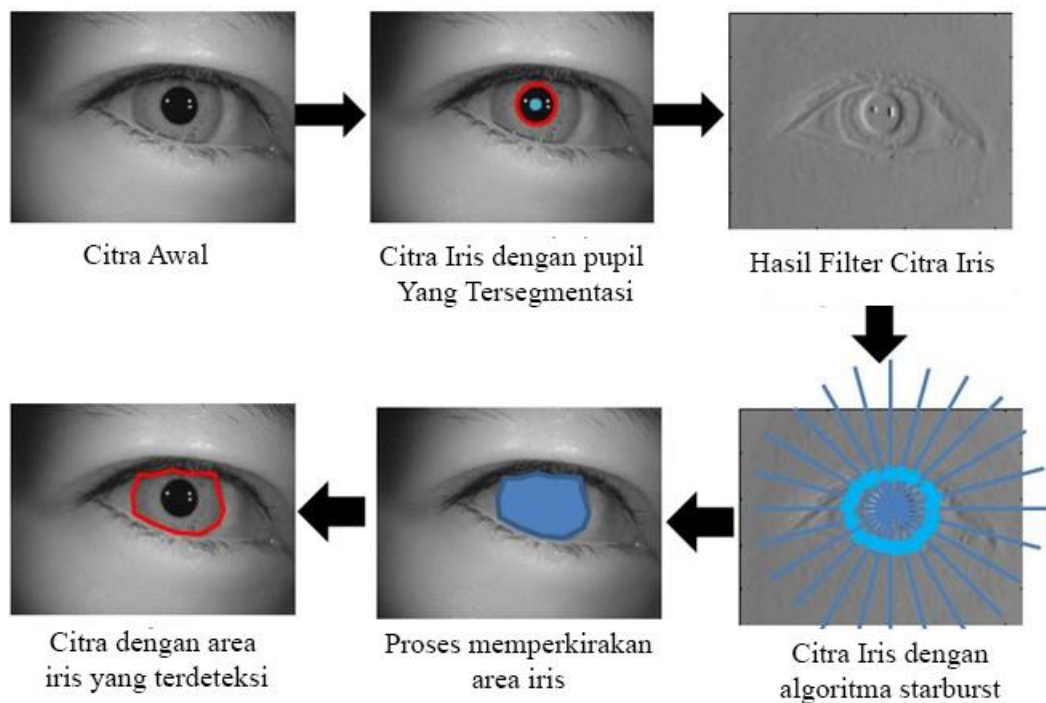
Gambar 4.7 Pupil Asli



Gambar 4.8 Pupil Iris Mata Terdeteksi

4.2.4 Menentukan Batas Iris

Untuk mendekati batas limbik iris, peneliti menggunakan algoritma *starburst* yang dimodifikasi. Algoritma *Starburst* telah berhasil digunakan untuk pemrosesan gambar berkualitas rendah dan dalam teknologi pelacakan lokasi mata. Algoritma *starburst* memberikan perkiraan kasar lokasi batas limbik iris untuk inisialisasi kontur. Gambar 4.9 menunjukkan proses aproksimasi batas limbik iris menggunakan algoritma *starburst* yang dimodifikasi.

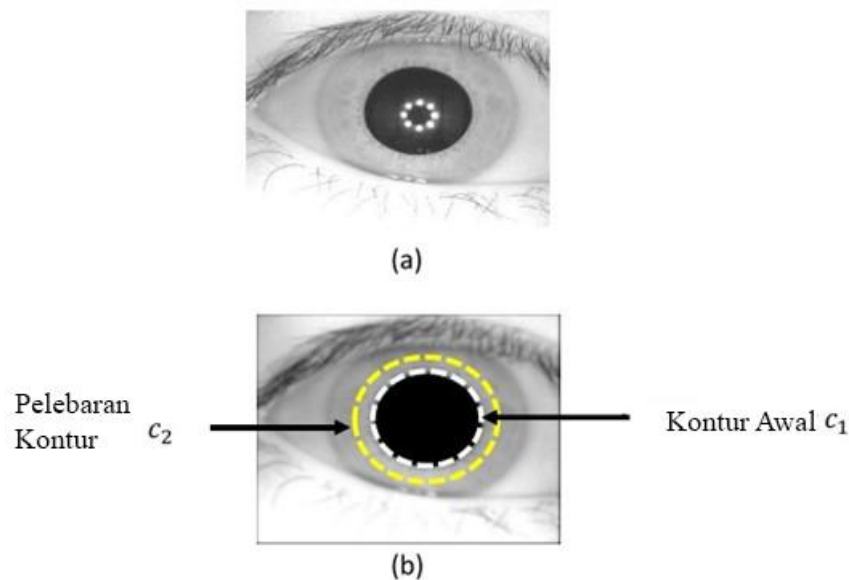


Gambar 4.9 Mendeteksi Lingkar Iris Dengan Algoritma *Starburst*

Dalam proses mencari iris, dibutuhkan *starburst* dalam algoritma *graph cut based active contour* (GCBAC). Ekstraksi batas iris yang akurat merupakan masalah penting pemrosesan citra. Ekstraksi batas iris yang akurat dan optimal memiliki tantangan besar yang meliputi oklusi batas, *noise* dan kecacatan lain pada citra.

Proses pendekatan berbeda dari kontur aktif tradisional karena menggunakan energi potong grafik untuk mengubah bentuk kontur secara *iterative*, sehingga menunjukkan bahwa kontur yang dihasilkan pada setiap iterasi adalah optimum global dalam *contour neighbourhood* (CN). Menurut (Michailovich et al., 2007) penggunaan *contour neighbourhood* (CN) digunakan untuk mengatasi kelemahan citra bias yang didapatkan dari potongan grafik ke batas yang lebih pendek. Selain itu, kombinasi dari dua pendekatan (pemotongan grafik dan kontur aktif) mampu mengatasi kelemahan model kontur aktif lainnya.

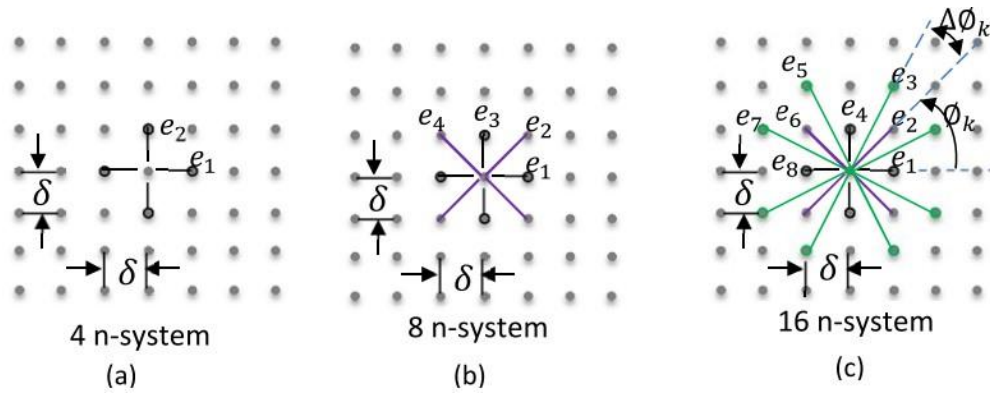
Misalkan c_1 menunjukkan kurva awal yang menyatakan perkiraan pupil dan batas limbik iris yang didapatkan dari proses mencari batas pupil sebelumnya. Serta misalkan c_2 menyatakan kontur baru yang diperoleh dengan proses melebarkan kontur awal ke area sekitarnya seperti Gambar 4.10.



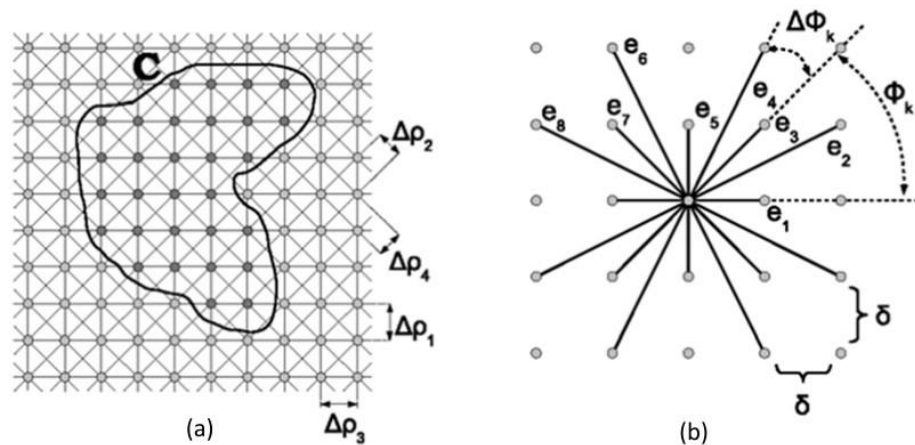
Gambar 4.10 Proses Mencari Iris

Penelitian serupa dalam proses mencari iris oleh (Ukpai, n.d. 2015) menggambarkan proses dalam mencari iris yang diawali dari pelebaran kontur

awal menuju area sekitarnya (piksel tetangga yang memiliki kontras berbeda) seperti pada Gambar 4.11.

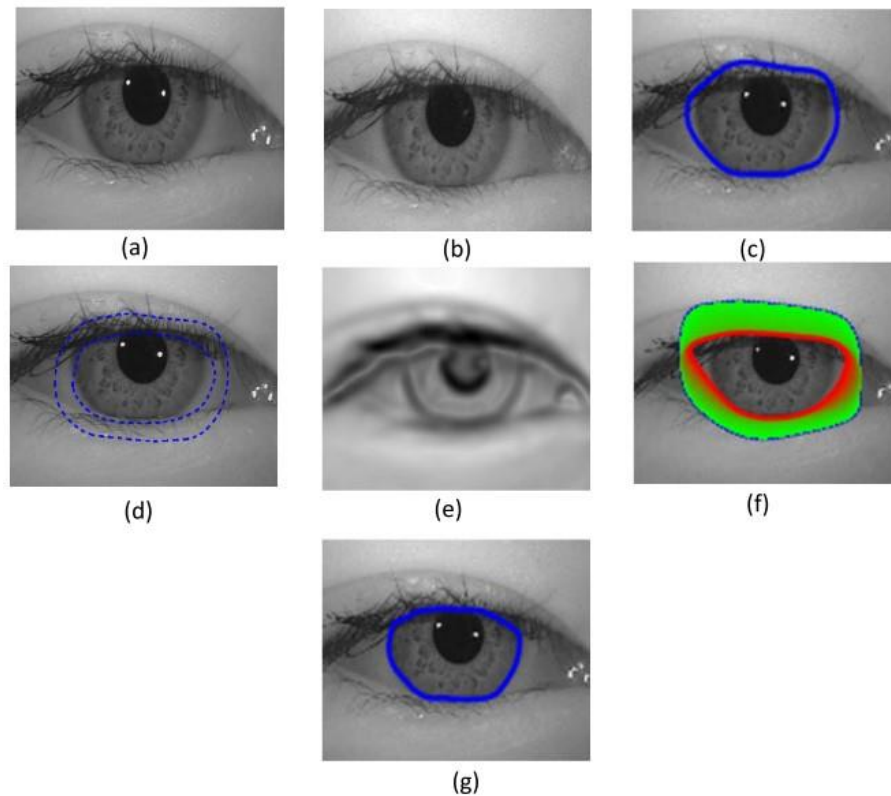


Gambar 4.11 Mencari Posisi Iris Pada Piksel Sekitar



Gambar 4.12 Mencari Iris Dengan Kontur Area Sekitar

Sehingga dari proses mencari iris dengan mencari kontur area sekitar (piksel sekitar) menghasilkan lokasi iris mata yang sesuai, Gambar 4.13 menunjukkan tahapan citra iris mata dalam menentukan iris.



Gambar 4.13 Tahapan Mencari Iris Mata

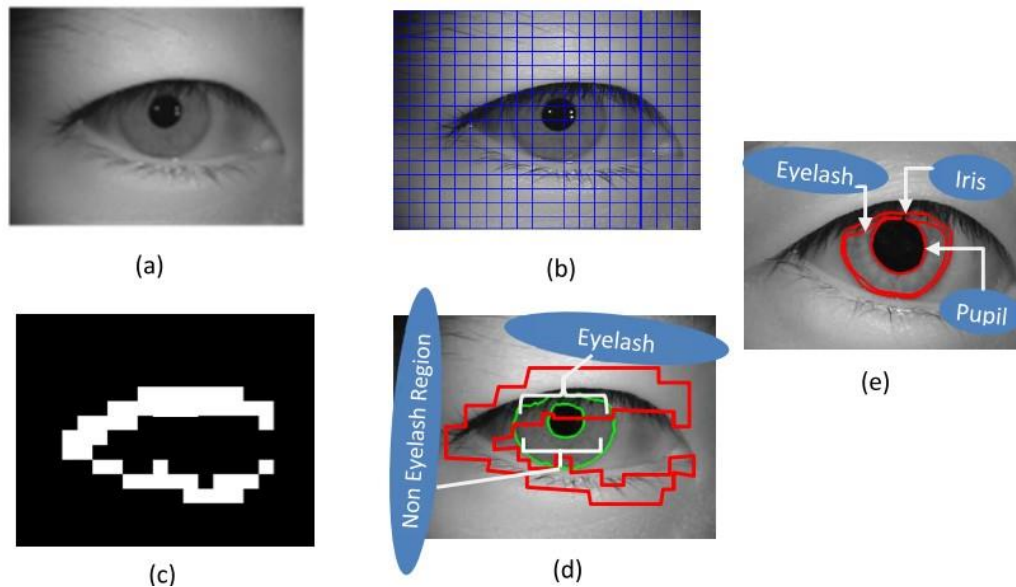
4.2.5 Mendeteksi Bulu Mata

Bulmata dapat merusak keutuhan iris pada citra iris, mengakibatkan dapat menghambat akurasi pengenalan jika tidak terdeteksi, sehingga diperlukan untuk menghilangkan bagian iris yang tertumpuk oleh bulmata. Pada umumnya bulmata yang terdapat pada citra iris memiliki intensitas tinggi berwarna gelap dibandingkan intensitas yang dimiliki oleh iris.

Sesuai persamaan (2.12) misalkan $I(N, M)$ merupakan citra iris asli yang dibagi menjadi blok-blok dimana setiap blok merupakan *sub*-citra dari citra iris aslinya. Maka, untuk setiap blok $n \times m$ akan dihitung menggunakan *Discrete Wavelet Transform* (DWT), yang menggunakan teknik *multiscale frequency decomposition*. Setelah blok pada citra asli terbagi, maka selanjutnya menghitung nilai maksimum energi lokal dalam balok tersebut dengan persamaan (2.13).

Dengan nilai frekuensi tertinggi $f(k)$ dari *sub-bands*, selanjutnya nilai tersebut dalam proses pengelompokan *block-wise* untuk mencari wilayah yang sesuai atau *Region of Interest* (ROI) ditunjukkan pada Gambar 4.14(c). Klasifikasi blok menjadi daerah kandidat bulu mata atau *candidate eyelash region* (CES) dan daerah non-bulu mata atau *non-eyelash* (NES) dicapai dengan menggunakan pendekatan klasifikasi Bayesian. Tiga blok didefinisikan sebagai berikut yaitu blok B_H frekuensi tinggi, blok B_L frekuensi rendah dan blok B_M frekuensi menengah. Blok frekuensi tinggi mencakup wilayah dalam citra dengan frekuensi tinggi (tepi tajam) seperti blok bulu mata, dan blok tepi pupil. Blok frekuensi rendah (tanpa tepi sama sekali atau tepi yang sangat halus) mencakup blok seperti blok tepi limbik dan area halus lainnya pada citra iris, sedangkan blok frekuensi menengah mencakup blok dengan tepi lemah seperti blok bulu mata yang dapat dipisahkan. *Block of Interest* (BOI) merupakan jumlah dari blok-blok frekuensi menengah B_M dan frekuensi tinggi B_H . Semua blok akan dicari nilainya dengan *k-means classifier* untuk mengelompokkan E_n menjadi tiga kelas, selanjutnya klasifikasi akhir menggunakan *bayes classifier*.

Setiap piksel pada iris jika memiliki rata-rata $\mu E_i^{\vec{d}}$ (energi dalam arah horizontal, vertikal, dan diagonal) lebih besar dari salah satu dari delapan nilai dari piksel tetangga (*neighbours*) dan lebih kecil dari T , maka piksel tersebut dianggap sebagai piksel bulu mata. Gambar 4.14 menunjukkan hasil dari pencarian area bulu mata pada citra iris.



Gambar 4.14 Mendeteksi Bulu Mata

4.2.6 Normalisasi Iris Mata

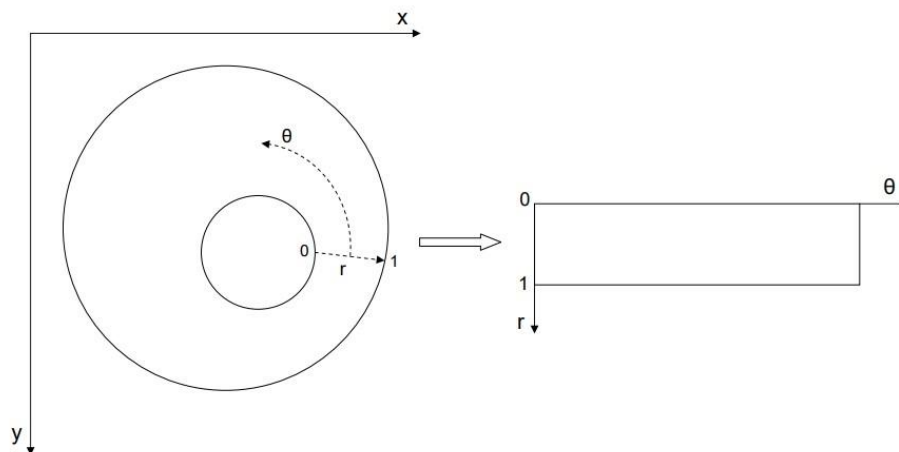
Setelah area iris diekstraksi, tahap normalisasi diperlukan untuk menghilangkan inkonsistensi dimensi antar iris. Normalisasi diperlukan untuk mendapatkan dua set fitur yang konsisten dari citra mata dua dimensi yang berbeda dari orang yang sama. Peneliti menggunakan Daugman *Rubber Sheet* untuk menormalisasi citra iris. Peneliti membuat *template* biner *.mat* berdasarkan kontur iris yang telah diekstraksi dan bulu mata yang terdeteksi. Dari proses segmentasi sebelumnya diketahui area iris yang terdeteksi tidak memiliki bentuk lingkaran atau elips yang sempurna, serta memiliki bentuk yang berbeda-beda pada setiap citra, maka akan digunakan sebuah strategi mencari lingkaran yang tepat oleh (Park & Savvides, 2015) untuk mengkompensasi bentuk serta ukuran dari iris.

Proses mengubah area iris yang terdeteksi menjadi bentuk persegi panjang, maka pertama-tama akan menentukan jari-jari dan koordinat titik pusat iris yang sesuai sangat diperlukan. Penelitian ini menentukan jari-jari R pada enam titik

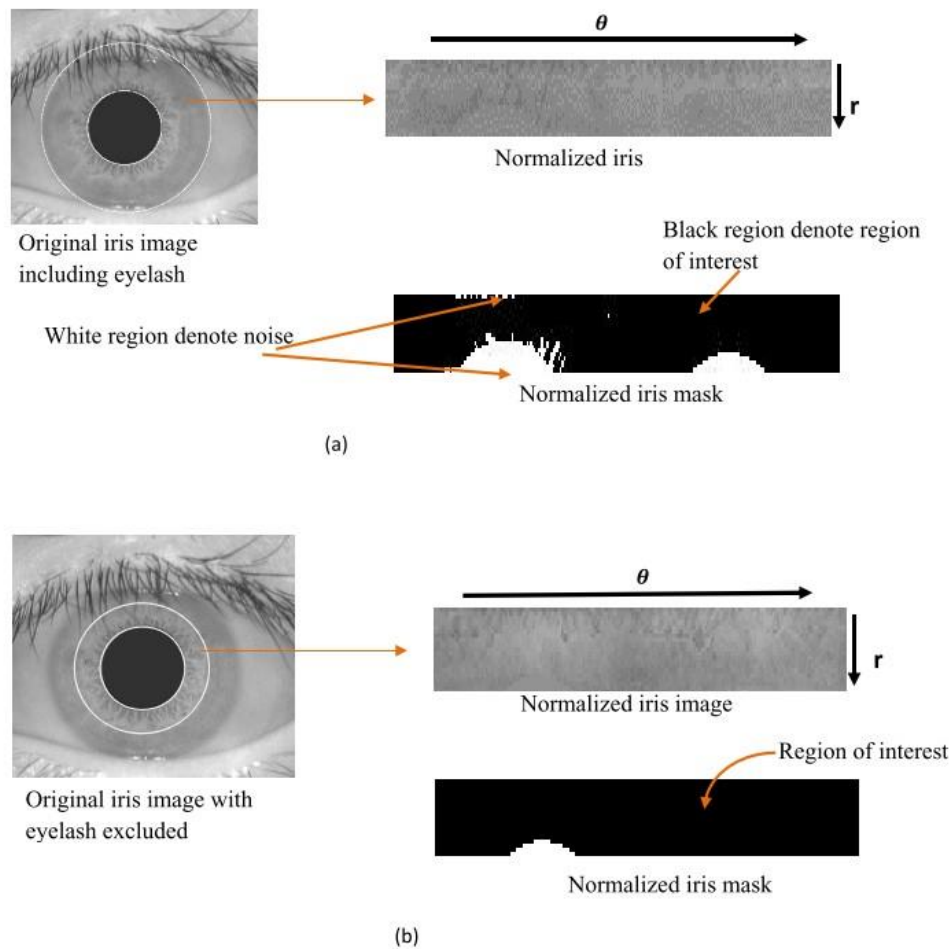
yang dipilih dari kontur iris yang telah diekstraksi yaitu pada sudut $[-30^\circ, 30^\circ, 0^\circ, 150^\circ, 180^\circ, 210^\circ]$ dengan terhubung sumbu horizontal dan perhitungan jarak rata-rata dari pusat pupil. Kemudian strategi mencari lingkaran tersebut digunakan pada semua titik piksel yang berada dalam radius atau jari-jari $R \pm 15$ piksel dari pupil.

Nilai pusat yang diperoleh melalui proses *fitting* elips dan radius perkiraan seperti lingkaran akan digunakan untuk proses *unwrapping*. Normalisasi menghasilkan array 2-D dengan dimensi horizontal yang sesuai dengan resolusi sudut, dan dimensi vertikal yang sesuai dengan resolusi radial.

Untuk mengimbangi deformasi elastis dari tekstur iris, maka area iris yang terlokalisasi dibuka menjadi blok persegi panjang yang dinormalisasi dengan ukuran tetap 20×480 piksel, dengan mengubah dari sistem koordinat Cartesian ke sistem koordinat kutub.

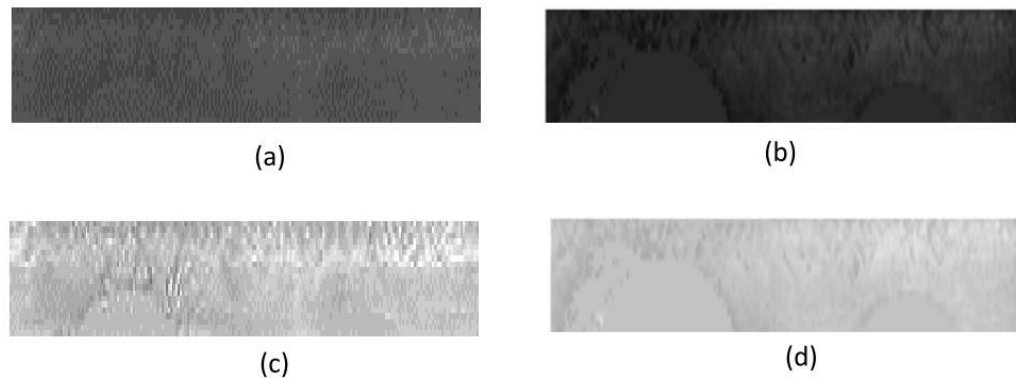


Gambar 4.15 Proses *Unwrapping* Iris Mata



Gambar 4.16 Perbedaan Citra Iris Yang Telah Bersih Dari Bulu Mata

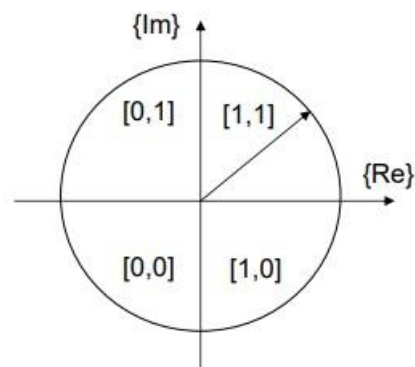
Gambar 4.17(a) dan (b), menunjukkan citra iris yang dinormalisasi. Karena citra iris yang dinormalisasi memiliki kontras yang relatif rendah, dan mungkin juga dipengaruhi oleh iluminasi yang tidak seragam karena penempatan sumber cahaya, teknik pemerataan histogram berbasis intensitas lokal diterapkan untuk menyempurnakan citra. Teknik ini meningkatkan kualitas citra iris yang dinormalisasi, sehingga meningkatkan akurasi pengenalan. Gambar 4.17(c) dan (d) menunjukkan efek peningkatan citra iris menggunakan teknik ekualisasi histogram.



Gambar 4.17 Peningkatan Kualitas Citra *Template* Iris

4.2.7 Fitur *Encoding*

Fitur *encoding* dalam proses pengenalan iris dilakukan untuk ekstraksi fitur iris dan pembuatan *template* untuk perbandingan antara citra iris yang berbeda. Prosedur pengkodean ini terdapat pada persamaan (2.22). Daugman menggunakan filter gabor 2D untuk mengekstraksi informasi *local* frekuensi. Koefisien bilangan kompleks yang merepresentasikan *phasor* (fasor) dalam bidang kompleks dihitung, kemudian sesuai dengan lokasi, sudut vektor akan dikuantisasi ke salah satu dari empat segmen lingkaran seperti pada Gambar 4.18 yang memberikan dua bit ke dalam *template* iris.



Gambar 4.18 Representasi Fasor

Dalam prosedur pengkodean akan dicari nilai biner yang dimiliki oleh setiap piksel pada *template*. Nilai yang dihasilkan $h_{\{Re,Im\}}$ sesuai dengan bagian

nyata dan imajiner terkuantisasi dari sebuah fasor. Setelah diaplikasikan kesetiap titik piksel iris yang dinormalisasi, maka diperoleh 2048-bit informasi. Dari proses tersebut maka akan menghasilkan *template* iris seperti Gambar 4.19.



Gambar 4.19 Representasi Citra *Template* Iris

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
20	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
21	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
22																									
23																									
24																									
25																									

Gambar 4.20 Nilai Biner *Template Iris*

```

▶ Number of files for enrolling: 756
Start enrolling...
0% 0/756 [00:00<?, ?it/s]/content/Skripsi_Iris
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:130:
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:130:
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:130:
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:130:
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:130:
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:130:
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:172:
mask[grid] = np.nan
0% 1/756 [00:05<1:08:25, 5.44s/it]/content/Sk
mask[grid] = np.nan
0% 2/756 [00:06<33:24, 2.66s/it] /content/Sk
mask[grid] = np.nan
0% 3/756 [00:06<20:38, 1.64s/it]/content/Skri
mask[grid] = np.nan
1% 4/756 [00:06<13:45, 1.10s/it]/content/Skri
mask[grid] = np.nan
/content/Skripsi_Iris/python/fnc/segment.py:172:
mask[grid] = np.nan
1% 6/756 [00:07<07:20, 1.70it/s]/content/Skri
mask[grid] = np.nan
100% 756/756 [11:30<00:00, 1.09it/s]

>>> Enrollment time: 690.9633502960205 [s]

```

Gambar 4.21 Total Waktu Segmentasi

Gambar 4.21 menunjukkan total waktu segmentasi pada dataset CASIA berjumlah 756 citra iris mata yaitu 690,963 detik. Jika waktu dibagi dengan total citra iris mata, maka setiap iris mata membutuhkan waktu 0,913 detik.

4.3 Perbandingan Iris

Proses membandingkan dua *template* iris menggunakan algoritma *Hamming Distance* serta dibantu dengan *Weighted Euclidean Distance* (WED) bertujuan untuk mengukur seberapa berbeda *template* dan untuk memutuskan apakah mereka milik individu yang sama atau tidak. Metode yang dirancang langsung oleh Daugman yaitu melalui pendekatan dengan *Hamming Distance*.

Serta menghitung jumlah bit yang berbeda dalam urutan biner X dan Y atas jumlah total M bit. Daugman memperluas pendekatan ini untuk *template* iris yaitu:

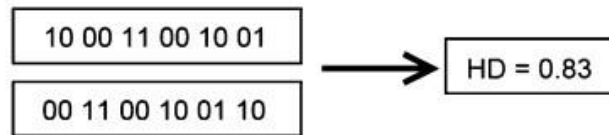
$$HD = \frac{\|(templateA \otimes templateB) \cap maskA \cap maskB\|}{\|maskA \cap maskB\|} \quad (4.6)$$

Dimana, *templateA* dan *templateB* merupakan dua *template* iris yang dibandingkan untuk mencari kesamaan, *maskA* dan *maskB* adalah *mask* yang sesuai untuk menentukan bit *template* mana area iris yang valid (tidak terhalang oleh kelopak mata atau bulu mata dan tidak memiliki pantulan spekular). *Hamming Distance* dan WED akan memiliki nilai 0 untuk menyatakan dua *template* yang sama atau cocok sempurna. Karena independensi *template* dari iris yang berbeda, diharapkan untuk memperoleh satu setengah dari bit yang cocok dan setengah lainnya yang tidak cocok untuk dua iris yang berkaitan dengan mata yang berbeda. Dapat disimpulkan bahwa sangat tidak mungkin terjadi kurang dari sepertiga bit *template* tidak akan setuju jika iris yang dibandingkan berbeda.

Selain langkah-langkah yang dijelaskan di atas, Daugman memodifikasi tahap pencocokan *template* iris untuk mengkompensasi kemungkinan rotasi iris, yang terjadi karena kemiringan kepala. Untuk melakukan ini, salah satu *template* iris yang dibandingkan akan diputar secara siklik. Sehingga dapat dengan mudah ditunjukkan bahwa memutar *template* dalam koordinat polar setara dengan rotasi iris dalam koordinat Cartesian. Algoritma *Hamming Distance* melakukan pencocokan dua *template* beberapa kali sambil menggeser salah satunya ke n lokasi berbeda.

Dengan proses yang menyerupai *Hamming Distance*, *Weighted Euclidian Distance* (WED) membantu menyempurnakan hasil perbandingan iris, sehingga

hasil yang diperoleh akan mendekati nilai yang sesungguhnya serta meminimalisir kegagalan pencocokan dari *Hamming Distance*.



Gambar 4.22 Perbandingan Dua *Template* Tanpa Pergeseran

Tabel 4.1 Hasil *Hamming Distance* dan *Euclidean Distance*

No.	<i>Template Iris</i>	HD	WED	Nilai Rata-Rata
1.	<p>Geser kekanan →</p> <p><i>Template 1</i>: 10 00 11 00 10 01</p> <p><i>Template 2</i>: 00 11 00 10 01 10</p> <p>Hasil pergeseran</p> <p><i>Template 1</i>: 01 10 00 11 00 10</p> <p><i>Template 2</i>: 00 11 00 10 01 10</p>	0,333	0,333	0,333
2.	<p><i>Template 18</i>: 01 01 11 11 01 01</p> <p><i>Template 20</i>: 10 11 11 11 01 01</p>	0,25	0,25	0,25
3.	<p>Geser 2-bit kekiri ←</p> <p><i>Template 59</i>: 00 11 00 10 01 10</p> <p><i>Template 60</i>: 11 00 10 01 10 00</p> <p>Hasil pergeseran</p> <p><i>Template 59</i>: 00 11 00 10 01 10</p> <p><i>Template 60</i>: 00 11 00 10 01 10</p>	0,0	0,0	0,0

Tabel 4.1 menggabungkan kedua nilai dari *Hamming Distance* dan WED, maka akan menghasilkan nilai rata-rata. Dari rentang 0 hingga 1, *template iris* yang dibandingkan dapat dikatakan sama akan memiliki nilai dari 0,0 hingga maksimal 0,4. Sedangkan apabila nilai yang dihasilkan dari kedua algoritma tersebut diatas 0,4, maka disimpulkan *template iris* yang dibandingkan tidak cocok.

```

>>> Start verifying /content/Skripsi_Iris/dataset/001_1_3.jpg

>>> 6 samples matched (descending reliability):
    001_1_1.jpg.mat
    001_2_2.jpg.mat
    001_1_2.jpg.mat
    001_2_3.jpg.mat
    001_2_4.jpg.mat
    001_2_1.jpg.mat

>>> Verification time: 3.905132293701172 [s]

```

Gambar 4.23 Hasil Perbandingan *Template* Iris

4.4 Evaluasi

Berikut proses evaluasi menggunakan *k-fold cross validation* untuk menghitung nilai akurasi dan efisiensi waktu sistem *Iris Recognition*. Dengan nilai $k = 7$, maka citra iris dengan jumlah 756 citra akan dibagi menjadi 7 bagian data *training* dan *testing*. Maka setiap k ada 108 citra iris yang diproses.

Tabel 4.2 Hasil Pengujian *Template* Iris

K	TRAIN	WAKTU TOTAL	TOTAL TEMPLATE	WAKTU TIAP TEMPLATE	SKOR	NILAI AKURASI
1	1-108	3,90	6	0,65	100,00	94,90
		3,88	6	0,65	100,00	
		3,74	6	0,62	100,00	
		4,15	6	0,69	100,00	
		3,80	6	0,63	100,00	
		3,65	6	0,61	100,00	
		4,23	5	0,85	83,00	
		4,02	5	0,80	83,00	
		3,88	6	0,65	100,00	
		3,80	5	0,76	83,00	
2	109- 216	4,11	6	0,69	100,00	96,60
		3,76	5	0,75	83,00	

		3,90	6	0,65	100,00	
		4,05	6	0,68	100,00	
		3,92	6	0,65	100,00	
		3,95	6	0,66	100,00	
		4,11	5	0,82	83,00	
		4,02	6	0,67	100,00	
		3,96	6	0,66	100,00	
		4,09	6	0,68	100,00	
3	217-324	3,91	6	0,65	100,00	96,60
		3,67	6	0,61	100,00	
		3,90	6	0,65	100,00	
		3,89	6	0,65	100,00	
		3,86	6	0,64	100,00	
		3,94	5	0,79	83,00	
		3,87	5	0,77	83,00	
		4,32	6	0,72	100,00	
		4,27	6	0,71	100,00	
		4,20	6	0,70	100,00	
4	325-432	4,03	6	0,67	100,00	96,60
		4,04	6	0,67	100,00	
		4,15	6	0,69	100,00	
		4,11	6	0,69	100,00	
		3,97	6	0,66	100,00	
		4,18	6	0,70	100,00	
		4,06	6	0,68	100,00	
		4,22	5	0,84	83,00	
		3,68	5	0,74	83,00	
		3,92	6	0,65	100,00	
5	433-540	3,75	6	0,63	100,00	96,60
		3,78	6	0,63	100,00	
		3,90	6	0,65	100,00	
		3,65	6	0,61	100,00	
		3,62	5	0,72	83,00	

		3,73	6	0,62	100,00	
		3,95	6	0,66	100,00	
		3,72	6	0,62	100,00	
		3,75	5	0,75	83,00	
		3,95	6	0,66	100,00	
6	541-648	3,89	6	0,65	100,00	100,00
		3,83	6	0,64	100,00	
		3,84	6	0,64	100,00	
		3,96	6	0,66	100,00	
		3,97	6	0,66	100,00	
		3,87	6	0,65	100,00	
		3,71	6	0,62	100,00	
		3,59	6	0,60	100,00	
		3,63	6	0,61	100,00	
		3,76	6	0,63	100,00	
7	649-756	3,87	6	0,65	100,00	100,00
		3,59	6	0,60	100,00	
		3,75	6	0,63	100,00	
		3,76	6	0,63	100,00	
		3,75	6	0,63	100,00	
		3,89	6	0,65	100,00	
		3,93	6	0,66	100,00	
		3,82	6	0,64	100,00	
		3,77	6	0,63	100,00	
		3,59	6	0,60	100,00	
TOTAL WAKTU				0,67	TOTAL AKURASI	97,33

Dapat dilihat pada Tabel 4.2, pengujian dilakukan dengan total 70 kali percobaan dengan ukuran citra iris 320x280 piksel pada dataset CASIA V1. Dalam membandingkan *template* citra iris yang telah dibuat pada proses segmentasi, menghasilkan nilai waktu dan akurasi yang berbeda-beda. Hal tersebut dikarenakan

faktor *noise* pada citra iris. Semakin banyak *noise* yang dimiliki pada citra tersebut, maka semakin memperlambat sistem. Selanjutnya, hasil evaluasi yang didapatkan dengan menggunakan *k-fold cross validation*, menghasilkan rata-rata waktu pengenalan iris 0,67 detik, serta nilai akurasi 97,33% dari total percobaan.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan penjelasan sebelumnya, maka dapat disimpulkan dalam sistem *Iris Recognition* dilakukan 3 tahapan, yaitu *preprocessing*, segmentasi, dan perbandingan. Total waktu yang dibutuhkan dari proses segmentasi, normalisasi, dan *encoding* pada dataset CASIA berjumlah 756 citra iris mata yaitu 690,963 detik. Jika waktu dibagi dengan total citra iris mata, maka setiap iris mata membutuhkan waktu 0,913 detik. Proses membandingkan dua *template* iris menggunakan algoritma *Hamming Distance* serta dibantu dengan *Weighted Euclidean Distance* (WED) bertujuan untuk mengukur seberapa berbeda *template* dan untuk memutuskan apakah mereka milik individu yang sama atau tidak. Hasil evaluasi perbandingan iris dengan menggunakan *k-fold cross validation* mendapatkan nilai waktu rata-rata 0,67 detik serta nilai akurasi 97,33% dari total 70 kali percobaan dari jumlah 756 citra iris dataset CASIA V1.

5.2 Saran untuk Penelitian Lanjutan

Dengan hasil yang diperoleh dari penelitian ini, dimana proses segmentasi, normalisasi, dan *encoding* menghabiskan waktu 0,913 detik, serta total waktu 0,67 detik pada tahap perbandingan. Saran terkait penelitian selanjutnya, yakni terkait efisiensi waktu dan nilai akurasi agar semakin ditingkatkan, namun dengan sistem *Iris Recognition* yang lebih ringan. Karena mengingat, semakin lama proses komputasi berlangsung, maka semakin berat *hardware* berkerja. Serta kekuatan *chipset* yang dimiliki oleh *smartphone* yang berbeda dengan *processor* di *desktop*,

maka dibutuhkan penelitian lebih lanjut agar dapat membuat algoritma yang lebih baik untuk meringankan beban kerja tersebut.

DAFTAR PUSTAKA

- Awcock, G. (1996). *Applied Image Processing*. Singapore: McGraw-Hill Book.
- Crammer, K. (2002). *On the algorithmic implementation of multiclass kernel-based vector machines* (Vol. 2). JMLR.
- Criminisi, A., Pérez, P., & Toyama, K. (2004). Region Filling and Object Removal by Exemplar-Based Image Inpainting. In *IEEE TRANSACTIONS ON IMAGE PROCESSING* (Vol. 13, Issue 9). www.csse.monash.edu.au/
- Daugman, J. (2004). How Iris Recognition Works. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1), 21–30. <https://doi.org/10.1109/TCSVT.2003.818350>
- Field, D. (1987). Relations between the Statistics of Natural Images and the Response Properties of Cortical Cells. 4.
- Feng, X., Fang, C., Ding, X., & Wu, Y. (2006). Iris localization with dual coarse-to-fine strategy. *Proceedings - International Conference on Pattern Recognition*, 4, 553–556. <https://doi.org/10.1109/ICPR.2006.725>
- Galdi, C. (2017). FIRE: Fast Iris REcognition on mobile phones by combining colour and texture features. *Pattern Recognition Letters*, 91, 45-51.
- Gan, G., Ma, C., & Wu, J. (2007). *Data Clustering Theory, Algorithms, and Applications*. United States of America: SIAM.
- Gonzalez. (2002). *Digital Image Processing Second Edition*.
- Hasegawa, M., Kako, T., Hirobayashi, S., Misawa, T., Yoshizawa, T., & Inazumi, Y. (2013). Image inpainting on the basis of spectral structure from 2-D nonharmonic analysis. *IEEE Transactions on Image Processing*, 22(8), 3008–3017. <https://doi.org/10.1109/TIP.2013.2253475>
- Juniati, C. K. (2017). *Pengenalan Iris Mata Menggunakan Ekstraksi Fitur Dimensi Fraktal Box Counting*. Surabaya: Math Jurnal Ilmiah Matematika UNESA.
- M. Quraish Shihab. (2002). *TAFSIR AL-MISBAH VOLUME 7 surah ibrahim - al-isra*. 7(Surah Ibrahim-Surah Al-Isra'), 568.
- Masek, L. (2003). *Recognition of Human Iris Patterns for Biometric Identification*.
- Michailovich, O., Rathi, Y., & Tannenbaum, A. (2007). Image segmentation using active contours driven by the Bhattacharyya gradient flow. *IEEE Transactions on Image Processing*, 16(11), 2787–2801. <https://doi.org/10.1109/TIP.2007.908073>
- Murti, D. (2005). *Clustering Data Non-Numerik Dengan Pendekatan Algoritma K-Means Dan Hamming Distance Studi Kasus Biro Jodoh*. Surabaya: Institut Teknologi Sepuluh November.

- Neha Kak, R. G. (2010). Iris Recognition System. *International Journal of Advanced Computer Science and Applications*, 1, 34-40.
- Nowak, R. D., & Baraniuk, R. G. (n.d.). *ADAPTIVE WEIGHTED HIGHPASS FILTERS USING MULTISCALE ANALYSIS*.
- Park, S. W., & Savvides, M. (2015). Iris Segmentation Using Active Contours. *Encyclopedia of Biometrics*, 4(4), 1037–1042. https://doi.org/10.1007/978-1-4899-7488-4_254
- Paulín-Martínez, F. J., Lara-Guevara, A., Romero-González, R. M., & Jiménez-Hernández, H. (2019). Implementation of the Hough Transform for Iris Detection and Segmentation. *Advances in Molecular Imaging*, 09(01), 6–18. <https://doi.org/10.4236/ami.2019.91002>
- Pundlik, S. J., Woodard, D. L., & Birchfield, S. T. (2008). Non-ideal iris segmentation using graph cuts. *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, June*. <https://doi.org/10.1109/CVPRW.2008.4563108>
- Putra, D. (2010). *Pengolahan Citra Digital (Edisi 1)*. Yogyakarta: Andi.
- RI., K. A. (2017). *Al-Qur'an dan Terjemahan: Juz 1-30*. Kemenag RI.
- Shafique. (2017). Modern Authentication Techniques in Smart Phones: Security and Usability Perspective. *International Journal of Advanced Computer Science and Applications*, 8(1), 2.
- Suhandy, A. (2007). Pengembangan Algoritma Image Processing Untuk Menduga Kemasakan Buah Manggis Segar.
- Sutoyo, T. (2009). *Teori Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi Offset.
- Ukpai, C. O. (n.d.). *Biometric Iris Image Segmentation and Feature Extraction for Iris Recognition By. March 2015*.
- Viriyavisuthisakul, S. S. (2015). A comparison of similarity measures for online social media Thai text classification. *International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 1-6.
- Wardani, A. S. (2020, May 1). *Apple Bakal Permudah Pengguna Buka Kunci iPhone meski Pakai Masker*. From Liputan6.com: (<https://www.liputan6.com/tekn/read/4241788/apple-bakal-permudah-pengguna-buka-kunci-iphone-meski-pakai-masker>) diakses 26 November 2021

LAMPIRAN

Lampiran 1 : Proses Segmentasi Citra Iris Mata

```
import argparse, os
from glob import glob
from tqdm import tqdm
from time import time
from scipy.io import savemat
from multiprocessing import cpu_count, Pool
from fnc.extractFeature import extractFeature

parser = argparse.ArgumentParser()
parser.add_argument("--data_dir", type=str, default="../CASIA1/",
                    help="Path to the directory containing CASIA1 images.")
parser.add_argument("--
temp_dir", type=str, default="./templates/CASIA1/",
                    help="Path to the directory containing templates.")
parser.add_argument("--n_cores", type=int, default=cpu_count(),
                    help="Number of cores used for enrolling template.")
args = parser.parse_args()

def pool_func(file):
    template, mask, _ = extractFeature(file, use_multiprocess=False)
    basename = os.path.basename(file)
    out_file = os.path.join(args.temp_dir, "%s.mat" % (basename))
    savemat(out_file, mdict={'template': template, 'mask': mask})

start = time()
if not os.path.exists(args.temp_dir):
    print("makedirs", args.temp_dir)
    os.makedirs(args.temp_dir)
files = glob(os.path.join(args.data_dir, "*_*_.jpg"))
n_files = len(files)
print("Number of files for enrolling:", n_files)
print("Start enrolling...")
pools = Pool(processes=args.n_cores)
for _ in tqdm(pools.imap_unordered(pool_func, files), total=n_files):
    pass

end = time()
print('\n>>> Enrollment time: {} [s]\n'.format(end-start))
```

Lampiran 2 : Proses Perbandingan Antar *Template* Iris

```
import argparse

from time import time

from fnc.extractFeature import extractFeature
from fnc.matching import matching
parser = argparse.ArgumentParser()

parser.add_argument("--file", type=str,
                    help="Path to the file that you want to verify.")

parser.add_argument("--temp_dir", type=str, default="./templates/",
                    help="Path to the directory containing templates.")

parser.add_argument("--thres", type=float, default=0.38,
                    help="Threshold for matching.")

args = parser.parse_args()
start = time()
print('>>> Start verifying {}\n'.format(args.file))
template, mask, file = extractFeature(args.file)

result = matching(template, mask, args.temp_dir, args.thres)

if result == -1:
    print('>>> No registered sample.')

elif result == 0:
    print('>>> No sample matched.')

else:
    print('>>> {} samples matched (descending reliability):'.format(len(
    result)))
    for res in result:
        print("\t", res)

end = time()
print('\n>>> Verification time: {} [s]\n'.format(end - start))
```

RIWAYAT HIDUP



Ahmad Zidan Nur Hakim lahir di Kabupaten Banyuwangi pada 6 Mei 2000. Memiliki nama panggilan Zidan. Bertempat tinggal di Jalan Gambor Dusun Gebang RT/RW 03/02 Desa Benelan Kidul, Kecamatan Singojuruh, Kabupaten Banyuwangi. Merupakan anak pertama dari Bapak Bambang Irawan, M.Pd dan Ibu Nur Holisah Hamim, S.Pt.

Pendidikan yang pernah ditempuh yaitu TK Harapan II Gladag, kemudian melanjutkan sekolah di SD Negeri 1 Gladag dan lulus pada tahun 2012. Menempuh Pendidikan SMP di Sekolah Menengah Pertama Negeri 3 Rogojampi lulus pada tahun 2015. Melanjutkan Pendidikan SMA di Sekolah Menengah Atas Negeri 1 Glagah lulus pada tahun 2018.

Tahun 2018 melanjutkan studi ke jenjang Pendidikan Strata 1 di Universitas Islam Negeri Maulana Malik Ibrahim Malang mengambil Program Studi Matematika dengan lintas minat Komputasi, Fakultas Sains dan Teknologi. Aktif mengikuti kegiatan organisasi intra kampus HMJ Integral Matematika UIN Malang (2018-2021), serta menjadi Tim IT (iTeam) Pusat Ma'had Al-Jami'ah UIN Malang (2019-2021).

Kegiatan-kegiatan yang pernah diikuti yaitu Tim IT Webinar Nasional Kompetisi Matematika (2020-2021), Tim *Livestreaming* Fakultas Sains dan

Teknologi UIN Malang tahun 2021, Kuliah Kerja Mahasiswa (KKM) UIN Malang mengabdikan tahun 2021, Praktek Kerja Lapangan (PKL) di Dinas Komunikasi dan Informatika Kota Malang di bidang Aptika tahun 2021, serta sebagai Tim Media Lembaga Penelitian dan Pengembangan kepada Masyarakat (LP2M) UIN Malang.



KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI

Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

BUKTI KONSULTASI SKRIPSI

Nama : Ahmad Zidan Nur Hakim
NIM : 18610070
Fakultas / Program Studi : Sains dan Teknologi / Matematika
Judul Skripsi : Implementasi Metode FIRE dan *Image Processing* Citra Mata untuk Mendeteksi Pola Iris pada Proses Autentikasi *Smartphone*
Pembimbing I : Hisyam Fahmi, M.Kom
Pembimbing II : Erna Herawati, M.Pd

No	Tanggal	Hal	Tanda Tangan
1.	15 Desember 2021	Konsultasi Bab 1	1.
2.	11 Januari 2022	Bimbingan Bab 1	2.
3.	13 Januari 2022	Bimbingan Kajian Agama	3.
4.	20 Januari 2022	Bimbingan Bab 2	4.
5.	1 Februari 2022	Bimbingan Kajian Agama	5.
6.	7 Februari 2022	Bimbingan Bab 2	6.
7.	10 Februari 2022	Bimbingan Bab 2	7.
8.	23 Februari 2022	Bimbingan Bab 3	8.
9.	25 Februari 2022	Bimbingan Bab 3 (Revisi)	9.
10.	31 Maret 2022	Acc Bab 1,2,3	10.
11.	14 April 2022	Bimbingan Kajian Agama	11.
12.	14 April 2022	Acc Seminar Proposal	12.
13.	29 April 2022	Bimbingan Bab 4	13.
14.	31 Mei 2022	Bimbingan Bab 4	14.
15.	6 Juni 2022	Acc Seminar Hasil	15.
16.	14 Juni 2022	Bimbingan Bab 4,5	16.
17.	16 Juni 2022	Bimbingan Kajian Agama	17.
18.	16 Juni 2022	Acc Sidang Skripsi	18.

Malang, 24 Juni 2022

Mengetahui,

Ketua Program Studi Matematika

Dr. Elly Susanti, M.Sc
NIP.197411292000122005