

IDENTIFIKASI POLA IRIS MATA DENGAN ALGORITME *DAUGMAN* DAN METODE *HAMMING DISTANCE*

Alfio Rezika¹, Ernawati², Aan Erlansari³

^{1,2,3}Program Studi Informatika, Fakultas Teknik, Universitas Bengkulu
^{1,2,3} Jl. W.R. Supratman Kandang Limun Bengkulu 38371A Indonesia
(telp: 0736-341022 ; fax: 0736-341022)

¹alfio.rezika@gmail.com

²ernawati@unib.ac.id

³aan_erlanshari@unib.ac.id

Abstrak: Penelitian ini membangun sebuah sistem identifikasi pola iris mata berbasis tekstur dengan algoritme *Daugman* untuk deteksi letak iris dan metode *Hamming Distance* untuk menghitung nilai kemiripan citra uji dengan citra latih. Citra yang digunakan sebagai objek penelitian adalah citra iris mata manusia dari *CASIA Iris Image Database*. Aplikasi ini dibangun dalam bahasa pemrograman Matlab dan dirancang dengan *Unified Model Language* (UML). Metode pengembangan sistem yang digunakan adalah *protoyping*. Hasil yang diperoleh pada penelitian ini adalah nilai akurasi sebesar 82.5% untuk pengujian citra uji terhadap citra latih.

Kata Kunci: biometrik, citra iris mata, *Daugman*, *Hamming distance*.

Abstract: This research build an iris identification system using Daugman algorithm on iris detection and Hamming Distance method to calculate the similarity value of test images over trained images. Iris images used in this research are provided by CASIA Iris Image Database. This application is built in Matlab programming language and designed with Unified Model Language (UML). System development method that used in this research is protoyping. The results obtained in this research is the identification accuracy value or Genuine Acceptance Rate (GAR) of 82.5% for testing the test images over the trained images.

Keywords: biometric, iris image, Daugman, Hamming Distance.

I. PENDAHULUAN

Iris atau selaput pelangi adalah bagian dari mata yang melingkari lingkaran pupil. Iris

berfungsi untuk memberi warna pada mata. Iris adalah bagian tubuh manusia yang dapat digunakan sebagai pengenalan seseorang dengan tingkat akurasi yang tinggi. Iris bersifat unik, karena pada satu individu punya tekstur iris mata berbeda antara iris sebelah kanan dan sebelah kirinya. Bahkan pada dua individu kembar identik pun, pola iris juga berbeda. Atas dasar inilah iris mata dapat dijadikan dasar bagi pengenalan biometrik.

Iris mata dinilai sebagai teknologi biometrik yang paling aman dan paling akurat. Iris mata tidak mengalami perubahan seperti halnya yang dapat terjadi pada wajah. Bila suatu individu mengalami kenaikan atau penurunan berat badan, maka akan berdampak pula pada bentuk wajahnya. Dimana akan berpengaruh pada akurasi saat proses pengenalan biometrik dilakukan.

Metode identifikasi pola iris mata yang diangkat dalam penelitian ini adalah algoritme *Daugman* dan *Daugman Rubber Sheet* pada proses deteksi dan normalisasi iris. Sedangkan untuk proses pembuatan template citra iris mata menggunakan metode *Connected Component Labeling*. Serta pada proses pencocokan atau identifikasi menggunakan metode *Hamming Distance* yang membandingkan kedua template citra iris mata serta menghitung jumlah *pixel* yang berbeda pada kedua template tersebut.

II. LANDASAN TEORI

A. Pengenalan Iris Mata

Pengenalan iris adalah proses mengenali seseorang dengan menganalisis pola acak dari iris. Iris adalah sebuah otot pada mata yang mengatur ukuran pupil, mengendalikan jumlah cahaya yang masuk ke mata. Ini adalah bagian berwarna dari mata, dan pewarnaan ini didasarkan pada jumlah melatonin pigmen dalam otot. Meskipun warna dan struktur iris terkait genetik, rincian polanya tidak. Iris berkembang selama pertumbuhan prenatal melalui proses yang ketat membentuk pelipatan membran jaringan [1].

Sebelum seorang individu lahir, terjadi degenerasi sehingga menyebabkan pembukaan pupil dan iris yang membentuk acak pola unik. Meskipun identik secara genetik, iris individu yang unik dan struktural dan berbeda memungkinkan untuk digunakan dalam pendeteksian.

B. Algoritme Daugman

Algoritme *Daugman* merupakan operator *integro differential* yang mencari lingkaran pupil dan lingkaran iris dari citra iris mata. Operator *integro differential* merupakan sebuah operator yang mengandung fungsi integral dan turunan sekaligus. Operator *Integro Differential* yang

digunakan oleh *Daugman* untuk menemukan wilayah iris adalah sebagai berikut [1].

$$\max_{(r, x_0, y_0)} \left| G_{\sigma}(r) * \frac{\partial}{\partial r} \oint_{r, x_0, y_0} \frac{I(x, y)}{2\pi r} ds \right| \quad (1)$$

Dimana:

$I(x, y)$: citra mata

r : radius

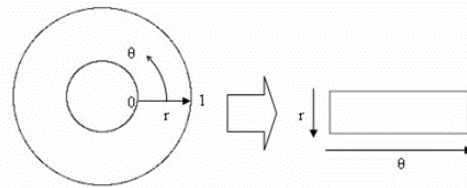
Ds : busur

(x_0, y_0) : koordinat pusat

$*$: lilitan

$G_{\sigma}(r)$: fungsi *smoothing Gaussian filter*

Setelah iris terdeteksi, proses selanjutnya adalah normalisasi dengan menggunakan metode *Daugman Rubber Sheet* dengan mengubah wilayah iris yang berbentuk gelang kedalam bentuk persegi panjang. Metode ini memetakan ulang setiap poin pada wilayah iris dari koordinat kartesius (x, y) menjadi koordinat kutub (r, θ) dimana r adalah interval $[0, 1]$ dan θ adalah sudut $[0, 2\pi]$.



Gambar 1. *Daugman Rubber Sheet Model*

Proses pemetaan ulang wilayah iris dari koordinat kartesius (x, y) menjadi koordinat kutub (r, θ) dirumuskan sebagai berikut [1].

$$I(x(r, \theta), y(r, \theta)) \Rightarrow I(r, \theta) \quad (2)$$

$$x(r, \theta) = (1-r) x_p(\theta) + r x_l(\theta) \quad (3)$$

$$y(r, \theta) = (1-r) y_p(\theta) + r y_l(\theta) \quad (4)$$

Keterangan:

$I(x, y)$: wilayah iris yang terdeteksi

(x, y) : koordinat kartesius

(r, θ) : koordinat kutub yang telah dinormalisasi

x_p, y_p : koordinat lingkaran pupil searah dengan θ

x_l, y_l : koordinat lingkaran iris searah dengan θ

C. Connected Component Labeling

Connected Component Labeling melakukan proses pemindaian dan pelabelan piksel pada citra biner atau *grayscale*. Setiap piksel pada citra biner yang bernilai 1 (*foreground*) dan saling terhubung akan diberikan nomor label yang sama, sedangkan piksel yang bernilai 0 (*background*) tidak diberi label atau diberi label 0 [2].

Connected Component Labeling bekerja dengan memindai citra, piksel demi piksel dari atas ke bawah dan dari kiri ke kanan untuk mengidentifikasi daerah piksel yang terhubung, yaitu daerah dengan piksel berdekatan yang memiliki nilai intensitas yang sama, yang sering diasumsikan sebagai V . Pada sebuah citra biner $V = \{1\}$; namun, dalam citra graylevel V akan memiliki berbagai nilai, misalnya: $V = \{51, 52, 53, \dots, 77, 78, 79, 80\}$ [2]. Piksel disebut *connected* bila mematuhi aturan *adjacency* atau aturan “kedekatan” piksel. Aturan kedekatan piksel ini memanfaatkan sifat ketetanggaan piksel. Dengan demikian piksel-piksel yang di katakan *connected* pada dasarnya memiliki sifat *adjacency* satu sama lain karena mereka masih memiliki hubungan *neighbourhood* atau ketetanggaan [2].

Terdapat dua jenis konektivitas yang digunakan yaitu 4-konektivitas (*4-Connected Neighbors*) dan 8-konektivitas (*8-Connected Neighbors*). Piksel-piksel yang berdekatan dikatakan memiliki hubungan 4-konektivitas jika piksel-piksel tersebut terletak berdampingan secara horizontal dan vertikal $N_4(P)$. Kumpulan dari piksel-piksel ini disebut dengan *4 neighbors of P* dirumuskan sebagai berikut.

$$N_4(p) = \{(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)\} \quad (5)$$

Keterangan:

p : piksel

(x, y) : koordinat piksel

	P(x, y-1)	
P(x-1, y)	P(x, y)	P(x+1, y)
	P(x, y+1)	

Piksel yang berdekatan dikatakan memiliki hubungan 8-konektivitas jika piksel tersebut terletak berdampingan secara horizontal dan vertikal $N_8(P)$ dirumuskan sebagai berikut [2].

$$N_8(p) = N_4 \cup \{(x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1)\} \quad (6)$$

Dimana:

p : piksel

(x, y) : koordinat piksel

P(x-1, y-1)	P(x, y-1)	P(x+1, y-1)
P(x-1, y)	P(x, y)	P(x+1, y)
P(x-1, y+1)	P(x, y+1)	P(x+1, y+1)

Dalam proses pelabelan piksel, digunakan pendekatan *4-connected neighbors* untuk menghubungkan antara piksel yang memiliki nilai yang sama dan menggunakan teknik *flood fill* untuk mengisi warna acak pada piksel terpilih ke sebuah matriks penampung yang biasa disebut *matrix mapping* [2].

D. Hamming Distance

Algoritme *Hamming Distance* merupakan salah satu dari algoritme *approximate string matching* yang ditemukan oleh Richard Hamming pada tahun 1950. Pada penelitian ini, proses pencocokan menggunakan *Hamming distance* untuk memutuskan apakah gambar (*template* yang terbentuk) adalah iris mata yang sama atau tidak. *Hamming distance* menggunakan operator *Boolean Exclusive - OR (XOR)* untuk membandingkan dua

pola iris. Operator *XOR* mendeteksi perbedaan antara setiap pasang bit. Berikut ini adalah formula dari [3] *Hamming distance* :

$$HD = \frac{1}{N} \sum_{j=1}^N X_j \oplus Y_j \quad (7)$$

Keterangan:

- HD* : jumlah bit yang berbeda dari kedua *string*
N : jumlah biner yang ditampilkan kedua *string*
X_j, Y_j : representasi untuk dibandingkan
 \oplus : *Exclusive-OR (XOR)*

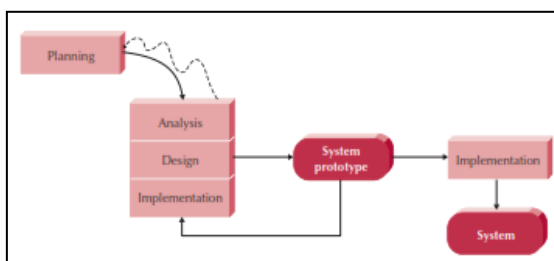
III. METODOLOGI

A. Data Set Penelitian

Data set penelitian yang digunakan dalam penelitian ini sebanyak 120 citra iris mata dari 20 iris mata berbeda. Masing-masing iris memiliki 6 citra, 4 diantaranya menjadi citra latih dan 2 lainnya menjadi citra uji yang diambil dari *CASIA Iris Database* (sumber: <http://biometrics.idealtest.org/>).

B. Metode Pengembangan Sistem

Metode Pengembangan sistem yang digunakan pada penelitian ini adalah metode *prototyping*. Metodologi *prototyping* melakukan tahap analisis, perancangan dan implementasi secara bersamaan, dan ketiga tahap tersebut dilakukan secara berulang dalam siklus hingga sistem selesai atau lengkap [4].



Gambar 2. Diagram Alir *Prototyping* [4]

C. Metode Pengujian Sistem

Pengujian sistem yang dilakukan pada penelitian ini dilakukan dengan *black box testing*. Pada *black box testing* akan digunakan metode pengujian partisi ekuivalensi, yaitu salah satu metode pengujian *black box* yang membagi domain input dari suatu program ke dalam kelas data dari mana *test case* dapat dilakukan. *Test case* yang ideal mengungkap kelas kesalahan yang akan memerlukan banyak kasus untuk dieksekusi sebelum kesalahan umum diamati [5].

D. Metode Uji Kelayakan Sistem

Uji kelayakan Dalam penelitian ini, uji kelayakan sistem aplikasi dilakukan dengan cara mengukur keefektifan aplikasi dalam mengenali kecocokan citra iris mata. Untuk menentukan keefektifan aplikasi dalam mengenali kecocokan citra iris mata, maka dilakukan sebuah evaluasi yang disebut evaluasi biometrika [6].

Evaluasi ini dilakukan dengan melakukan pencocokan citra uji dengan citra latih pada *database*. Pengukuran keberhasilan pencocokan dari sistem biometrika adalah dengan menguji seberapa besar respon sistem dalam menerima atau menolak. Tingkat kesalahan sistem dalam sistem biometrika dihitung berdasarkan kesalahan penerimaan dan kesalahan sebagai berikut:

- 1) *False Rejection Rate (FRR)* atau *False Positive / False Match / False Alarm Rate* merupakan persentase dari pengguna yang sah (asli), namun dikenali oleh sistem sebagai penyusup (*imposter*).
- 2) *False Acceptance Rate (FAR)* atau *False Negative / False Nomatch*. Merupakan persentase dari pengguna yang tidak sah (penyusup), namun dikenali oleh sistem sebagai pengguna yang sah (asli).

Unjuk kerja suatu sistem biometrika dinyatakan dengan *Decision Error Rate* (rasio kesalahan), yaitu *False Matching Rate/FMR* (Rasio Kesalahan Pencocokan) dan *False Non Matching Rate/FNMR* (Rasio Kesalahan Ketidakcocokan). Kesalahan dihitung dengan menggunakan persamaan, berikut:

$$FMR = \frac{\text{Jumlah Kesalahan}}{\text{Jumlah Keseluruhan Proses}} \times 100\% \quad (8)$$

Untuk menyatakan tingkat kesuksesan pengenalan suatu sistem biometrika (bukan tingkat kesalahan) disebut sebagai *Genuine Acceptance Rate* (GAR) dan dihitung menggunakan persamaan berikut :

$$GAR = 1 - FMR \text{ atau } GAR = 1 - FNMR \quad (9)$$

Kinerja biometrika ditentukan berdasarkan kedua parameter tersebut. Jika tingkat kesalahannya tinggi, maka sistem biometrika harus ditinjau apakah memang ada kesalahan algoritme atau kesalahan pembacaan sensor atau memang tipe biometrika itu sendiri tidak bisa memenuhi kekhasan. Umumnya semakin besar ukuran *database* atau jumlah pengguna dalam *database*, maka bagian tumpang tindih akan semakin besar yang artinya FMR/FNMR meningkat dan demikian pula sebaliknya.

IV. ANALISIS DAN PERANCANGAN

A. Analisis Masalah

Sebelumnya telah ada sistem identifikasi iris mata, namun menggunakan metode ekstraksi yang berbeda-beda seperti Transformasi *Wavelet* yang memiliki akurasi pengenalan iris mata mencapai 97,33%, *ICA* (*Independent Component Analysis*) dengan akurasi pengenalan sebesar 97,5 %, dan Jaringan Syaraf Tiruan. Dari beberapa metode tersebut masih bisa digabungkan dengan metode lain yang bertujuan untuk meningkatkan akurasi dalam identifikasi iris mata. Metode lain yang

dapat ditambahkan tersebut misalnya metode pada proses *segmentasi* terhadap objek yang akan dideteksi. Untuk itu pada penelitian ini dilakukan penambahan metode *segmentasi* memanfaatkan algoritme *Daugman*. Dengan ditambahkan metode tersebut, diharapkan akurasi dari metode *Hamming Distance* dalam mengidentifikasi iris mata dapat meningkat dari akurasi metode lain yang diterapkan pada penelitian atau sistem identifikasi iris mata sebelumnya.

B. Pemahaman Kerja Sistem

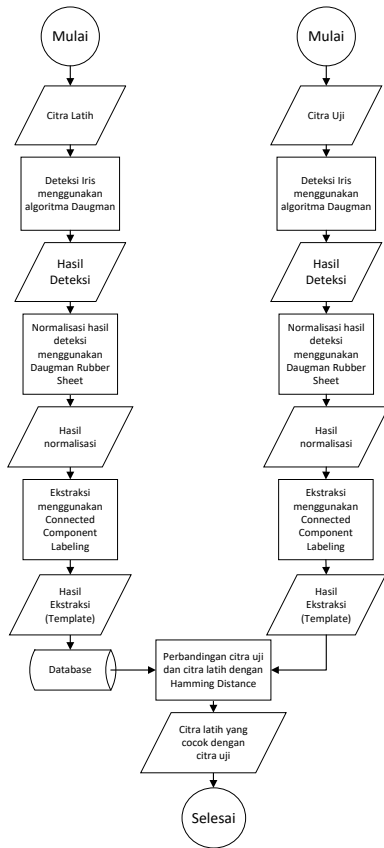
Alir sistem merupakan hasil analisis perancangan tahapan kerja sistem yang akan dibangun. Alir ini dimulai dari *user* memasukkan (*input*) data sampai menghasilkan keluaran (*output*). Dalam sistem ini, *input*-nya berupa citra iris mata manusia dan informasi citra tersebut. Sedangkan untuk *output*-nya adalah citra latih mata yang memiliki nilai kemiripan terdekat dengan citra uji seperti pada Gambar 3.

C. Analisis Fungsional

Analisis fungsional berarti melakukan analisis dari fungsi-fungsi pada sistem yang akan dibangun. Fungsi-fungsi pada sistem yang dimaksud adalah fitur-fitur yang terdapat didalam sistem. Berikut adalah beberapa fitur yang terdapat pada sistem:

- Sistem yang dibangun mampu melakukan segmentasi dengan algoritme *Daugman* dari citra yang diinputkan oleh *user*.
- Sistem yang dibangun dapat melakukan pencocokan citra iris mata dengan menggunakan metode ekstraksi *Connected Component Labeling* dan menggunakan metode *Hamming Distance* untuk mengukur peluang kemiripan antara citra uji dengan citra latih.

Sistem yang dibangun dapat menampilkan *output* berupa identitas citra. Identitas citra tersebut sebelumnya telah disimpan kedalam *database* oleh *user*.



Gambar 3. Diagram Alir Sistem

V. PEMBAHASAN

A. Implementasi Antarmuka

1. Menu Utama

Halaman utama (Gambar 5) yang pertama kali diakses pada saat menjalankan sistem. Pada saat *user* membuka aplikasi maka akan muncul menu utama seperti pada Gambar 5.1. Pada menu utama ini terdapat judul dari sistem yang telah dibuat serta berbagai tombol untuk mengakses menu lainnya yaitu tombol input citra latih, tabel citra latih, pengujian, bantuan dan keluar.

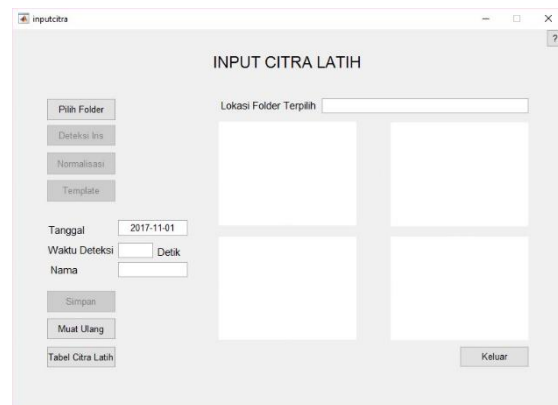
2. Menu Input Citra Latih

Menu input citra latih merupakan tampilan yang akan digunakan untuk melakukan proses

input citra latih. Citra latih ini nantinya akan digunakan dalam proses pengujian dengan citra uji. Berikut adalah tampilan menu input citra latih seperti yang terlihat pada Gambar 6.



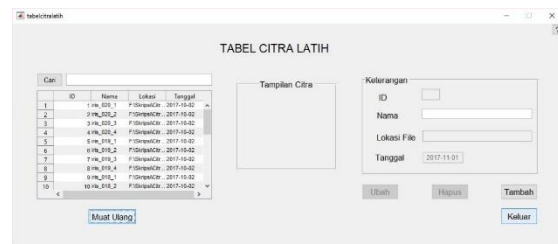
Gambar 5. Menu Utama



Gambar 6. Menu Input Citra Latih

3. Tabel Citra Latih

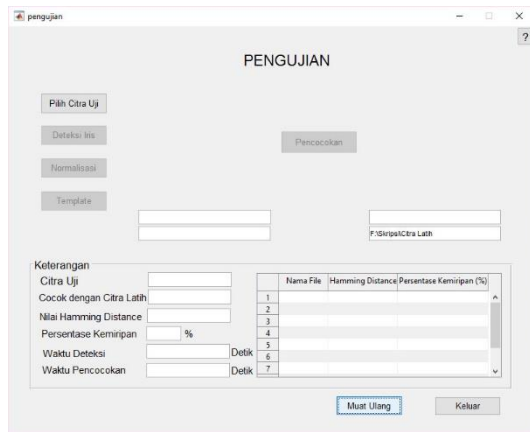
Menu tabel citra latih berfungsi untuk mengelola citra latih yang telah disimpan. Pada menu ini *user* dapat mengubah nama serta menghapus citra latih. Tampilan antarmuka menu tabel citra latih dapat dilihat pada Gambar 7.



Gambar 7. Menu Tabel Citra Latih

4. Pengujian

Menu pengujian dapat diakses pada menu utama. Menu ini berfungsi untuk melakukan pencocokan antara citra uji dengan citra latih. Berikut adalah tampilan dari menu pengujian seperti yang terlihat pada Gambar 8.



Gambar 8. Menu Pengujian

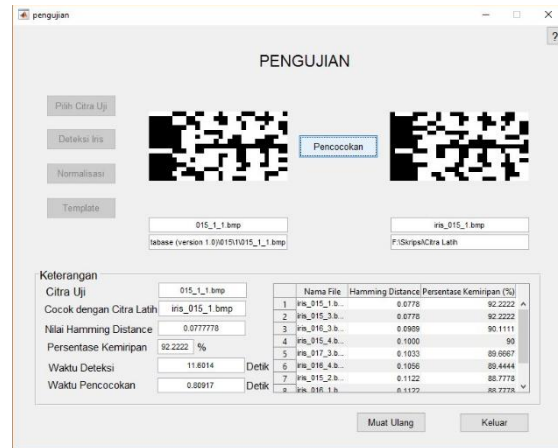
B. Uji Kelayakan Sistem

1. Pengujian Citra Uji Terhadap Citra Latih

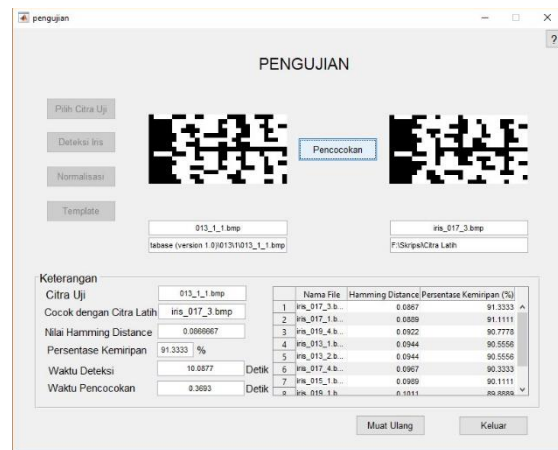
Pengujian yang pertama adalah pengujian terhadap algoritme *Daugman* dan metode *Hamming Distance* dengan menggunakan 120 citra, terdiri dari 80 citra latih dan 40 citra uji. Pengujian ini ditujukan untuk melakukan proses pencocokan citra iris mata yang telah dipilih sebelumnya pada menu input citra latih yang kemudian diinputkan kedalam sistem. Inputan yang diuji merupakan citra uji dengan berbagai variasi keadaan. Gambar 9 berikut merupakan hasil dari citra uji yang berhasil dikenali dengan tepat terhadap citra latihnya.

Dari Gambar 9 dapat dilihat bahwa citra uji yang berhasil dikenali dengan tepat, ditandai dengan munculnya citra latih dan nama citra yang sama dengan citra uji. Selain itu sistem juga menampilkan identitas dari citra iris mata sebagai informasi mengenai nama file iris mata. Terdapat

tabel yang menampilkan 10 data citra latih yang memiliki nilai kemiripan terdekat. Sedangkan untuk citra yang gagal dikenali dengan tepat, dapat dilihat pada Gambar 10.



Gambar 9. Citra Uji Berhasil Dikenali

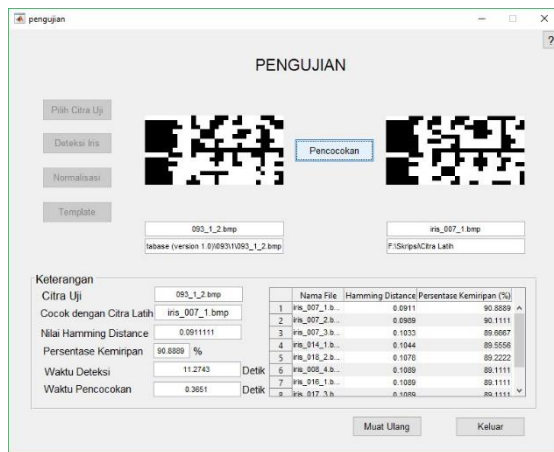


Gambar 10. Citra Uji Gagal Dikenali

Dari Gambar 10 dapat dilihat bahwa citra uji yang dipilih gagal dikenali dengan tepat oleh sistem. Hal tersebut ditandai dengan ditampilkannya nama citra latih 017 yang tidak tepat dengan citra uji 013. Identitas citra tetap ditampilkan karena hal tersebut merupakan informasi dari citra latih yang ditampilkan. dari 40 citra yang diuji pada sistem, banyaknya citra yang berhasil dikenali dengan tepat oleh sistem sebanyak 33 citra, sedangkan citra yang gagal dikenali dengan tepat oleh sistem sebanyak 7 citra.

2. Pengujian Citra Uji yang Tidak Ada pada Folder Citra Latih

Pada pengujian ini, citra uji yang digunakan adalah citra yang tidak memiliki citra latih pada folder citra latih. Citra tersebut juga diproses sama seperti citra uji yang lainnya kemudian dilakukan proses pencocokan dengan seluruh citra latih yang ada folder citra latih. Ketika pengujian, didapatkan hasil bahwa sistem mengenali citra uji tersebut sebagai salah satu citra yang berada di dalam citra latih, walaupun di dalam citra latih tidak terdapat citra tersebut. Pengujian tersebut dapat dilihat pada Gambar 11.



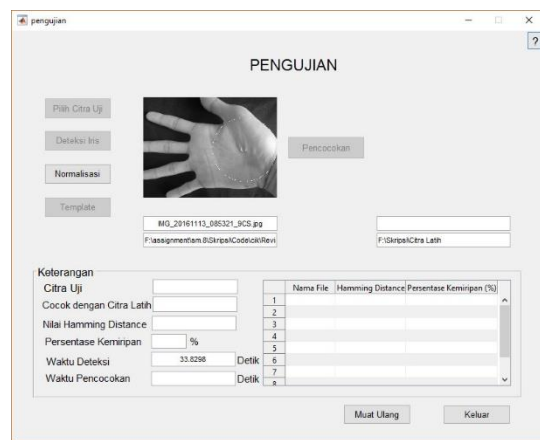
Gambar 11. Hasil Pengujian Citra Uji yang Tidak Ada pada Folder Citra Latih

Dari Gambar 11 dapat dijelaskan bahwa pengujian ini membuktikan sistem masih mampu mengenali citra uji dimana citra tersebut tidak terdapat didalam citra latih karena sistem berusaha menemukan citra latih dengan nilai kemiripan terdekat dengan citra uji yang diukur dengan metode *Hamming distance*.

Ketika citra uji diukur nilai kemiripannya dengan citra latih menggunakan metode *Hamming distance*, nilai kemiripan terdekatnya didapat dengan salah satu citra latih. Nilai kemiripan terdekat yang diperoleh dari hasil pengujian ini adalah sebesar 0.091111.

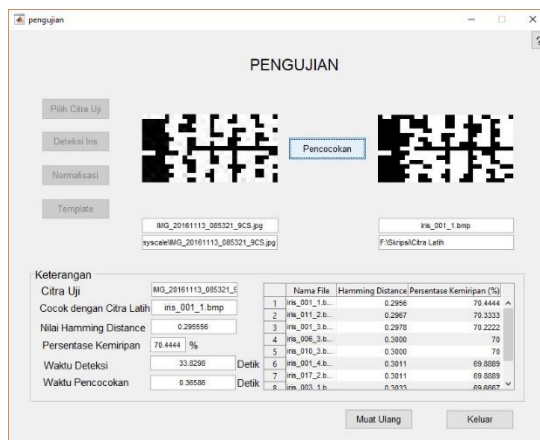
3. Pengujian Citra Bukan Iris Mata Sebagai Citra Uji

Pada pengujian ini, citra uji yang digunakan adalah telapak tangan, dimana tidak terdapat unsur iris mata sama sekali. Ketika pengujian, didapat hasil bahwa sistem tetap mampu melakukan proses deteksi, normalisasi, template, serta pencocokan. Namun pada proses deteksi, hasil deteksi berupa bagian wilayah telapak tangan seperti yang terlihat pada Gambar 12.



Gambar 12. Kesalahan Deteksi

Hal ini disebabkan karena tidak ada objek iris mata pada citra uji sehingga sistem hanya menggambarkan lingkaran yang normalnya untuk menandai wilayah iris ke wilayah yang memiliki nilai gradien lingkaran maksimum pada citra telapak tangan tersebut.



Gambar 13. Pengujian Citra Bukan Iris Mata

Proses pencocokan masih dapat berlangsung karena ketika citra uji diukur nilai kemiripannya dengan citra latih menggunakan metode *Hamming Distance*, nilai kemiripan terdekatnya didapat dengan salah satu citra latih, sehingga sistem akan menganggap bahwa citra uji tersebut memiliki kemiripan terdekat dengan citra latih yang memiliki nilai kemiripan terdekat tersebut. Nilai kemiripan terdekat yang diperoleh dari hasil pengujian ini adalah sebesar 0.295556 seperti yang terlihat pada Gambar 13.

VI. KESIMPULAN

Berdasarkan penelitian dan pembahasan hasil yang telah dilakukan, maka terdapat beberapa kesimpulan yang diperoleh dari penelitian ini, diantaranya sebagai berikut:

1. Penelitian ini telah menghasilkan sebuah aplikasi identifikasi pola iris mata dengan algoritme *Daugman* dan metode *Hamming distance*.
2. Identifikasi pola iris mata dengan algoritme *Daugman* dan metode *Hamming distance* memiliki tingkat kesuksesan pengenalan atau *Genuine Acceptance Rate (GAR)* sebesar 82.5%.
3. Faktor yang menyebabkan adanya kesalahan identifikasi iris mata adalah terdapat citra latih lain yang memiliki nilai kemiripan yang lebih dekat dengan citra uji.

VII. SARAN

Berdasarkan analisis yang sudah dilakukan pada penelitian ini, terdapat beberapa saran yang dapat diberikan dalam pengembangan penelitian ini kedepannya. Berikut saran yang dapat diberikan:

1. Pengembangan sistem untuk identifikasi secara *real time* dengan mengidentifikasi iris yang diambil langsung dari perangkat kamera oleh sistem.
2. Penambahan fungsi *image enhancement* yang bisa meningkatkan akurasi pencocokan iris.

REFERENSI

- [1] J. Daugman, "How Iris Recognition Works," IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, vol. 14, 2004.
- [2] R. Fisher, S. Perkins, A. Walker and E. Wolfart, "Connected Component Labeling," 2003.
- [3] A. S. Anwar, An Iris Detection and Recognition System to Measure The Performance of E-Security, Dhaka: Brac University, 2016.
- [4] A. Dennis, B. H. Wixom and D. Tegarden, Sistem Analysis and Design with UML Version 2.0, United States of America: John Wiley & Sons, Inc., 2005.
- [5] D. Sundar, Software Engineering, New Delhi: University Science Press, 2010.
- [6] H. Ramza and Y. Dewanto, Teknik Pemrograman Menggunakan Matlab, Jakarta: Grasindo, 2007.