

**PENGOLAHAN CITRA UNTUK PENGENALAN WAJAH MANUSIA  
MENGUNAKAN *PRINCIPAL COMPONENT ANALYSIS* DAN  
*EUCLIDEAN DISTANCE***

**TUGAS AKHIR SKRIPSI**

Diajukan Kepada Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Negeri Yogyakarta

Untuk Memenuhi Sebagian Persyaratan Guna Memperoleh Gelar Sarjana Sains



Oleh:

Salamah Nur Aqidah

NIM 15306141049

**PROGRAM STUDI FISIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS NEGERI YOGYAKARTA**

**2019**

# PENGOLAHAN CITRA UNTUK PENGENALAN WAJAH MANUSIA MENGUNAKAN *PRINCIPAL COMPONENT ANALYSIS* DAN *EUCLIDEAN DISTANCE*

Oleh:

Salamah Nur Aqidah  
NIM 15306141049

## ABSTRAK

Penelitian ini bertujuan mendeteksi wajah pada citra menggunakan metode *skin detection* berdasarkan model warna YCbCr, menganalisis *principal component* dan *euclidean distance* untuk pengenalan wajah, serta mengetahui tingkat keakuratan dalam mengenali wajah.

Penelitian ini dimulai dengan pendeteksian bagian wajah dan bukan wajah pada citra menggunakan metode *skin detection*. Citra asli RGB dikonversi ke dalam model warna YCbCr, kemudian dilakukan pemisahan atau segmentasi piksel warna kulit pada komponen krominan berdasarkan *threshold* yang telah ditentukan. Hasil citra segmentasi dikenai morfologi *opening* (erosi diikuti dilasi) untuk menghilangkan derau. Kemudian dilakukan pelabelan dan pemotongan citra yang diidentifikasi sebagai wajah. Setelah bagian wajah terdeteksi, langkah selanjutnya proses ekstraksi ciri menggunakan *principal component analysis* dengan cara mereduksi citra 2D menjadi 1D; mencari rata-rata citra; menormalisasi citra; menghitung matriks kovarian; menghitung nilai eigen dan vektor eigen; menghitung *eigenfaces*; dan menghitung bobot citra. Klasifikasi citra menggunakan metode *euclidean distance* dengan mencari jarak minimum antara bobot citra uji dengan bobot citra latih.

Metode PCA pada penelitian ini dapat mereduksi fitur-fitur wajah yang semula matriksnya berukuran  $34 \times 20400$  menjadi matriks berukuran  $34 \times 200$ , matriks tersebut merupakan komponen terpilih karena memberikan pengaruh yang signifikan terhadap *database*. Tingkat akurasi pengenalan wajah pada penelitian ini sebesar 95% karena mampu mengidentifikasi wajah meskipun wajah dengan ekspresi yang berbeda dan menggunakan aksesoris berupa kacamata.

**Kata Kunci:** pengolahan citra, pengenalan wajah, *principal component analysis*, *euclidean distance*.

# **IMAGE PROCESSING FOR HUMAN FACE RECOGNITION USING PRINCIPAL COMPONENT ANALYSIS AND EUCLIDEAN DISTANCE**

By:

Salamah Nur Aqidah

NIM 15306141049

## **ABSTRACT**

The study aims to detect faces in an image using skin detection method based on YCbCr color model, to analyze principal component and euclidean for face recognition, and to know the level of accuracy in face recognition.

The study was begun with face detection using real face (not an image) by skin detection method. The original RGB image was converted into the YCbCr color model, then the skin color pixel segmentation of the chrominance component was based on the threshold have been determined. The results of image segmentation are subject to opening morphology (erosion followed by dilation) to eliminate noise. Then labeling and cropping of the image identified as faces. After the face is detected, the next step is the feature extraction process using principal component analysis by reducing 2D images to 1D; determining the average of the image; normalizing the image; calculating covariant matrices; calculating eigenvalues and eigenvectors; determining eigenfaces; and calculating the weight of the image. Image classification uses the euclidean distance method by finding the minimum distance between the weight of the test image and the weight of the training image.

The PCA method in this study can reduce facial features whose matrix size is  $34 \times 20400$  to  $34 \times 200$ , and this matrix is the chosen component because it has a significant influence on the database. The accuracy rate of facial recognition in this study was 95% because it was able to identify faces even though faces with different expressions and using accessories in the form of glasses.

**Keywords:** image processing, face recognition, principal component analysis, euclidean distance.

## SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini:

Nama : Salamah Nur Aqidah  
NIM : 15306141049  
Program Studi : Fisika  
Judul Tas : Pengolahan Citra Untuk Pengenalan Wajah Manusia  
Menggunakan *Principal Component Analysis* dan  
*Euclidean Distance*

menyatakan bahwa skripsi ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Yogyakarta, 02 Agustus 2019

Yang menyatakan,



Salamah Nur Aqidah

NIM. 15306141049

**LEMBAR PERSETUJUAN**

Tugas Akhir Skripsi dengan Judul

**PENGOLAHAN CITRA UNTUK PENGENALAN WAJAH MANUSIA  
MENGUNAKAN *PRINCIPAL COMPONENT ANALYSIS* DAN  
*EUCLIDEAN DISTANCE***

Disusun Oleh:

Salamah Nur Aqidah

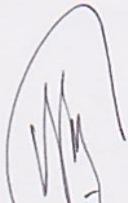
NIM 15306141049


Telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan  
Ujian Akhir Tugas Akhir Skripsi bagi yang bersangkutan.

Yogyakarta, 2 Agustus 2019

Mengetahui,  
Ketua Program Studi

Disetujui,  
Dosen Pembimbing

  
Nur Kadarisman. M.Si.  
NIP. 19640205 199101 1 001

  
Agus Purwanto, M.Sc.  
NIP. 19650813 199512 1 001

**HALAMAN PENGESAHAN**

Tugas Akhir Skripsi




**PENGOLAHAN CITRA UNTUK PENGENALAN WAJAH MANUSIA  
MENGUNAKAN *PRINCIPAL COMPONENT ANALYSIS* DAN  
*EUCLIDEAN DISTANCE***

Disusun Oleh:

Salamah Nur Aqidah  
NIM 15306141049

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi Fisika  
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Yogyakarta  
Pada Tanggal 14 Agustus 2019

**TIM PENGUJI**

Nama/Jabatan	Tanda Tangan	Tanggal
Agus Purwanto, M.Sc. Ketua Penguji/Pembimbing		21-8-2019
Dr. Heru Kuswanto, M.Si. Penguji		19-08-2019
Sumarna, M.Si., M.Eng. Sekretaris		19-08-2019

Yogyakarta, 22 Agustus 2019  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Negeri Yogyakarta



**Dr. Hartono**

NIP. 19620329 198702 1 002

## **MOTTO**

Balasan dari kebaikan adalah kebaikan selanjutnya

**HALAMAN PERSEMBAHAN**

*LAA ILAHA ILLALLAH WAHDAHU LAA SYARIKA LAH  
LAHUL MULKU WA LAHUL HAMDU  
WA HUWA 'ALA KULLI SYAI'IN QODIR*

*Teruntuk kedua orang tua, serta kakak-kakakku.*



## KATA PENGANTAR

Puji syukur saya panjatkan kepada Allah SWT atas berkat, rahmat, dan karunia-Nya. Tugas Akhir Skripsi untuk memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Sains dengan judul “**Pengolahan Citra Untuk Pengenalan Wajah Manusia Menggunakan *Principal Component Analysis* dan *Euclidean Distance***” dapat disusun sesuai dengan harapan.

Tugas Akhir Skripsi ini dapat diselesaikan tidak lepas dari bantuan dan kerjasama dengan pihak lain. Berkenaan dengan hal tersebut, penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. Hartono, M.Si. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Yogyakarta yang telah memberikan banyak kemudahan dan fasilitas selama melakukan studi.
2. Bapak Yusman Wiyatmo, M.Si., selaku Ketua Jurusan Pendidikan Fisika FMIPA Universitas Negeri Yogyakarta yang telah memberikan izin penelitian tugas akhir skripsi ini.
3. Bapak Nur Kadarisman, M.Si., selaku Ketua Program Studi Fisika FMIPA Universitas Negeri Yogyakarta yang telah memberikan izin dalam pelaksanaan skripsi ini.
4. Bapak Agus Purwanto, M.Sc. selaku Dosen Pembimbing skripsi yang senantiasa memberikan bimbingan dengan penuh kesabaran, pengarahan, ilmu, kritik dan saran yang sangat membangun sehingga skripsi ini dapat terselesaikan dengan baik.
5. Bapak Wipar Sunu Brams Dwandaru, S.Si., M.Sc., Ph.D., selaku Penasehat Akademik yang telah memberikan izin dan memberikan bimbingan serta arahan dalam pelaksanaan perkuliahan dan tugas akhir skripsi.
6. Semua Dosen Jurusan Pendidikan Fisika FMIPA UNY yang telah memberikan pengajaran dan ilmu yang bermanfaat.

7. Kedua orang tua penulis, Ibu Yayat Mulyati dan Bapak Endang Komarudin yang selalu memberikan kasih sayang dan perhatian yang tidak terhingga kepada penulis.
8. Kakak-kakak penulis, Anas Al Amin, Umul Kirom, Nur Habibah, dan Husnul Rofiqoh yang selalu menginspirasi.
9. Teman-teman Fisika kelas E 2015 yang telah menemani selama berproses di tanah perantauan.
10. Wuri, Ratna, Riski, Emi, Hanan, Naza, dan Udin yang telah bersedia difoto untuk menjadi objek penelitian penulis.
11. Teman-teman Pejuang Cumlaude (Riski, Wuri, dan Kendis) yang selalu meluangkan waktu dan hatinya untuk bermain dan berjuang bersama.
12. Semua pihak yang telah membantu dan mendukung baik secara langsung dan tidak langsung yang tidak mampu disebutkan satu-persatu.

Penulis menyadari bahwa Tugas Akhir Skripsi ini masih jauh dari sempurna. Oleh sebab itu, kritik yang membangun akan diterima dengan senang hati untuk perbaikan lebih lanjut. Semoga skripsi ini dapat bermanfaat bagi masyarakat dan dunia pendidikan.

Yogyakarta, 02 Agustus 2019

Penulis,



Salamah Nur Aqidah

NIM. 15306141049

## DAFTAR ISI

	Halaman
Halaman Judul.....	i
ABSTRAK .....	ii
SURAT PERNYATAAN.....	iv
LEMBAR PERSETUJUAN.....	v
HALAMAN PENGESAHAN.....	vi
MOTTO .....	vii
HALAMAN PERSEMBAHAN .....	viii
KATA PENGANTAR .....	ix
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR .....	xiv
DAFTAR LAMPIRAN.....	xvi

### **BAB I PENDAHULUAN**

A. Latar Belakang .....	1
B. Identifikasi Masalah .....	5
C. Batasan Masalah.....	5
D. Perumusan Masalah .....	6
E. Tujuan .....	6
F. Manfaat .....	7

### **BAB II KAJIAN PUSTAKA**

A. Pengolahan Citra Digital .....	8
B. Ruang Warna.....	9
C. Pemodelan Warna Kulit Dalam Ruang YCbCr .....	13
D. Morfologi Citra .....	15
E. Pelabelan .....	19
F. Citra <i>Grayscale</i> .....	21

G. Interpolasi Citra.....	21
H. Pengenalan Pola .....	23
I. Ekstraksi Ciri.....	24
J. <i>Principal Component Analysis</i> .....	25
K. <i>Eigenface</i> .....	28
L. <i>Euclidean Distance</i> .....	29
M. Contoh Algoritma PCA .....	30
 <b>BAB III METODE PENELITIAN</b>	
A. Waktu dan Tempat Penelitian .....	39
B. Variabel Penelitian .....	39
C. Teknik Pengumpulan Data.....	40
D. Instrumen penelitian.....	41
E. Prosedur Penelitian dan Teknik Analisis .....	41
F. Diagram Alir .....	44
 <b>BAB IV HASIL DAN PEMBAHASAN</b>	
A. Deteksi Citra Wajah .....	48
B. Pra-Proses.....	54
C. <i>Principal Component Analysis</i> .....	55
D. Tahap Pengenalan .....	63
E. Tahap Pengujian.....	65
 <b>BAB V KESIMPULAN DAN SARAN</b>	
A. Kesimpulan .....	74
B. Saran.....	75
 <b>DAFTAR PUSTAKA</b> .....	 76
 <b>LAMPIRAN</b> .....	 78

## DAFTAR TABEL

Tabel 1. Matriks augmentasi citra data pelatihan.....	57
Tabel 2. Matriks kovarian .....	60
Tabel 3. Hasil pangujian pada 22 sampel citra .....	70

## DAFTAR GAMBAR

Gambar 1. Koordinat citra digital .....	9
Gambar 2. Skema Kubus Warna RGB.....	10
Gambar 3. Kubus warna RGB 24-bit.....	11
Gambar 4. Komponen Citra RGB .....	12
Gambar 5. Komponen YCbCr .....	13
Gambar 6. Distribusi bagian kulit dan bukan kulit dalam ruang Cb-Cr .....	14
Gambar 7. Deteksi bagian kulit dalam ruang Cb-Cr.....	14
Gambar 8. Beberapa contoh <i>strel</i> .....	15
Gambar 9. Ilustrasi morfologi dilasi. ....	17
Gambar 10. Ilustrasi morfologi erosi. ....	18
Gambar 11. <i>Downsampling</i> dengan metode <i>Nearest Neighbor</i> .....	23
Gambar 12. <i>Upsampling</i> dengan metode <i>Nearest Neighbor</i> .....	23
Gambar 13. Formasi vektor wajah .....	27
Gambar 14. Data citra wajah dengan lima pose.....	40
Gambar 15. Data citra wajah dengan enam ekspresi wajah.....	41
Gambar 16. Diagram alir pengenalan wajah.....	44
Gambar 17. Diagram alir deteksi wajah.....	45
Gambar 18. Diagram alir ekstraksi ciri .....	46
Gambar 19. Diagram alir identifikasi wajah .....	47
Gambar 20. Contoh <i>database</i> dengan 5 pose yang berbeda .....	48
Gambar 21. Contoh <i>database</i> dengan 6 ekspresi wajah yang berbeda .....	48
Gambar 22. Komponen Citra YCbCr .....	49
Gambar 23. Citra <i>skin likelihood</i> .....	50
Gambar 24. Grafik distribusi kemungkinan kulit pada sampel citra .....	50
Gambar 25. Morfologi <i>Opening</i> .....	51
Gambar 26. Representasi citra morfologi <i>opening</i> .....	52
Gambar 27. Pelabelan 4 Konektivitas.....	52
Gambar 28. Area yang diidentifikasi sebagai wajah.....	53
Gambar 29. Beberapa contoh citra hasil deteksi wajah .....	53
Gambar 30. Konversi citra RGB menjadi citra <i>grayscale</i> .....	54
Gambar 31. Histogram <i>resize</i> dengan metode INN .....	55
Gambar 32. Citra hasil <i>resize</i> resolusi 120 x 170 piksel .....	55
Gambar 33. <i>Database</i> citra pelatihan.....	56
Gambar 34. Reduksi dimensi citra 2D menjadi 1D .....	57
Gambar 35. Citra rata-rata pelatihan.....	58
Gambar 36. Citra hasil normalisasi.....	59
Gambar 37. Spektrum <i>Eigenvalue</i> .....	62
Gambar 38. Citra <i>eigenfaces</i> .....	62
Gambar 39. Bobot citra pelatihan .....	64
Gambar 40. Nilai maksimum dari jarak <i>euclidean</i> bobot citra pelatihan .....	65
Gambar 41. <i>Database</i> citra pengujian.....	66

Gambar 42. Sampel Citra pengujian .....	66
Gambar 43. Contoh fitur citra pengujian .....	67
Gambar 44. Bobot sampel citra pengujian .....	67
Gambar 45. <i>Euclidean distance</i> sampel citra pengujian .....	68
Gambar 46. Citra pengenalan indeks ke-7 .....	68
Gambar 47. Citra pengujian selain <i>database</i> .....	70

## DAFTAR LAMPIRAN

Lampiran 1. Matriks kovarian (C) berukuran simetris bujur sangkar $35 \times 35$ ....	78
Lampiran 2. Eigen vektor.....	81
Lampiran 3. Bobot citra pelatihan.....	84



## **BAB I**

### **PENDAHULUAN**

#### **A. Latar Belakang**

Pengolahan citra dan *computer vision* merupakan sebuah penemuan di bidang komputer yang digunakan untuk menghasilkan suatu sistem yang hampir mendekati dengan sistem visual manusia pada umumnya. Pengolahan citra adalah salah satu jenis teknologi untuk menyelesaikan masalah mengenai pemrosesan gambar. Dalam pengolahan citra, gambar yang ada diolah sedemikian rupa sehingga gambar tersebut lebih mudah diproses, sedangkan *computer vision* mempunyai tujuan utama yaitu untuk membuat suatu keputusan yang berguna tentang objek fisik nyata yang didapat dari perangkat atau sensor. Penerapan pengolahan citra dan *computer vision* saat ini banyak digunakan pada perusahaan atau lembaga untuk meningkatkan sistem keamanan berbasis data pada karakteristik tubuh atau perilaku yang disebut teknologi biometrik.

Teknologi biometrik merupakan pendekatan yang berkaitan dengan mengidentifikasi seseorang berdasarkan karakteristik fisiknya yang unik dan sifat biologisnya. Ada beberapa pengembangan pendekatan teknologi biometrik antara lain pengenalan wajah (*face recognition*), pengenalan sidik jari, geometri telapak tangan, pengenalan iris mata (*iris recognition*) dan pengenalan tulisan tangan (*handwriting recognition*). Salah satu teknik biometrik yang sangat menarik adalah aplikasi yang mampu mendeteksi dan

mengidentifikasi pola wajah.

Mengenali wajah seseorang merupakan suatu hal yang mudah dilakukan oleh manusia. Seseorang akan cepat mengenal, menghafal dan membedakan wajah setiap orang yang dikenal sebelumnya walaupun dengan ekspresi yang berbeda-beda ketika ditemui, atau bahkan dalam keadaan terang maupun gelap. Tetapi tidak demikian untuk komputer, agar dapat mengenali wajah manusia diperlukan perlakuan khusus sehingga ketika diberi *input* sebuah gambar, komputer dapat mendeteksi apakah dalam gambar tersebut terdapat sebuah gambar wajah atau tidak dan dapat mengenali wajah tersebut.

Pengenalan wajah adalah salah satu aplikasi yang paling berkembang dalam *computer vision* dan *pattern recognition* (pengenalan pola). Tujuan utamanya adalah untuk mengenali wajah manusia dari gambar atau video menggunakan *database* wajah yang tersimpan. Pengembangan sistem pengenalan wajah adalah masalah yang penting karena wajah manusia terlalu kompleks sehingga untuk mengembangkan suatu perhitungan yang ideal adalah hal yang sulit (Alwakeel & Shaaban, 2010: 25). Pengenalan wajah merupakan identifikasi personal yang didasarkan dari geometrik atau fitur statistik yang didapat dari citra wajah manusia. Proses pengenalan wajah dipengaruhi oleh faktor variabilitas *extra-personal* dan *intra-personal*. Faktor variabilitas *extra-personal* adalah faktor yang timbul saat proses pengenalan wajah yang dilakukan pada wajah yang berbeda karena faktor ras dan genetika. Faktor variabilitas *intra-personal* adalah variasi yang muncul pada wajah yang sama dimana hal ini disebabkan oleh perubahan-perubahan internal

(*deformations*) seperti posisi wajah, keberadaan komponen struktural, ekspresi wajah, oklusi, orientasi citra dan kondisi pencitraan (Bradski, 2010: 1808). Aplikasi dari pengenalan wajah dapat diterapkan dalam bidang keamanan (*security system*) seperti ijin akses masuk ruangan, pengawasan lokasi (*surveillance*) dan sebagainya.

Pengenalan wajah memiliki beberapa tahapan penting agar hasil pengenalan dapat berjalan dengan baik yaitu akuisisi data, *pre-processing*, representasi data dan pengambilan keputusan. Akuisisi data adalah semua hal yang berhubungan dengan pengambilan citra wajah yang akan digunakan untuk pengenalan wajah, misalnya dengan menggunakan *digital camera* dan *scanner*. *Pre-processing* adalah cara bagaimana membuat citra wajah tersebut agar memiliki batasan yang sesuai kebutuhan untuk proses selanjutnya. *Pre-processing* dapat dilakukan dalam banyak proses misalnya dengan *grayscaleing*, *face detection*, *cropping*, *resizing* dan hal yang dapat mengondisikan citra wajah. Representasi data umumnya adalah representasi dari fitur-fitur yang diekstraksi dan diseleksi dari citra wajah untuk mengurangi dimensionalitas data. Pengambilan keputusan adalah tahap dimana dilakukan pengklasifikasian dengan mengukur seberapa mirip data uji dengan representasi data yang ada dalam *database* baik menggunakan *euclidean distance*, jarak antara dua vektor, koefisien korelasi dan sebagainya.

Pendeteksian wajah adalah tahap awal yang penting dalam sistem pengenalan wajah. Tujuan dari pendeteksian wajah adalah mengidentifikasi semua area yang ada dalam citra untuk menemukan area wajah dan area bukan

wajah. Yang, *et al* (2002) menyajikan suatu survei algoritma pendeteksian wajah yang kritis dan menyeluruh. Salah satu metode pendeteksian wajah yang cukup populer terutama dalam citra berwarna adalah melalui *skin detection*, hal ini karena warna kulit adalah fitur penting dalam wajah.

Metode *skin detection* akan memisahkan nilai RGB menjadi informasi *luminance* dan *chrominance*. Ruang warna RGB pada citra asli masih mengandung efek cahaya yang menyebabkan karakteristik warna kulit bisa berubah, karenanya perlu dikonversi ke dalam bentuk warna kromatik. Untuk mengurangi efek pencahayaan itu digunakan model warna YCbCr, model warna ini terdiri dari 3 komponen yaitu: Y bernilai *luminance* (tingkat keterangan), Cb bernilai *Chrominance Blue* (tingkat kebiruan) dan Cr bernilai *Chrominance Red* (tingkat kemerahan) (Chang *et al*, 2000).

Secara umum sistem pengenalan citra wajah dibagi menjadi 2 jenis, yaitu sistem *feature based* dan sistem *image based*. Pada sistem *feature based* digunakan fitur yang diekstraksi dari komponen citra wajah seperti mata, hidung dan mulut yang kemudian hubungan antara fitur-fitur tersebut dimodelkan secara geometris. Sedangkan sistem *image based* menggunakan informasi mentah dari piksel citra yang kemudian direpresentasikan dalam metode tertentu, salah satunya *Principal Component Analysis* (PCA) (Xiaoguang Lu, 2008: 7).

*Eigenface* merupakan suatu metode yang digunakan untuk mentransformasikan dan mereduksi dimensi dari suatu citra. *Eigenface* menggunakan metode *Principal Component Analysis* (PCA) yaitu suatu

metode matematika untuk merepresentasikan sebuah objek, mengekstraksi ciri-ciri sebuah objek dan mereduksi sebuah objek dengan cara mentransformasikannya menggunakan *eigenvalue* dan vektor eigen secara linier.

## **B. Identifikasi Masalah**

Dari uraian latar belakang di atas, maka dapat diidentifikasi permasalahan yaitu

1. Perbedaan antara wajah dan bukan wajah pada citra menggunakan metode *skin detection*.
2. Ekstraksi ciri citra wajah menggunakan *Principal Component Analysis*.
3. Pencocokan citra wajah sesuai *database* menggunakan metode *euclidean distance*.

## **C. Batasan Masalah**

Batasan masalah dalam penelitian ini adalah;

1. Program dibuat menggunakan Matlab R2013a.
2. Hanya satu objek wajah pada satu citra.
3. Citra yang digunakan dengan format jpg.
4. Citra wajah yang akan dijadikan citra pelatihan dan citra uji berukuran sama.
5. Proses deteksi wajah menggunakan metode *skin detection* berdasarkan model warna YCbCr.

6. Proses ekstraksi ciri citra wajah menggunakan *Principal Component Analysis*.
7. Proses pencocokan citra menggunakan metode *euclidean distance*.

#### **D. Perumusan Masalah**

Berdasarkan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut

1. Bagaimana mendeteksi bagian wajah pada citra menggunakan metode *skin detection* berdasarkan model warna YCbCr?
2. Bagaimana menganalisis *principal component* untuk pengenalan pola wajah manusia?
3. Bagaimana menganalisis *euclidean distance* untuk mencocokkan wajah manusia sesuai *database* yang ada?
4. Berapa tingkat keakuratan pengenalan wajah manusia dengan menggunakan *Principal Component Analysis* dan *euclidean distance*?

#### **E. Tujuan**

1. Mendeteksi wajah pada citra menggunakan metode *skin detection* berdasarkan model warna YCbCr.
2. Menganalisis *principal component* untuk pengenalan pola wajah manusia.

3. Menganalisis *euclidean distance* untuk mencocokkan wajah manusia sesuai *database* yang ada.
4. Mengetahui tingkat keakuratan pengenalan wajah manusia dengan menggunakan *Principal Component Analysis* dan *euclidean distance*.

#### **F. Manfaat**

1. Mengetahui cara kerja pengenalan wajah menggunakan *Principal Component Analysis* dan *Euclidean Distance*.
2. Sebagai tambahan referensi dalam penelitian analisis pola wajah manusia.
3. Dapat digunakan sebagai pijakan penelitian lanjutan untuk aplikasi dari *face recognition*.

## BAB II

### KAJIAN PUSTAKA

Pada bab ini akan dibahas literatur dan landasan teori yang relevan dengan penelitian.

#### A. Pengolahan Citra Digital

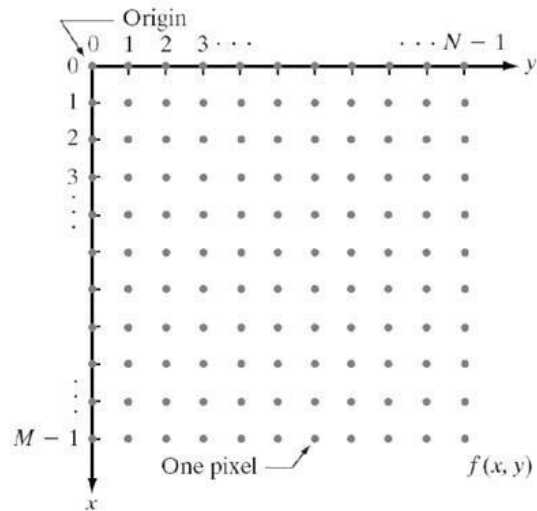
Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi suatu objek. Citra yang berupa *output* dari sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan (Sutoyo *et al*, 2009:9).

Secara umum pengolahan citra digital menunjuk pada pemrosesan gambar dimensi dua menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data berdimensi dua. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu (Putra, 2010:19).

Suatu citra dapat didefinisikan sebagai fungsi  $f(x, y)$  berukuran  $M$  baris dan  $N$  kolom, dengan  $x$  dan  $y$  adalah koordinat spasial, dan amplitudo  $f$  di titik koordinat  $(x,y)$  dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai  $(x,y)$ , dan nilai amplitudo  $f$  secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra



tersebut adalah citra digital. Gambar 1 menunjukkan posisi koordinat citra digital.



Gambar 1. Koordinat citra digital  
Sumber (Gonzalez & Woods, 2008:55)

Citra digital dapat ditulis dalam bentuk matriks sebagai berikut

$$f(x, y) = \begin{bmatrix} f(1,1) & f(1,2) & \cdots & f(1,N) \\ f(2,1) & f(2,2) & \cdots & f(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ f(M,1) & f(M,2) & \cdots & f(M,N) \end{bmatrix}$$

Nilai pada suatu irisan antara baris dan kolom (pada posisi  $x, y$ ) disebut dengan *picture elements*, *image elements*, *pels*, atau *pixel* (Gonzalez & Woods, 2008:56).

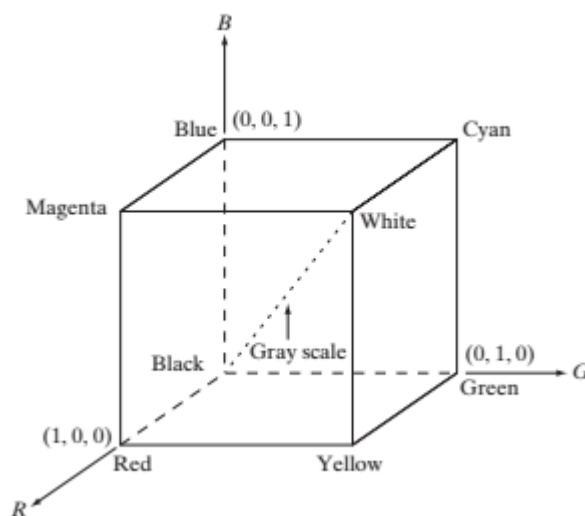
## B. Ruang Warna

Ruang warna atau model warna merupakan suatu spesifikasi sistem koordinat dan subruang dengan setiap warna dinyatakan pada satu titik di dalamnya. Tujuan dibentuknya ruang warna adalah untuk memfasilitasi

spesifikasi warna dalam bentuk standar. Tiga model warna yang paling populer adalah RGB digunakan dalam grafik komputer; YIQ, YUV atau YCbCr digunakan dalam sistem video; dan CMYK digunakan dalam percetakan (Datta *et al*, 2016:48).

### 1. Model Warna RGB

RGB adalah suatu model warna yang terdiri dari komponen warna primer yaitu merah, hijau, dan biru. Model ini didasarkan pada sistem koordinat *Cartesius*. Subruang warna ditunjukkan pada Gambar 2, di mana nilai primer RGB ada di tiga sudut; warna sekunder *cyan*, magenta, dan kuning ada di tiga sudut lainnya; hitam berada di daerah asal; dan putih di sudut terjauh dari daerah asalnya. Dalam model ini, skala abu-abu berada sepanjang garis diagonal yang menghubungkan titik hitam ke titik putih. Diasumsikan bahwa semua nilai warna telah dinormalisasi sehingga kubus yang ditunjukkan pada Gambar 2 adalah satuan kubus sehingga semua nilai R, G, dan B diasumsikan berada dalam kisaran [0, 1] (Woods & Gonzalez, 2008:402).



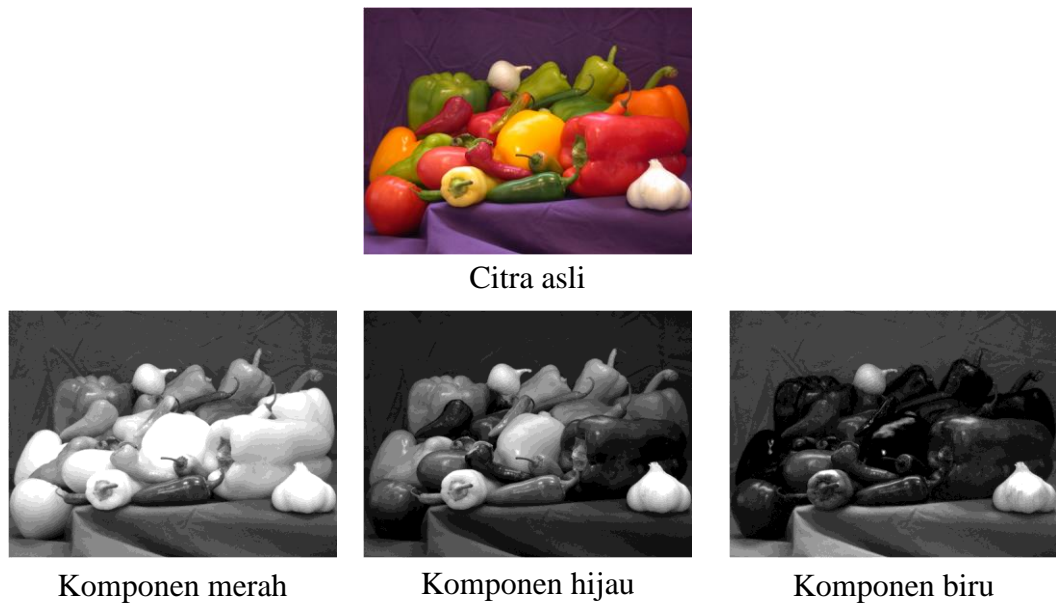
Gambar 2. Skema Kubus Warna RGB  
Sumber (Woods & Gonzalez, 2008:402)



Gambar 3. Kubus warna RGB 24-bit  
Sumber (Woods & Gonzalez, 2008:403)

Gambar yang diwakili dalam model warna RGB terdiri dari tiga komponen. Ketika dimasukkan ke monitor, ketiga komponen ini bergabung pada layar untuk menghasilkan warna komposit. Jumlah bit yang digunakan untuk mewakili setiap piksel dalam ruang RGB disebut *pixel depth*. Setiap komponen warna menggunakan delapan bit (nilainya berkisar antara 0 sampai dengan 255). Gambar 3 menunjukkan kubus warna secara nyata dengan resolusi 24-bit sehingga jumlah total warna dalam kubus warna tersebut adalah  $(2^8)^3 = 16.777.216$  variasi warna. Variasi ini sudah lebih dari cukup untuk memvisualisasikan seluruh warna yang dapat dilihat penglihatan manusia. (Woods & Gonzalez, 2008:403).

Gambar 4 merupakan salah satu contoh komponen model warna RGB; citra asli RGB (255, 255, 255); komponen merah (255, 0, 0); komponen hijau (0, 255, 0); dan komponen biru (0, 0, 255).



Gambar 4. Komponen Citra RGB

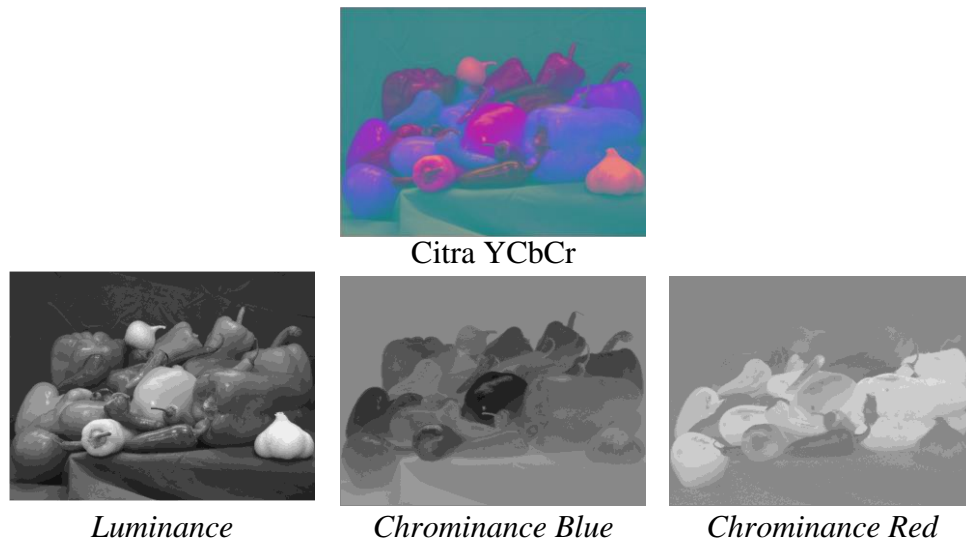
## 2. Ruang Warna YCbCr

Ruang warna YCbCr banyak digunakan untuk video digital. Dalam format ini, informasi *luminance* disimpan sebagai komponen tunggal (Y), dan informasi *chrominance* disimpan sebagai komponen *color-difference* (Cb dan Cr). Cb mewakili perbedaan antara komponen biru dan nilai referensi. Cr mewakili perbedaan antara komponen merah dan nilai referensi (Qidwai & Chen, 2009:28). Transformasi RGB ke YCbCr dapat dilakukan dengan menggunakan persamaan 2.1 (Shih, 2010:148).

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 65,481 & 128,553 & 24,966 \\ -39,797 & -74,203 & 112 \\ 112 & -93,786 & -18,214 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \quad (2.1)$$

Gambar 5 menunjukkan transformasi dari RGB ke YCbCr mengambil nilai input RGB dengan masing-masing komponen dalam kisaran [0-255] dan mengubahnya menjadi Y, Cb, dan Cr, dalam rentang

[0, 255], [-128, 127], dan [-128, 127] (Datta *et al*, 2016:88).

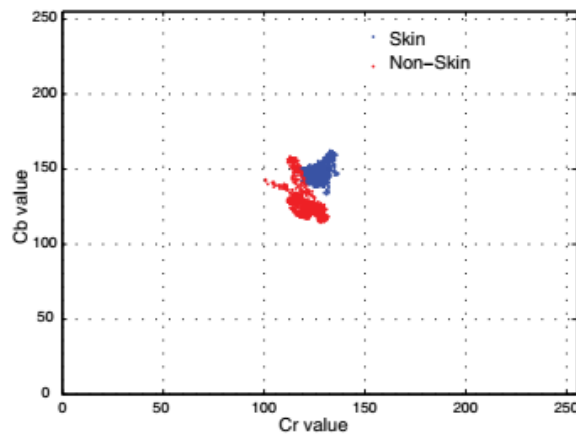


Gambar 5. Komponen YCbCr

### C. Pemodelan Warna Kulit Dalam Ruang YCbCr

Keutamaan dari mengonversi gambar ke domain YCbCr adalah pengaruh luminositas atau tingkat kecerahan dapat dihilangkan selama pemrosesan citra. Dalam domain RGB, setiap komponen citra (merah, hijau dan biru) memiliki kecerahan yang berbeda. Namun, dalam domain YCbCr, semua informasi tentang kecerahan diberikan oleh komponen Y, karena komponen Cb (krominan biru) dan Cr (krominan merah) terpisah dari luminositas. Komponen Cb dan Cr memberikan indikasi yang baik tentang apakah suatu piksel merupakan bagian dari kulit atau tidak. Gambar 6 menunjukkan korelasi kuat antara nilai Cb dan Cr untuk citra kulit (berwarna biru). Gambar 6 juga menunjukkan distribusi citra non-kulit (berwarna merah) di ruang Cb-Cr. Oleh karena itu dengan menerapkan nilai ambang batas maksimum dan minimum untuk komponen kulit maka komponen Cb dan Cr

dapat dengan mudah disegmentasi dari gambar latar belakang. Gambar 7 menunjukkan segmentasi kulit (Datta *et al*, 2016:91).



Gambar 6. Distribusi bagian kulit dan bukan kulit dalam ruang Cb-Cr  
 Sumber (Datta *et al*, 2016:90)



Gambar 7. Deteksi bagian kulit dalam ruang Cb-Cr  
 Sumber (Datta *et al*, 2016:90)

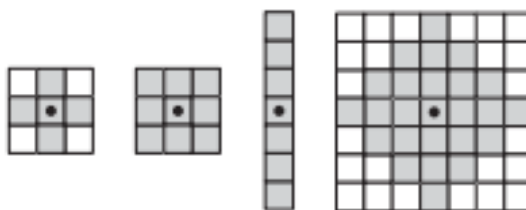
Nilai ambang batas maksimum dan minimum untuk deteksi warna kulit berdasarkan model warna YCbCr telah diteliti oleh Chai dan Ngan (1999:555) dengan aturan sebagai berikut

$$\text{warna kulit} = \begin{cases} 1, & \begin{cases} 77 \leq Cb \leq 127 \\ 133 \leq Cr \leq 173 \end{cases} \\ 0 & \text{bukan kulit} \end{cases}$$

## D. Morfologi Citra

Morfologi adalah teknik pengolahan citra berdasarkan bentuk segmen citra. Morfologi bertujuan untuk memperbaiki hasil segmentasi. Operator morfologi melibatkan dua larik piksel. Larik pertama berupa citra yang akan dikenai operasi morfologi, sedangkan larik kedua dinamakan sebagai kernel atau *structuring element (strel)*. Operasi dasar dalam morfologi adalah dilasi dan erosi, yang dapat dikombinasikan secara berurutan untuk menghasilkan operasi lain, seperti *opening* dan *closing* (Shih, 2010:64).

*Structuring element (strel)* memiliki ukuran yang biasanya lebih kecil dibandingkan dengan citra yang akan diolah. Nilai piksel pada matriks *strel* berisi nol atau satu. *Origin* pada bagian tengah dari *strel* menjadi pusat dalam melakukan operasi terhadap citra (Zhou *et al*, 2010:32). *Strel* memiliki beberapa bentuk yang biasanya digunakan, yaitu *disk*, *linear*, *rectangle*, *square*, dan *diamond*. Gambar 8 menunjukkan beberapa contoh *strel*



Gambar 8. Beberapa contoh *strel*  
Sumber (Gonzalez & Woods, 2008:629)

### 1. Dilasi

Operasi dilasi dilakukan untuk memperbesar ukuran segmen objek dengan menambah lapisan di sekeliling objek. Terdapat dua cara untuk melakukan operasi ini, yaitu dengan cara mengubah semua titik latar yang

bertetangga dengan titik batas menjadi titik objek, atau lebih mudahnya set setiap titik yang tetangganya adalah titik objek menjadi titik objek. Cara kedua yaitu dengan mengubah semua titik di sekeliling titik batas menjadi titik objek, atau lebih mudahnya set semua titik tetangga sebuah titik objek menjadi titik objek.

Dengan  $A$  dan  $B$  sebagai set dalam  $Z^2$ , dilasi dari  $A$  terhadap  $B$  dilambangkan  $A \oplus B$ , didefinisikan sebagai persamaan 2.2 (Gonzalez & Woods, 2008:633)

$$A \oplus B = \{z \mid [(\hat{B})_z \cap A] \subseteq A\} \quad (2.2)$$

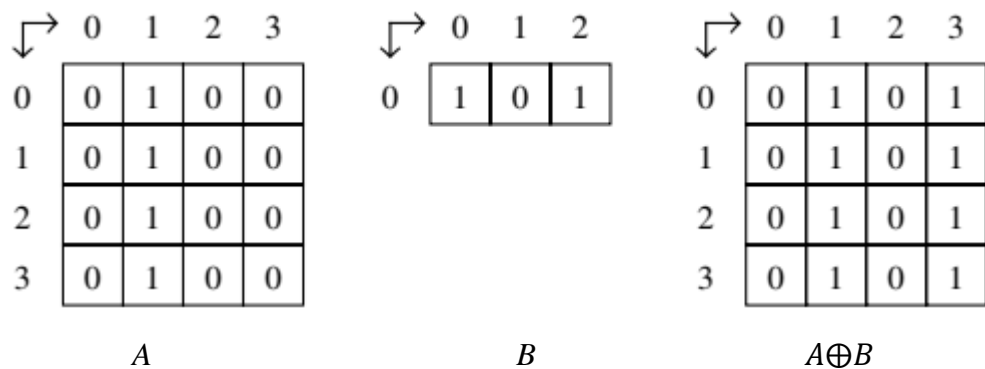
$B$  adalah *strel* dan  $A$  adalah himpunan (objek citra) yang akan dilakukan operasi dilasi. Gambar 9 menunjukkan ilustrasi morfologi dilasi.

Beberapa contoh dilasi menurut Shih (2010:66)

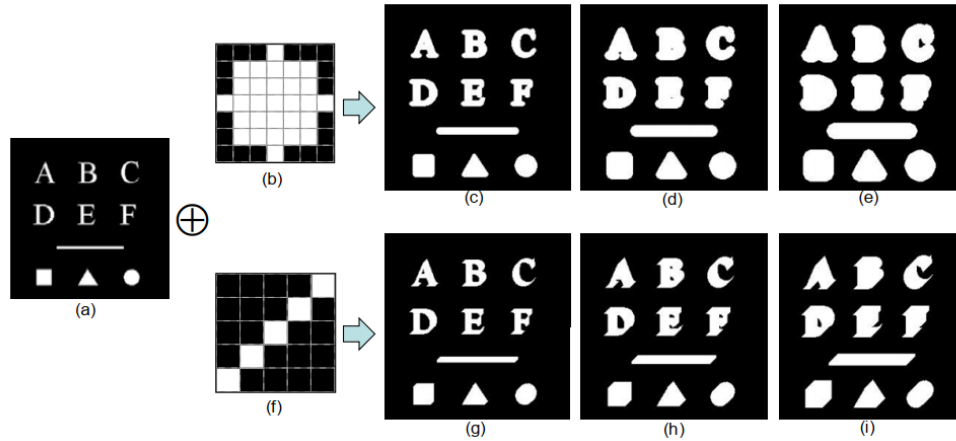
$$A = \{(0,1), (1,1), (2,1), (3,1)\}$$

$$B = \{(0,0), (0,2)\}$$

$$A \oplus B = \{(0,1), (1,1), (2,1), (3,1), (0,3), (1,3), (2,3), (3,3)\}$$







Gambar 9. Ilustrasi morfologi dilasi. (a) Citra biner; (b) kernel *disk*; (c) dilasi citra (a) oleh kernel (b); (d) setelah dua kali dilasi oleh kernel (b); (e) setelah tiga kali dilasi oleh kernel (b); (f) kernel *line*; (g) dilasi citra (a) oleh kernel (f); (h) setelah dua kali dilasi oleh kernel (f); (i) setelah tiga kali dilasi oleh kernel (f).

Sumber (Zhou *et al*, 2010:35)

## 2. Erosi

Operasi erosi adalah kebalikan dari operasi dilasi. Pada operasi ini, ukuran objek diperkecil dengan mengikis sekeliling objek. Cara yang dapat dilakukan juga ada dua. Cara pertama yaitu dengan mengubah semua titik batas menjadi titik latar dan cara kedua dengan menset semua titik di sekeliling titik latar menjadi titik latar.

Dengan  $A$  dan  $B$  sebagai set dalam  $Z^2$ , erosi dari  $A$  terhadap  $B$  dilambangkan  $A \ominus B$ , didefinisikan sebagai persamaan 2.3 (Gonzalez & Woods, 2008:631)

$$A \ominus B = \{z | (\hat{B})_z \subseteq A\} \quad (2.3)$$

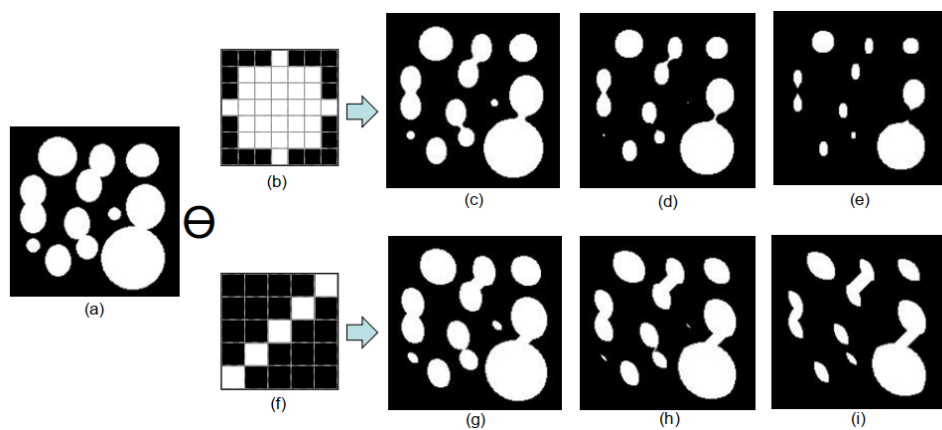
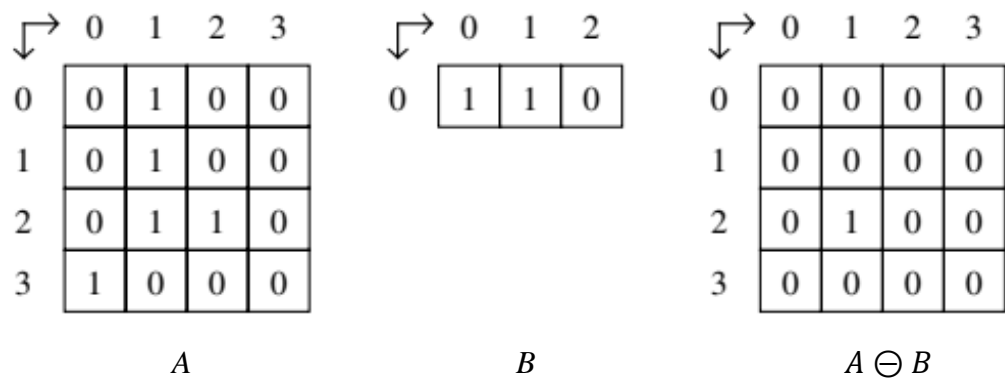
Erosi tidak memiliki sifat komutatif. Istilah yang setara dengan erosi adalah “menyusut” dan “mengurangi”. Gambar 10 menunjukkan ilustrasi morfologi erosi.

Beberapa contoh erosi menurut Shih (2010:67)

$$A = \{(0,1), (1,1), (2,1), (2,2), (3,0)\}$$

$$B = \{(0,0), (0,1)\}$$

$$A \ominus B = \{(2,1)\}$$



Gambar 10. Ilustrasi morfologi erosi. (a) Citra biner; (b) kernel *disk*; (c) erosi citra (a) oleh kernel (b); (d) setelah dua kali erosi oleh kernel (b); (e) setelah tiga kali erosi oleh kernel (b); (f) kernel *line*; (g) erosi citra (a) oleh kernel (f); (h) setelah dua kali erosi oleh kernel (f); (i) setelah tiga kali erosi oleh kernel (f).

Sumber (Zhou *et al*, 2010:36)

### 3. *Opening*

Operasi *opening* merupakan kombinasi antara operasi erosi dan dilasi yang dilakukan secara berurutan, tetapi citra asli dierosi terlebih dahulu baru kemudian hasilnya didilasi. Operasi ini digunakan untuk memutus bagian-bagian dari objek yang hanya terhubung dengan satu atau dua buah titik saja, atau menghilangkan objek-objek kecil dan secara umum mensmoothkan batas dari objek tanpa mengubah area objek secara signifikan. *Opening* adalah idempotent yaitu apabila operasi *opening* diulang-ulang tidak memberikan dampak yang berkelanjutan. Persamaan 2.4 menunjukkan *opening*  $A$  oleh  $B$  diperoleh dari erosi  $A$  oleh  $B$ , diikuti oleh dilasi citra hasil oleh  $B$ , yaitu

$$A \circ B = (A \ominus B) \oplus B \quad (2.4)$$

### 4. *Closing*

Operasi *closing* adalah kombinasi antara operasi dilasi dan erosi yang dilakukan secara berurutan. Citra asli didilasi terlebih dahulu, kemudian hasilnya dierosi. Operasi ini digunakan untuk menutup atau menghilangkan lubang-lubang kecil yang ada dalam segmen objek, menggabungkan objek yang berdekatan dan secara umum mensmoothkan batas dari objek tanpa mengubah objek secara signifikan.

## E. Pelabelan

*Connected Component Labeling* adalah sebuah algoritma pengelompokan sederhana yang bertujuan untuk mengisolasi, mengukur, dan mengidentifikasi potensi daerah obyek dalam citra (Rajaraman & Chokkalingam, 2013:83). Metode ini akan menghasilkan sebuah citra dengan

label baru yang sudah terkait satu sama lainnya antar sesama komponen. Dengan metode ini, akan dilakukan segmentasi untuk memisahkan setiap karakter yang terdapat pada citra.

Operasi pelabelan dari daerah obyek akan memberikan nama atau nomor yang unik ke semua piksel bernilai 1 yang termasuk dalam daerah tersebut. Hasil pelabelan adalah komponen individu yang dapat diekstraksi. Menurut Gonzalez and Woods (2008:69), ada dua jenis konektivitas yang digunakan pada citra dimensi dua 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 terhadap  $P(x,y)$ . Kumpulan dari piksel-piksel ini disebut dengan *4 neighbors of P(x,y)* yaitu piksel pada posisi horizontal  $\{P(x-1,y), P(x+1,y)\}$  dan posisi vertikal  $\{P(x,y-1), P(x,y+1)\}$ . Piksel tersebut diberi label yang sama dan dianggap satu objek. Pada konsep *4-Connected Neighbors* bila terdapat dua pixel yang bersinggungan secara diagonal maka akan dianggap dua objek. Piksel-piksel yang berdekatan dikatakan memiliki hubungan 8-konektivitas jika piksel-piksel tersebut terletak berdampingan secara horizontal, vertikal, dan diagonal terhadap  $P(x,y)$  yaitu pada posisi horizontal  $\{P(x-1,y), P(x+1,y)\}$ , posisi vertikal  $\{P(x,y-1), P(x,y+1)\}$ , dan posisi diagonal  $\{P(x-1,y-1), P(x+1,y-1), P(x-1,y+1), P(x+1,y+1)\}$ . Pada konsep 8-Konektivitas jika terdapat dua piksel yang bersinggungan baik secara diagonal maupun secara horizontal dan vertikal maka akan hanya dianggap satu objek.

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

4-Konektivitas

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

8-Konektivitas

## F. Citra *Grayscale*

Menurut Shih (2010:41), citra *grayscale* adalah suatu citra yang hanya memiliki warna tingkat keabuan. Warna abu-abu pada citra *grayscale* adalah warna RGB yang memiliki intensitas yang sama. Sehingga dalam citra *grayscale* hanya membutuhkan nilai intensitas tunggal dibandingkan dengan citra berwarna yang membutuhkan tiga intensitas untuk tiap pikselnya. Intensitas dari citra *grayscale* disimpan dalam 8 bit *integer* yang memberikan 256 kemungkinan yang dimulai dari level 0 sampai dengan 255 (0 untuk hitam dan 255 untuk putih dan nilai diantaranya adalah derajat keabuan). Transformasi citra RGB menjadi citra *grayscale* menggunakan persamaan 2.5 (Charbit & Blanchet, 2014:192).

$$Y = gray = 0,299 \times R + 0,587 \times G + 0,114 \times B \quad (2.5)$$

## G. Interpolasi Citra

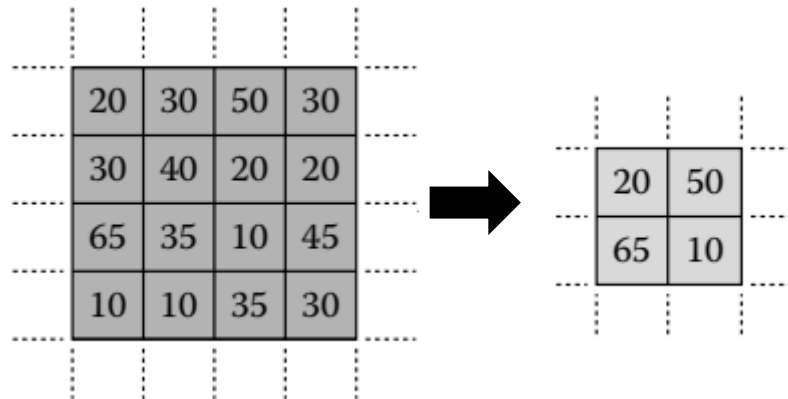
Interpolasi adalah proses pembuatan ulang (*resample*) piksel baru dari piksel yang telah diketahui dengan suatu fungsi tertentu. Menambah ukuran dari suatu citra disebut *upsampling*, sedangkan mengurangi ukurannya disebut *downsampling* (Sachs, 2001:1).

Ketika suatu gambar ditambah ukurannya, jumlah piksel akan meningkat dengan tetap mengacu pada citra aslinya. Semakin gambar diperbesar, maka

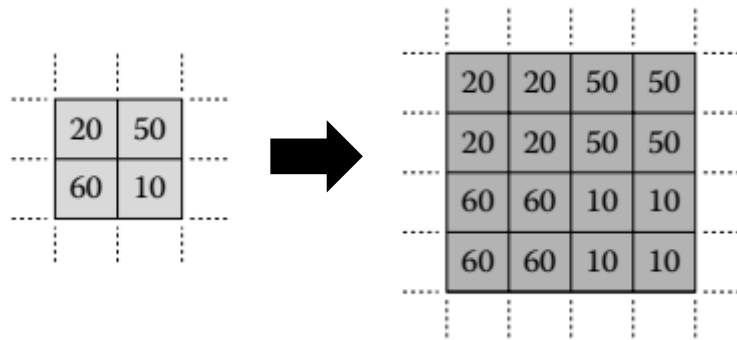
gambar biasanya akan semakin kurang ketajamannya karena jumlah informasi per piksel semakin menurun.

Ketika dilakukan *downsample* terhadap suatu gambar, informasi dari gambar asal harus dibuang agar gambar dapat diperkecil. Namun, jika dilakukan *downsample* lalu melakukan *upsample*, tidak akan didapatkan kembali seluruh detail gambar seperti aslinya.

Menurut Gonzalez and Woods (2010:65) *zoom* membutuhkan dua langkah: pembuatan lokasi piksel baru, dan menyimpan nilai abu-abu ke lokasi baru tersebut. Misalkan gambar berukuran  $500 \times 500$  piksel harus diperbesar 1,5 kali menjadi  $750 \times 750$  piksel. Cara sederhana untuk memvisualisasikan pembesaran adalah meletakkan  $750 \times 750$  piksel imajiner di gambar aslinya. Kemudian melakukan pengambilan nilai abu-abu untuk setiap piksel dengan mencari piksel terdekat pada gambar asli dan menetapkan nilai abu-abu ke piksel baru. Metode ini disebut interpolasi *nearest neighbor* atau interpolasi tetangga terdekat karena metode ini memberikan intensitas tetangganya yang terdekat ke setiap lokasi baru pada gambar asli. Metode ini paling sederhana, karena memanfaatkan teknik replikasi piksel. Gambar 11 dan Gambar 12 menunjukkan *downsampling* dan *upsampling* dengan metode *Nearest Neighbor*. Metode ini baik digunakan apabila warna piksel yang diinginkan tidak berubah, namun metode ini kurang baik jika digunakan untuk memperbesar gambar foto karena akan menimbulkan efek blok dan gambar yang kasar.



Gambar 11. *Downsampling* dengan metode *Nearest Neighbor*  
 Sumber (Qidwai & Chen, 2009:71)



Gambar 12. *Upsampling* dengan metode *Nearest Neighbor*  
 Sumber (Qidwai & Chen, 2009:71)

## H. Pengenalan Pola

Perkembangan teknologi saat ini akan didominasi oleh sistem dan mesin-mesin dengan kecerdasan buatan (*machine intelligence*). Teknik pengenalan pola merupakan salah satu komponen penting dari mesin atau sistem cerdas tersebut yang digunakan baik untuk mengolah data maupun dalam pengambilan keputusan.

Secara umum pengenalan pola (*pattern recognition*) adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif fitur (ciri) atau sifat utama dari suatu objek. Pola sendiri

adalah suatu entitas yang terdefinisi dan dapat diidentifikasi serta diberi nama. Sidik jari adalah suatu contoh pola. Pola bisa merupakan kumpulan hasil pengukuran atau pemantauan dan bisa dinyatakan dalam notasi vektor atau matriks (Putra, 2010:303).

Secara garis besar metode-metode pengenalan pola dapat dibagi menjadi tiga kelompok yaitu metode statistik, metode struktural dan metode jaringan syaraf tiruan. Metode statistik adalah pengenalan pola dengan mengukur jarak ciri fitur untuk kemudian diklasifikasikan pada tingkat kesamaan ciri. Metode struktural adalah pengenalan pola dengan mencari ciri khas/fitur yang unik dari suatu citra tertentu. Metode jaringan syaraf tiruan adalah pengenalan pola dengan melakukan proses pembelajaran atau pelatihan ciri fitur pada tiap masukan untuk kemudian dilakukan proses pengenalan.

## **I. Ekstraksi Ciri**

Fitur citra merupakan atribut dari suatu citra yang ditentukan oleh tampilan visual atau hasil manipulasi tertentu dari piksel citra. Ekstraksi fitur adalah prosedur menghasilkan deskripsi dari suatu obyek dalam hal parameter terukur yang mewakili sifat yang relevan dari obyek, dan dapat digunakan untuk pengelompokan dengan menetapkan objek ke kelas. Fitur citra yang digunakan untuk segmentasi adalah fitur warna dan tekstur, meskipun sifat dari fitur terpisah dimana tekstur menggunakan tingkat keabu-abuan citra, sedangkan warna mengekstraksi semua informasi pada ruang warna.

Secara umum, Zhao and Chellapa (2006:11) menjelaskan ada dua



macam ciri (*feature*) pada wajah, yaitu *holistic features* dan *facial feature*. *Facial feature* mendeskripsikan ciri dari warna, bentuk, letak hidung, mulut, mata, telinga, dan lain-lain. Sedangkan pada *holistic feature* setiap cirinya merupakan suatu karakteristik dari seluruh wajah. wajah dianggap sebagai kesatuan yang utuh. Untuk melakukan ekstaksi ciri pada wajah, Zhao dan Chellapa (2006:14) membedakan teknik pengenalan wajah menjadi tiga metode, sebagai berikut:

1. *Holistic matching methods*

Metode ini digunakan untuk menyelesaikan kasus *holistic features* yaitu mencari informasi secara keseluruhan pada wajah.

2. *Feature based (structural) matching methods*

Metode ini digunakan untuk menyelesaikan kasus *facial features* yaitu mencari informasi berdasarkan ciri yang ada pada wajah seperti hidung, mulut, mata, telinga dan lain-lain.

3. *Hybrid methods*

Metode ini merupakan penggabungan dari metode *holistic methods* dan *feature based*.

## **J. *Principal Component Analysis***

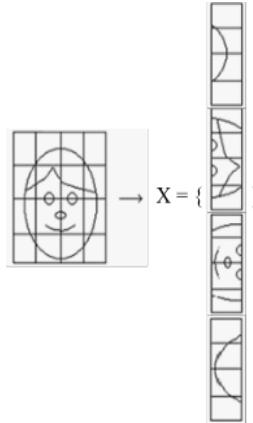
Dalam statistika, *Principal Component Analysis* (PCA) adalah teknik yang digunakan untuk menyederhanakan suatu data dengan cara mentransformasi linier sehingga terbentuk sistem koordinat baru dengan varians maksimum. PCA dapat digunakan untuk mereduksi dimensi suatu data tanpa mengurangi karakteristik data tersebut secara signifikan (Smith, 2002). Teknik ini yang diperkenalkan oleh Pearson (1901) dan Hotelling

(1933), telah menjadi teknik pemrosesan data dan pereduksi data yang populer, dengan berbagai aplikasi dalam bidang teknik, biologi, ekonomi dan ilmu sosial (Keho, 2012).

Prosedur PCA pada dasarnya bertujuan untuk menyederhanakan variabel yang diamati dengan cara menyusutkan (mereduksi) dimensinya. Hal ini dilakukan dengan cara menghilangkan korelasi diantara variabel bebas melalui transformasi variabel bebas asal ke variabel baru yang tidak berkorelasi sama sekali tanpa menghilangkan informasi penting yang ada di dalamnya atau yang biasa disebut dengan *principal component*. Dengan reduksi ini maka waktu komputasi dapat dikurangi dan kompleksitas dari citra wajah yang tidak perlu dapat dihilangkan. *Principal Component Analysis* menggunakan vektor-vektor yang disebut dengan vektor eigen dan nilai-nilai yang disebut dengan nilai eigen untuk mendapatkan fitur yang paling signifikan pada *database*.

### **Langkah-langkah komputasi PCA**

Misalkan  $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M$  adalah  $M \times 1$  vektor yang merupakan himpunan  $n$  subjek. Presentasi data dapat dilakukan dengan *lexicographical ordering* untuk setiap wajah yang akan dilatihkan dimana kolom yang satu diletakkan di samping kolom yang lain sehingga membentuk vektor wajah yang merupakan vektor kolom seperti ditunjukkan pada Gambar 13.



Gambar 13. Formasi vektor wajah  
(Sumber: Kaymak, 2003:4)

Langkah 1. Menghitung rata-rata data  $\Psi = \frac{1}{M} \sum_{k=1}^M \Gamma_k$

Langkah 2. Menormalisasi data

$$\Phi_k = \Gamma_k - \Psi ; A = [\Phi_1, \Phi_2, \Phi_3, \dots, \Phi_M]$$

Langkah 3. Menghitung matriks kovarian  $C = \frac{1}{M} \sum_{k=1}^M \Phi_k \Phi_k^T = AA^T$

Langkah 4. Menghitung nilai eigen dari  $C$ :  $\lambda_1 > \lambda_2 > \dots > \lambda_M$

Langkah 5. Menghitung vektor eigen dari  $C$ :  $v_1, v_2, \dots, v_M$

Nilai eigen dapat diurutkan mulai yang terbesar sampai yang terkecil, dan vektor eigen yang bersesuaian dengan nilai eigen tersebut juga diurutkan.

Dalam *pattern recognition*, fitur-fitur suatu citra menggunakan PCA direpresentasikan sebagai vektor eigen. Vektor eigen didefinisikan sebagai kumpulan hubungan karakteristik-karakteristik dari suatu citra untuk mengenali citra tersebut secara spesifik.

## K. Eigenface

*Eigenface* adalah salah satu algoritma pengenalan pola wajah berdasarkan pada *Principal Component Analysis* (PCA) yang dikembangkan di MIT. Dalam fase pengenalan, *eigenface* mereduksi dimensi dari *input* gambar dengan memproyeksikannya ke dalam subruang yang ditemukan selama pelatihan. Memproyeksikan ke dalam subruang berarti menemukan titik terdekat dari gambar pelatihan. Algoritma *eigenface* secara keseluruhan cukup sederhana. *Training image* direpresentasikan dalam sebuah *vektor flat* (gabungan vektor) dan digabung menjadi sebuah matriks tunggal. Vektor eigen kemudian diekstraksi dan disimpan dalam file *temporary* atau *database*. *Training image* kemudian diproyeksikan dalam *feature space*, bernama *face space* yang ditentukan oleh vektor eigen. Vektor eigen ( $v_i$ ) dan nilai eigen ( $\lambda_i$ ) dari matrik kovarian adalah

$$Cv_i = \lambda_1 v_1 \quad (2.6)$$

$$Cv_i = \lambda_i I v_i \quad (2.7)$$

Solusi menentukan nilai eigen

$$\det(C - \lambda_i I) = 0 \quad (2.8)$$

Solusi menentukan vektor eigen

$$(C - \lambda_i I)v_i = 0 \quad (2.9)$$

Menentukan *eigenface*

$$\mu_i = \sum_{k=1}^M v_{ik} \Phi_k \quad (2.10)$$

Wajah diklasifikasikan dengan memproyeksikan wajah  $\Gamma$  ke ruang wajah sebagai berikut:

$$\omega_k = \mu_k^T (\Gamma_k - \Psi) \quad (2.11)$$

Bobot membentuk vektor  $\Omega_k^T = [\omega_1, \omega_2, \dots, \omega_M]$ , yang berisi proyeksi ke setiap vektor eigen. Klasifikasi dilakukan dengan menghitung jarak  $\Omega_k$  dari  $\Omega$ , di mana  $\Omega$  merupakan vektor bobot yang mendefinisikan beberapa kelas (Shih, 2010:290).

#### L. *Euclidean Distance*

Salah satu metode klasifikasi yang paling banyak digunakan adalah metode *euclidean distance*. Metode ini dilakukan dengan menghitung akar dari kuadrat perbedaan antara koordinat dari sepasang objek. Semakin kecil nilai jarak yang dihasilkan maka data tersebut dapat diklasifikasi menjadi 1 kelompok. Metode ini dapat dituliskan secara matematis pada Persamaan 2.12.

$$\varepsilon_k^2 = \|\Omega_{test} - \Omega_k\|^2 \quad (2.12)$$

$$k = [1, 2, \dots, M]$$

## M. Contoh Algoritma PCA

1. Representasi matriks pelatihan dengan ukuran  $2 \times 2$  dimensi dua diubah menjadi matriks  $2^2 \times 1$  dimensi satu

$$\text{Matriks } A (\Gamma_1) = \begin{bmatrix} 3 & 5 \\ 7 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 3 \\ 5 \\ 7 \\ 8 \end{bmatrix}$$

$$\text{Matriks } B (\Gamma_2) = \begin{bmatrix} 2 & 2 \\ 5 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 2 \\ 2 \\ 5 \\ 8 \end{bmatrix}$$

$$\text{Matriks } C (\Gamma_3) = \begin{bmatrix} 2 & 7 \\ 7 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 2 \\ 7 \\ 7 \\ 8 \end{bmatrix}$$

Kemudian hasil tersebut dimasukkan dalam satu matriks

$$\Gamma = [\Gamma_1, \Gamma_2, \dots, \Gamma_M]$$

$$\Gamma = \begin{bmatrix} 3 & 2 & 2 \\ 5 & 2 & 7 \\ 7 & 5 & 7 \\ 8 & 8 & 8 \end{bmatrix}$$

2. Menghitung rata-rata data ke arah baris

$$\Psi = \frac{1}{M} \sum_{k=1}^M \Gamma_k$$

$$\Psi_1 = \frac{3 + 2 + 2}{3} = 2,3333$$

$$\Psi_2 = \frac{5 + 2 + 7}{3} = 4,6667$$

$$\Psi_3 = \frac{7 + 5 + 7}{3} = 6,3333$$

$$\Psi_4 = \frac{8 + 8 + 8}{3} = 8,000$$

Dikelompokkan menjadi satu matriks

$$\Psi = \begin{bmatrix} 2,3333 \\ 4,6667 \\ 6,3333 \\ 8 \end{bmatrix}$$

3. Menormalisasikan data

$$\Phi_k = \Gamma_k - \Psi$$

$$\Phi_1 = \begin{bmatrix} 3 \\ 5 \\ 7 \\ 8 \end{bmatrix} - \begin{bmatrix} 2,3333 \\ 4,6667 \\ 6,3333 \\ 8 \end{bmatrix} = \begin{bmatrix} 0,6667 \\ 0,3333 \\ 0,6667 \\ 0 \end{bmatrix}$$

$$\Phi_2 = \begin{bmatrix} 2 \\ 2 \\ 5 \\ 8 \end{bmatrix} - \begin{bmatrix} 2,3333 \\ 4,6667 \\ 6,3333 \\ 8 \end{bmatrix} = \begin{bmatrix} -0,3333 \\ -2,6667 \\ -1,3333 \\ 0 \end{bmatrix}$$

$$\Phi_3 = \begin{bmatrix} 2 \\ 7 \\ 7 \\ 8 \end{bmatrix} - \begin{bmatrix} 2,3333 \\ 4,6667 \\ 6,3333 \\ 8 \end{bmatrix} = \begin{bmatrix} -0,3333 \\ 2,3333 \\ 0,6667 \\ 0 \end{bmatrix}$$

Dikelompokkan menjadi satu matriks A

$$A = [\Phi_1, \Phi_2, \dots, \Phi_M]$$

$$A = \begin{bmatrix} 0,6667 & -0,3333 & -0,3333 \\ 0,3333 & -2,6667 & 2,3333 \\ 0,6667 & -1,3333 & 0,6667 \\ 0 & 0 & 0 \end{bmatrix}$$

4. Kovarian matriks dari matriks A

$$C = A^T A$$

$$C = \begin{bmatrix} 0,6667 & 0,3333 & 0,66667 & 0 \\ -0,3333 & -2,6667 & -1,3333 & 0 \\ -0,3333 & 2,3333 & 0,6667 & 0 \end{bmatrix} \times \begin{bmatrix} 0,6667 & -0,3333 & -0,3333 \\ 0,3333 & -2,6667 & 2,3333 \\ 0,6667 & -1,3333 & 0,6667 \\ 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 9 & -7 \\ 1 & -7 & 6 \end{bmatrix}$$

5. Menentukan nilai eigen dan vektor eigen dari matriks C

$$\det(C - \lambda_i I) = 0$$

$$\text{Det} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 9 & -7 \\ 1 & -7 & 6 \end{bmatrix} - \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

$$\text{Det} \begin{bmatrix} 1 - \lambda & -2 & 1 \\ -2 & 9 - \lambda & -7 \\ 1 & -7 & 6 - \lambda \end{bmatrix} = 0$$

$$(1 - \lambda \begin{vmatrix} 9 - \lambda & -7 \\ -7 & 6 - \lambda \end{vmatrix}) - (-2 \begin{vmatrix} -2 & -7 \\ 1 & 6 - \lambda \end{vmatrix}) + (1 \begin{vmatrix} -2 & 9 - \lambda \\ 1 & -7 \end{vmatrix}) = 0$$

$$(-\lambda^3 + 16\lambda^2 - 20\lambda + 5) + (4\lambda - 10) + (5 + \lambda) = 0$$

$$-\lambda^3 + 16\lambda^2 - 15\lambda = 0$$

$$-\lambda(\lambda^2 - 16\lambda + 15) = 0$$

$$-\lambda(\lambda - 1)(\lambda - 15) = 0$$

$$\lambda_1 = 0, \lambda_2 = 1, \lambda_3 = 15$$

Dari matriks kovarian C, didapatkan 3 nilai eigen  $\lambda$  yaitu 0, 1, dan 15. Nilai eigen tersebut digunakan untuk mencari vektor eigen yang berhubungan



dengan tiap nilai eigennya.

$$(C - \lambda_i I)v_i = 0$$

➤ Untuk  $\lambda_1 = 0$

$$\begin{bmatrix} 1-0 & -2 & 1 \\ -2 & 9-0 & -7 \\ 1 & -7 & 6-0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

Matriks di atas dikalikan dan menghasilkan persamaan

$$v_1 - 2v_2 + v_3 = 0 \quad (1)$$

$$-2v_1 + 9v_2 - 7v_3 = 0 \quad (2)$$

$$v_1 - 7v_2 + 6v_3 = 0 \quad (3)$$

Dari persamaan (1) didapat

$$v_1 = 2v_2 - v_3 \quad (4)$$

Substitusi persamaan (4) ke persamaan (3) menghasilkan

$$2v_2 - v_3 - 7v_2 + 6v_3 = 0$$

$$-5v_2 + 5v_3 = 0$$

$$v_2 = v_3$$

Sehingga

$$v_1 = v_2 = v_3$$

Normalisasi

$$\underline{v_1^*} = \frac{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}}{\left( [v_1 \ v_2 \ v_3] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \right)^{\frac{1}{2}}}$$

$$\underline{v_1^*} = \frac{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}}{\left( [v_1 \ v_2 \ v_3] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \right)^{\frac{1}{2}}}$$

$$\underline{v_1^*} = \frac{\begin{bmatrix} v_1 \\ v_1 \\ v_1 \end{bmatrix}}{(3v_1^2)^{\frac{1}{2}}} = \frac{\begin{bmatrix} 1 \\ \sqrt{3} \\ 1 \end{bmatrix}}{\sqrt{3}} = \begin{bmatrix} 0.5774 \\ 0.5774 \\ 0.5774 \end{bmatrix}$$

➤ Untuk  $\lambda_2 = 1$

$$\begin{bmatrix} 1-1 & -2 & 1 \\ -2 & 9-1 & -7 \\ 1 & -7 & 6-1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

Matriks di atas dikalikan dan menghasilkan persamaan

$$-2v_2 + v_3 = 0 \tag{5}$$

$$-2v_1 + 8v_2 - 7v_3 = 0 \tag{6}$$

$$v_1 - 7v_2 + 5v_3 = 0 \tag{7}$$

Dari persamaan 5 didapat

$$v_3 = 2v_2 \tag{8}$$

Substitusi persamaan (8) ke persamaan (6) menghasilkan

$$-2v_1 + 8v_2 - 14v_2 = 0$$

$$v_1 = -3v_2$$

Normalisasi

$$\underline{v_2^*} = \frac{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}}{\left( [v_1 \quad v_2 \quad v_3] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \right)^{\frac{1}{2}}}$$

$$\underline{v_2^*} = \frac{\begin{bmatrix} -3v_2 \\ v_2 \\ 2v_2 \end{bmatrix}}{\left( [-3v_2 \quad v_2 \quad 2v_2] \begin{bmatrix} -3v_2 \\ v_2 \\ 2v_2 \end{bmatrix} \right)^{\frac{1}{2}}}$$

$$\underline{v_2^*} = \frac{\begin{bmatrix} -3v_2 \\ v_2 \\ 2v_2 \end{bmatrix}}{(14v_2^2)^{\frac{1}{2}}} = \begin{bmatrix} -\frac{3}{\sqrt{14}} \\ \frac{1}{\sqrt{14}} \\ \frac{2}{\sqrt{14}} \end{bmatrix} = \begin{bmatrix} -0.8018 \\ 0.2673 \\ 0.5345 \end{bmatrix}$$

➤ Untuk  $\lambda_3 = 15$

Dengan cara yang sama seperti di atas vektor eigen untuk  $\lambda_3 = 15$  adalah

$$\underline{v_3^*} = \begin{bmatrix} 0.1543 \\ -0.7715 \\ 0.6172 \end{bmatrix}$$

Kemudian nilai tiap vector eigen dimasukan berdasarkan urutan dalam satu matriks.

$$\lambda_1 = 0 \quad \lambda_2 = 1 \quad \lambda_3 = 15$$

$$v = \begin{bmatrix} 0,5774 & -0,8018 & 0,1543 \\ 0,5774 & 0,2673 & -0,7715 \\ 0,5774 & 0,5345 & 0,6172 \end{bmatrix}$$

Nilai eigen  $\lambda_1 = 0$  diabaikan untuk perhitungan selanjutnya kemudian nilai eigen diurutkan dari yang terbesar hingga yang terkecil, sehingga didapat vektor eigen sebagai berikut

$$v = \begin{bmatrix} 0,1543 & -0,8018 \\ -0,7715 & 0,2673 \\ 0,6172 & 0,5345 \end{bmatrix}$$

## 6. Eigenfaces

$$\mu_i = Av_i$$

$$\mu_1 = \begin{bmatrix} 0,6667 & -0,3333 & -0,3333 \\ 0,3333 & -2,6667 & 2,3333 \\ 0,6667 & -1,3333 & 0,6667 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0,1543 \\ -0,7715 \\ 0,6172 \end{bmatrix} = \begin{bmatrix} 0,1543 \\ 3,5490 \\ 1,5430 \\ 0 \end{bmatrix}$$

$$\mu_2 = \begin{bmatrix} 0,6667 & -0,3333 & -0,3333 \\ 0,3333 & -2,6667 & 2,3333 \\ 0,6667 & -1,3333 & 0,6667 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} -0,8018 \\ 0,2673 \\ 0,5345 \end{bmatrix} = \begin{bmatrix} -0,8018 \\ 0,2673 \\ -0,5345 \\ 0 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 0,1543 & -0,8018 \\ 3,5490 & 0,2673 \\ 1,5430 & -0,5345 \\ 0 & 0 \end{bmatrix}$$

## 7. Memproyeksikan A ke dalam Eigenspace

$$\omega_1 = \begin{bmatrix} 0,1543 & 3,5490 & 1,5430 & 0 \\ -0,8018 & 0,2673 & -0,5345 & 0 \end{bmatrix} \begin{bmatrix} 0,6667 \\ 0,3333 \\ 0,6667 \\ 0 \end{bmatrix} = \begin{bmatrix} 2,3146 \\ -0,8018 \end{bmatrix}$$

$$\omega_2 = \begin{bmatrix} 0,1543 & 3,5490 & 1,5430 & 0 \\ -0,8018 & 0,2673 & -0,5345 & 0 \end{bmatrix} \begin{bmatrix} -0,3333 \\ -2,6667 \\ -1,3333 \\ 0 \end{bmatrix} = \begin{bmatrix} -11,5728 \\ 0,2673 \end{bmatrix}$$

$$\omega_2 = \begin{bmatrix} 0,1543 & 3,5490 & 1,5430 & 0 \\ -0,8018 & 0,2673 & -0,5345 & 0 \end{bmatrix} \begin{bmatrix} -0,3333 \\ 2,3333 \\ 0,6667 \\ 0 \end{bmatrix} = \begin{bmatrix} 9,2582 \\ 0,5345 \end{bmatrix}$$

$$\Omega = \begin{bmatrix} 2,3146 & -11,5728 & 9,2582 \\ -0,8018 & 0,2673 & 0,5345 \end{bmatrix}$$

### Contoh Pengujian Algoritma PCA

1. Representasi matriks pengujian dengan ukuran  $2 \times 2$  dimensi dua diubah menjadi matriks  $2^2 \times 1$  satu dimensi

$$\text{matriks } D (\Gamma_{test}) = \begin{bmatrix} 2 & 4 \\ 5 & 9 \end{bmatrix} \rightarrow \begin{bmatrix} 2 \\ 4 \\ 5 \\ 9 \end{bmatrix}$$

2. Mengurangi nilai matriks pengujian dengan matriks rata-rata dari data pelatihan

$$A_{test} = \Gamma_{test} - \Psi$$

$$A_{test} = \begin{bmatrix} 2 \\ 4 \\ 5 \\ 9 \end{bmatrix} - \begin{bmatrix} 2,3333 \\ 4,6667 \\ 6,3333 \\ 8 \end{bmatrix} = \begin{bmatrix} -0,3333 \\ -0,6667 \\ -1,3333 \\ 1 \end{bmatrix}$$

3. Memproyeksikan matriks pengujian dalam *eigen space*

$$\Omega_{test} = \begin{bmatrix} 0,1543 & 3,5490 & 1,5430 & 0 \\ -0,8018 & 0,2673 & -0,5345 & 0 \end{bmatrix} \begin{bmatrix} -0,3333 \\ -0,6667 \\ -1,3333 \\ 1 \end{bmatrix} = \begin{bmatrix} -4,4748 \\ 0,8018 \end{bmatrix}$$

4. Menentukan jarak antar vektor setiap citra dengan *Euclidean distance*

$$\varepsilon_1 = \sqrt{(-4,4748 - 2,3146)^2 + (0,8018 + 0,8018)^2} = 6,9761$$

$$\varepsilon_2 = \sqrt{(-4,4748 + 11,5728)^2 + (0,8018 - 0,2673)^2} = 7,1181$$

$$\varepsilon_3 = \sqrt{(-4,4748 - 9,2582)^2 + (0,8018 - 0,5345)^2} = 13,7356$$

Dari perhitungan jarak antara bobot matriks pengujian ( $D$ ) dan matriks pelatihan jarak terdekat yaitu 6,7961. Sehingga dapat disimpulkan matriks pengujian ( $D$ ) mirip dengan matriks pelatihan indeks ke-1 yaitu matriks ( $A$ ).

## **BAB III**

### **METODE PENELITIAN**

#### **A. Waktu dan Tempat Penelitian**

Penelitian ini dilakukan pada bulan Januari 2019 sampai dengan bulan Juni 2019. Penelitian dilakukan di Laboratorium Elektronika dan Instrumentasi, Jurusan Pendidikan Fisika, FMIPA UNY.

#### **B. Variabel Penelitian**

##### 1. Variabel bebas

Variabel bebas adalah variabel yang menjadi sebab berubahnya suatu variabel lain yaitu variabel terikat. Variabel bebas dalam penelitian ini adalah wajah manusia.

##### 2. Variabel terikat

Variabel terikat adalah variabel yang dipengaruhi oleh akibat adanya variabel lain yaitu variabel bebas. Variabel terikat dalam penelitian ini adalah identifikasi wajah manusia sesuai *database* yang ada.

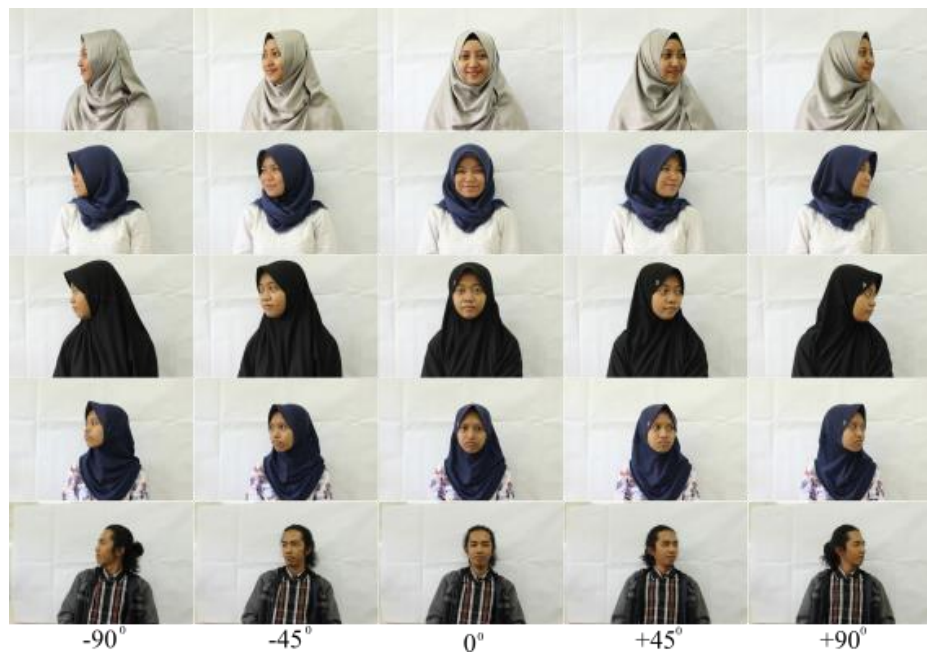
##### 3. Variabel Kontrol

Variabel kontrol adalah variabel yang dibuat sama sehingga tidak mempengaruhi variabel terikat. Variabel kontrol dalam penelitian ini adalah intensitas cahaya ketika pengambilan gambar dengan cara mengatur ISO kamera dan sudut pandang (*angle*) saat pengambilan gambar.

### C. Teknik Pengumpulan Data

Sampel penelitian merupakan citra wajah manusia berukuran 3456 x 5184 piksel, berformat file JPG dan merupakan citra RGB. Pada pengambilan data dilakukan variasi pose wajah dan ekspresi wajah yang berbeda. Hal tersebut dilakukan untuk data pelatihan dan meningkatkan tingkat akurasi.

Pengambilan data wajah dilakukan pada tanggal 1 April 2019 dengan sampel sebanyak 5 orang. Masing-masing individu mempunyai 11 citra wajah yang terdiri dari lima pose ( $-90^{\circ}$ ,  $-45^{\circ}$ ,  $0^{\circ}$ ,  $+45^{\circ}$ , dan  $+90^{\circ}$ ) seperti yang ditunjukkan pada Gambar 14; dan enam ekspresi yang berbeda seperti yang ditunjukkan pada Gambar 15.



Gambar 14. Data citra wajah dengan lima pose





Gambar 15. Data citra wajah dengan enam ekspresi wajah

#### D. Instrumen penelitian

Perangkat keras yang digunakan adalah kamera digital dan satu perangkat notebook Lenovo ThinkPad X220 dengan spesifikasi *processor* Intel(R) Core(TM) i5, RAM 8 GB. Adapun perangkat lunak yang digunakan adalah MATLAB R2013a.

#### E. Prosedur Penelitian dan Teknik Analisis

##### 1. Deteksi wajah

Penelitian ini menggunakan metode model warna YCbCr untuk segmentasi citra wajah. Model warna YCbCr dikenal mampu memisahkan antara komponen yang mengandung warna kulit dengan komponen warna lainnya. Citra yang telah dipisahkan antara komponen warna kulit dengan komponen lainnya dilakukan *mathematical morphology* untuk menghilangkan derau. *Mathematical morphology* yang digunakan antara

lain erosi dan dilasi. Setelah didapat bagian wajah, kemudian bagian wajah tersebut dicrop dan citra ditransformasi kembali ke model warna RGB.

## 2. *Pre-Processing*

### a. Konversi citra RGB ke citra *grayscale*

Mengubah citra RGB menjadi citra *grayscale* dengan 256 tingkat keabu-abuan. Hal ini dapat dilakukan dengan cara mengeliminasi informasi gradasi warna (*hue*) dan kejenuhan warna (*saturation*) dengan tetap mempertahankan tingkat kecerahan (*luminance*), sehingga dihasilkan matrik  $M \times N$  warna hitam-putih dalam 256 tingkat keabu-abuan. Hal ini dilakukan karena dengan citra hitam-putih, proses perhitungan menjadi jauh lebih sederhana.

### b. *Resize*

*Resize* adalah mengubah ukuran piksel citra. Setiap citra hasil *cropping* deteksi wajah memiliki ukuran piksel yang berbeda-beda, maka diperlukan penyeragaman ukuran citra dengan metode Interpolasi *Nearest Neighbor* (INN). Interpolasi *nearest neighbor* merupakan algoritma interpolasi citra yang memanfaatkan teknik replikasi piksel. Masing-masing citra diperkecil resolusinya menjadi 120 x 170 piksel. Memperkecil ukuran citra artinya menampilkan suatu citra dengan jumlah pixel yang lebih sedikit dengan proses *sampling* piksel berjarak.

## 3. Pengenalan Pola

Tahapan pengenalan pola menggunakan algoritma *eigenface* yang memanfaatkan *Principal Component Analysis* (PCA) untuk mereduksi

dimensi citra guna menemukan vektor-vektor yang mempunyai nilai terbaik untuk distribusi citra wajah di dalam ruang citra masukan. Pada tahapan ini akan dianalisis proses pada sistem pengenalan wajah. Analisis ini bertujuan untuk mengkaji dan menguraikan proses yang terdapat pada sistem pengenalan wajah. Pada proses pengenalan wajah terdapat dua proses, yaitu

a. Proses Pelatihan

Proses pelatihan yaitu proses melatih data yang akan disimpan dalam *database*.

b. Proses pengenalan

Proses pengenalan yaitu proses pencocokan data uji dengan data latih, dengan cara menghitung perbandingan ciri masing-masing data.

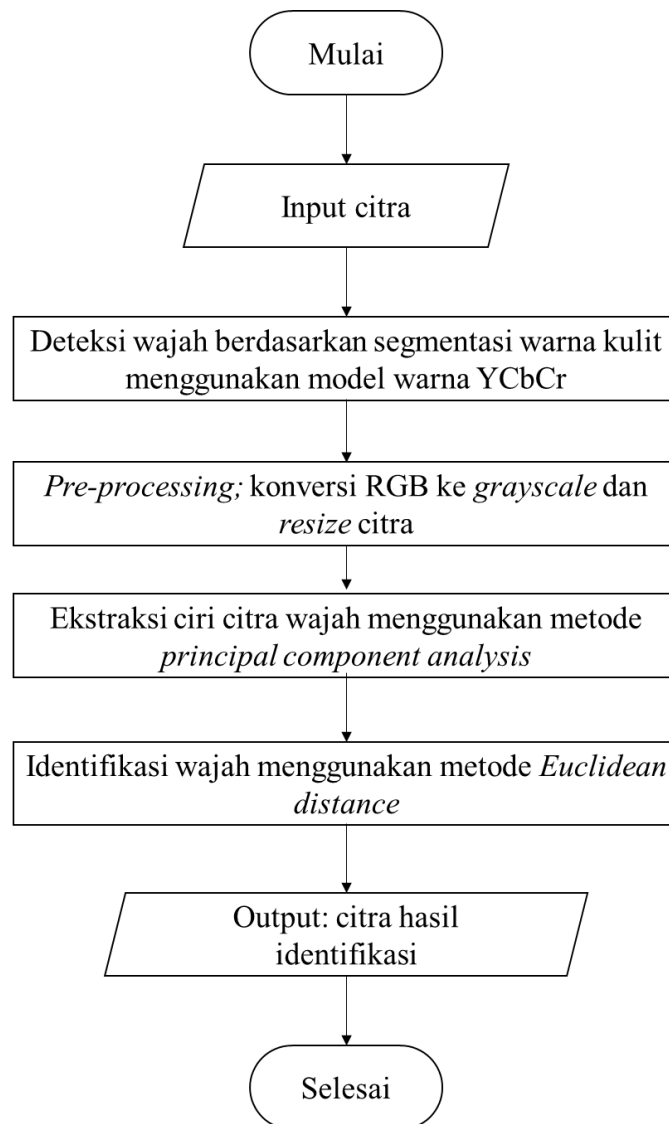
4. Klasifikasi

Klasifikasi dilakukan untuk mendapatkan region-region homogen yang tergabung dalam suatu kelas (*class*) tertentu. Hasil klasifikasi berupa region-region yang mempunyai label kelas tertentu. Pada proses pengenalan pola akan dilakukan penghitungan jarak menggunakan metode *euclidean distance*. Pengukuran jarak dilakukan dengan mengukur kemiripan dua vektor ciri citra antara pelatihan dan pengujian. Semakin kecil jarak, maka semakin mirip kedua vektor yang dicocokkan/dibandingkan. Sebaliknya semakin besar jarak, maka semakin berbeda kedua vektor yang dicocokkan/dibandingkan. Citra

pengujian yang telah melewati proses klasifikasi dan pengambilan keputusan akan diidentifikasi sebagai pola wajah tertentu.

## F. Diagram Alir

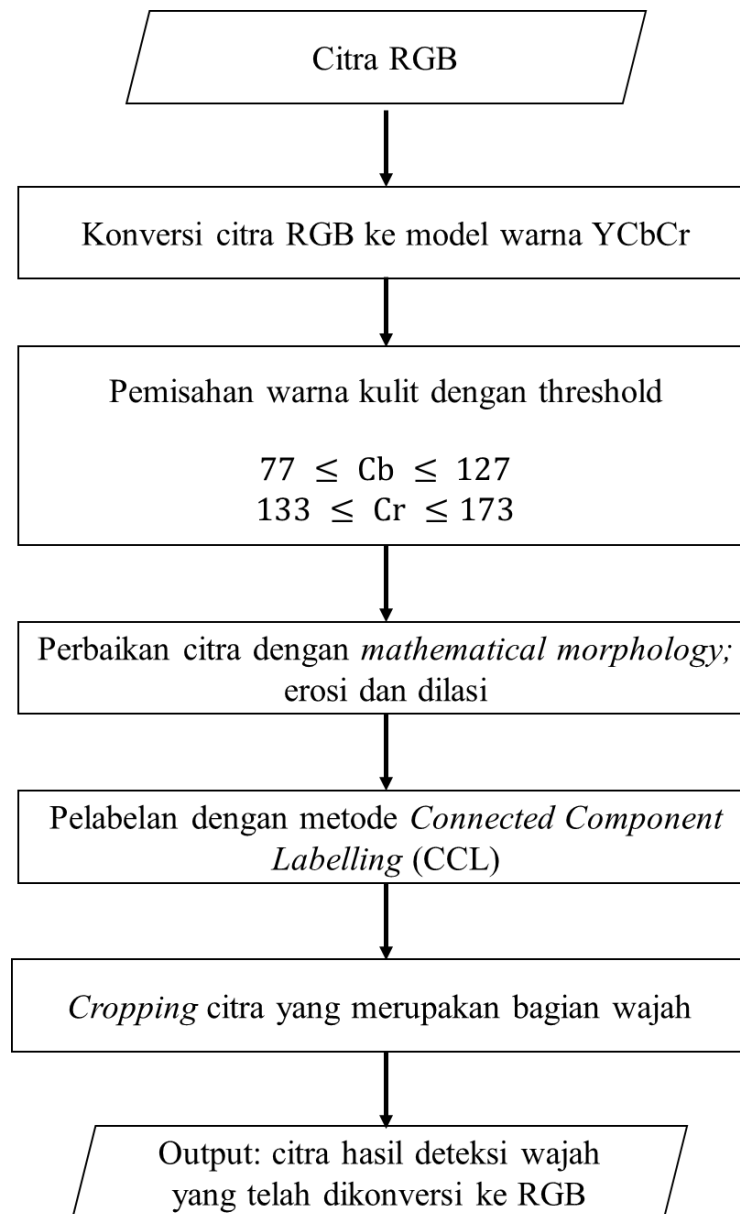
Penelitian ini bertujuan untuk mengidentifikasi citra wajah berdasarkan *database* yang tersimpan. Gambar 16 merupakan diagram alir keseluruhan dalam sistem pengenalan wajah.



Gambar 16. Diagram alir pengenalan wajah

1. Diagram alir deteksi wajah berdasarkan model warna YCbCr

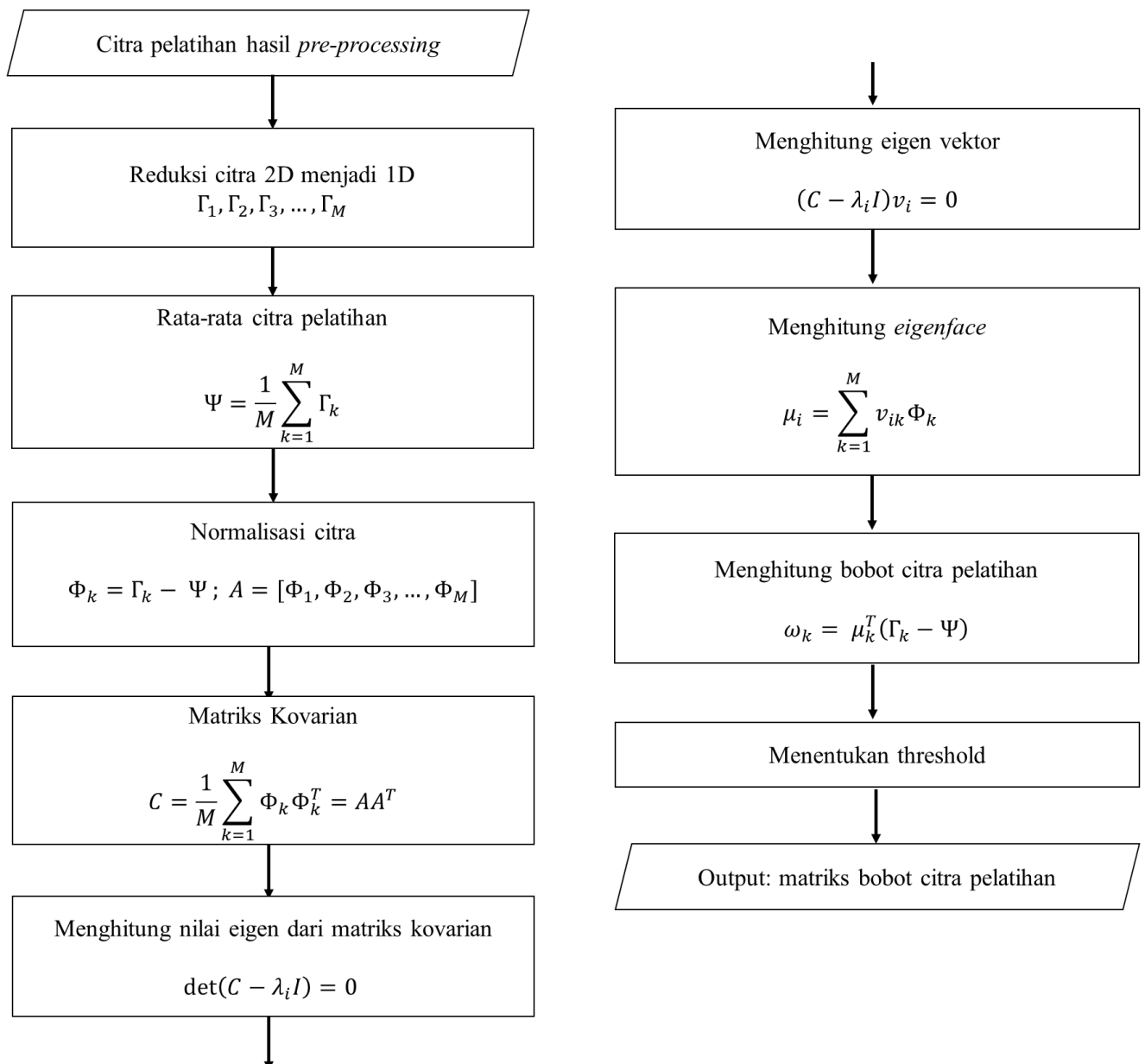
Tahap awal pada penelitian ini adalah deteksi wajah pada citra RGB. Deteksi wajah bertujuan untuk menentukan wilayah wajah pada citra yang telah diinput. Gambar 17 menunjukkan diagram alir deteksi wajah menggunakan *skin detecton* berdasarkan model warna YCbCr



Gambar 17. Diagram alir deteksi wajah

2. Diagram alir ekstraksi dengan metode *principal component analysis*

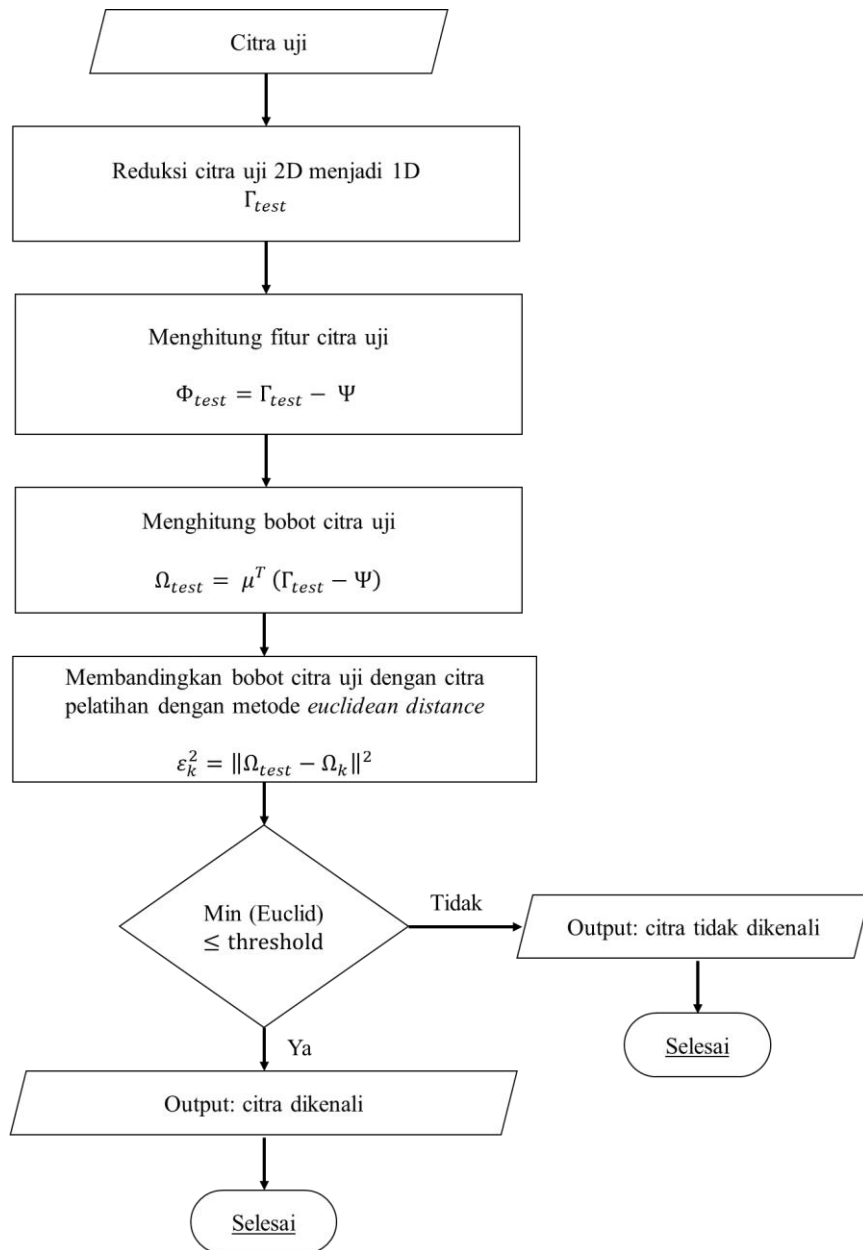
Ekstraksi ciri bertujuan untuk mendapatkan ciri dari masing-masing citra pada *database* menggunakan *principal component analysis*. Gambar 18 merupakan diagram alir ekstraksi ciri



Gambar 18. Diagram alir ekstraksi ciri

3. Diagram alir identifikasi wajah berdasarkan jarak dengan metode *euclidean distance*

Identifikasi wajah bertujuan untuk mengenali citra uji terhadap *database* yang tersimpan, dengan cara membandingkan bobot citra uji dengan bobot database. Gambar 19 menunjukkan diagram alir indetifikasi wajah dengan metode *euclidean distance*



Gambar 19. Diagram alir identifikasi wajah

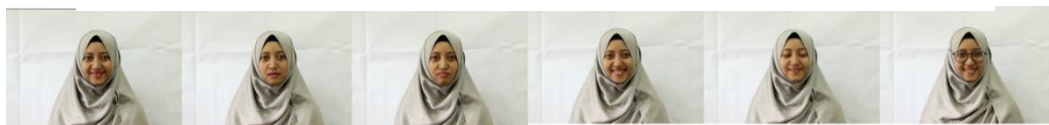
## BABABAB IV

### HASIL DAN PEMBAHASAN

Pada penelitian ini, dibuat suatu sistem untuk mengenali wajah dengan metode *Principal Component Analysis*. Sedangkan untuk mendeteksi wajah digunakan metode *skin detection* dengan menentukan jangkauan Y, Cb, Cr. Citra wajah sebanyak 55 *image* wajah dari 5 orang yang diperoleh dengan menggunakan kamera digital disimpan di dalam *database* yang dibagi menjadi dua bagian yaitu *database* pelatihan dan *database* pengujian. Masing-masing individu mempunyai 11 *image* wajah yang terdiri dari berbagai pose dan ekspresi. Gambar 20 dan Gambar 21 adalah salah satu contoh hasil pengambilan data citra wajah dengan masing-masing citra berukuran 3456 x 5184 piksel:



Gambar 20. Contoh *database* dengan 5 pose yang berbeda



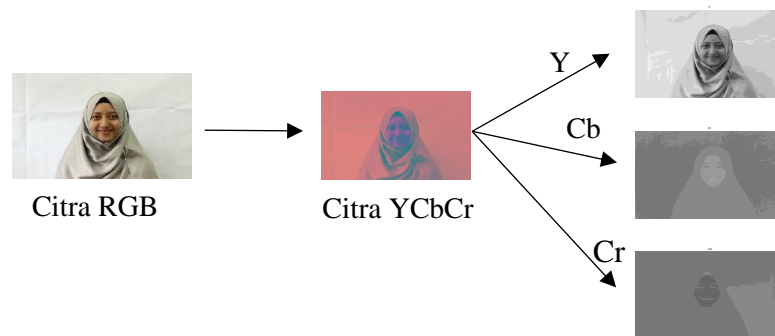
Gambar 21. Contoh *database* dengan 6 ekspresi wajah yang berbeda

#### A. Deteksi Citra Wajah

Pada penelitian ini untuk mendeteksi citra wajah digunakan metode segmentasi warna kulit. Segmentasi warna kulit adalah proses pemisahan antara objek (*foreground*) yaitu kulit dengan *background*. Metode ini akan memisahkan nilai RGB menjadi nilai *luminance* dan *chrominance*. Ruang



warna RGB pada citra asli masih mengandung efek cahaya yang menyebabkan karakteristik warna kulit bisa berubah, karenanya perlu dikonversi ke dalam bentuk warna kromatik. Warna kromatik dikenal juga sebagai warna asli tanpa adanya pencahayaan. Untuk mengurangi efek pencahayaan itu digunakan model warna YCbCr karena pada model warna ini semua efek pencahayaan berada dalam komponen Y, model warna ini terdiri dari 3 komponen yang ditunjukkan pada Gambar 22 yaitu : Y bernilai *luminance* (tingkat keterangan), Cb bernilai *Chrominance Blue* (tingkat kebiruan) dan Cr bernilai *Chrominance Red* (tingkat kemerahan).



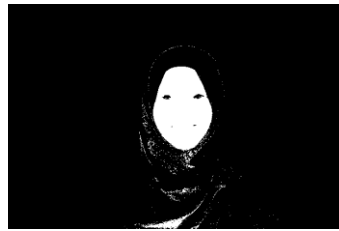
Gambar 22. Komponen Citra YCbCr

Segmentasi warna kulit hanya dilakukan pada komponen *chrominance blue* (Cb) dan *chrominance red* (Cr) karena komponen Y (*luminance*) mengandung pencahayaan sehingga diabaikan. Menurut Chai and Ngan (1999 : 555) rentang Cb dan Cr paling representatif untuk peta warna kulit adalah:

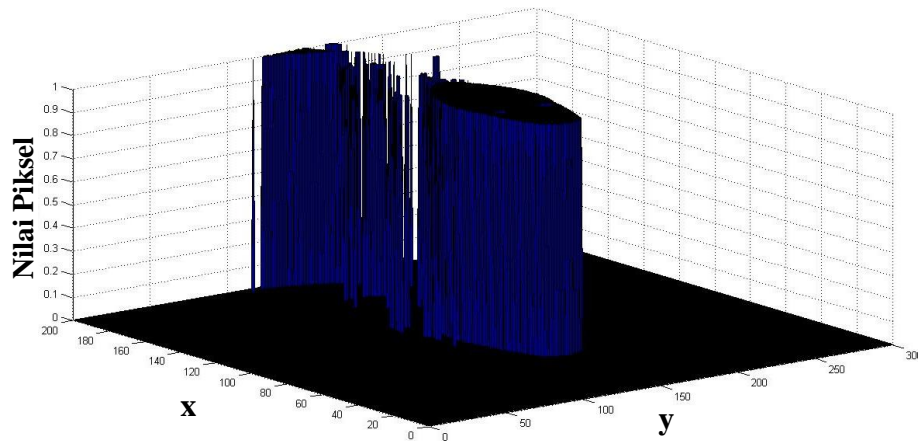
$$\text{warna kulit} = \begin{cases} 1, & \begin{cases} 77 \leq \text{Cb} \leq 127 \\ 133 \leq \text{Cr} \leq 173 \end{cases} \\ 0 & \text{bukan kulit} \end{cases}$$

Proses deteksi kulit dengan memisahkan antara kulit dan bukan kulit biasa disebut proses *binary image*. Gambar 23 menunjukkan kemungkinan warna

kulit yang dikategorikan dengan warna putih (1), dan bukan kulit dikategorikan warna hitam (0). Gambar 24 menunjukkan grafik distribusi kemungkinan kulit pada sampel citra:



Gambar 23. Citra *skin likelihood*

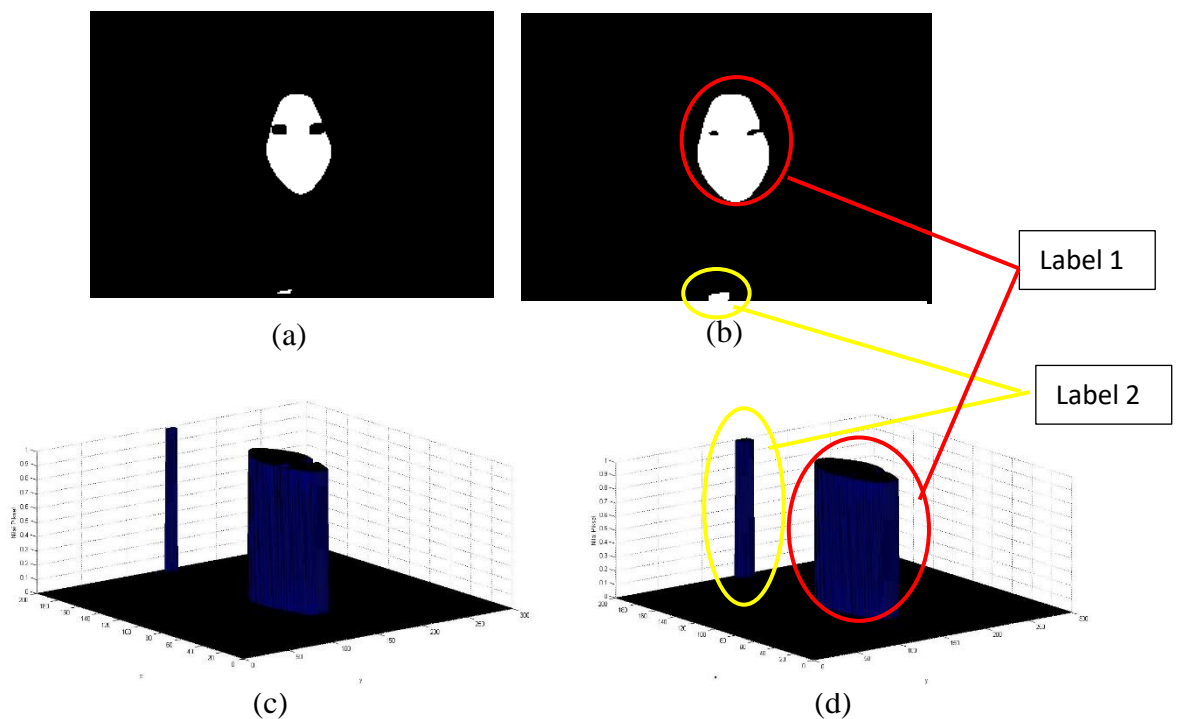


Gambar 24. Grafik distribusi kemungkinan kulit pada sampel citra

Gambar 23 hasil segmentasi kemungkinan kulit menunjukkan bahwa daerah terdeteksi belum tentu sesuai dengan kulit. Hanya dapat disimpulkan bahwa wilayah terdeteksi memiliki warna sama dengan kulit. Maka dari itu dibutuhkan perbaikan citra dengan teknik morfologi. Teknik morfologi bertujuan untuk memperbaiki hasil segmentasi dengan menggabungkan titik-titik latar yang ada di dalam obyek menjadi bagian dari obyek. Teknik morfologi yang digunakan dalam penelitian ini adalah teknik *opening*. Teknik *opening* merupakan proses erosi yang diikuti dengan dilasi; efek yang

dihasilkan adalah menghilangnya objek-objek kecil, dan men-*smooth*-kan batas dari objek besar tanpa mengubah area objek secara signifikan. Teknik *opening* menggunakan struktur elemen atau kernel berupa matrik bujur sangkar  $5 \times 5$ , sebagai berikut

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$



Gambar 25. Morfologi *Opening*. (a) citra erosi (b) citra dilasi (c) grafik erosi (d) grafik dilasi

Citra erosi pada Gambar 25 (a) memperlihatkan tepi objek mengecil atau mengikis karena titik objek berwarna putih (1) yang bertetangga dengan titik latar berwarna hitam (0) menjadi titik latar berwarna hitam (0). Sebaliknya citra dilasi pada Gambar 25 (b) menunjukkan objek menjadi lebih luas karena titik

latar berwarna hitam (0) yang bertetangga dengan titik objek berwarna putih (1) menjadi titik objek berwarna putih (1).

Gambar 25 (b) merupakan hasil morfologi *opening*, dimana piksel di dalam gambar tersebut direpresentasikan seperti tercantum pada Gambar 26

0	0	1	1	1	1	0	0
0	1	1	1	1	1	1	0
0	1	1	0	1	0	1	0
0	1	1	1	1	1	1	0
0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	0
0	1	1	0	0	0	0	0

Gambar 26. Representasi citra morfologi *opening*

Dalam menentukan wilayah wajah diperlukan pelabelan yang bertujuan untuk memisahkan beberapa luas wilayah piksel yang tidak saling berhubungan. Pelabelan menggunakan metode *Connected Component Labelling* (CCL) dengan konfigurasi 4 konektivitas; pengecekan label hanya dilakukan pada 2 tetangga terdekat saja yaitu atas, bawah, kanan dan kiri. Gambar 26 ketika diberi pelabelan dengan 4 konektivitas didapat hasil seperti pada Gambar 27

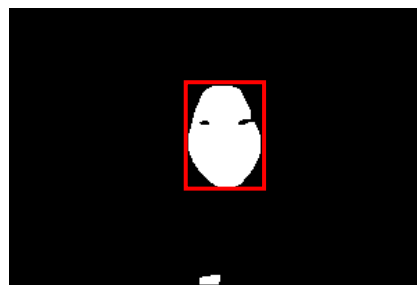
0	0	1	1	1	1	0	0
0	1	1	1	1	1	1	0
0	1	1	0	1	0	1	0
0	1	1	1	1	1	1	0
0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0
0	2	2	2	0	0	0	0
0	2	2	0	0	0	0	0

Gambar 27. Pelabelan 4 Konektivitas

Pada Gambar 27 terdapat dua label yang diberi label “1” dan “2” sesuai dengan letak urutan perkolom. Sekumpulan label disebut *holes* (lubang). Dari

dua *holes* tersebut pasti ada satu *holes* yang merupakan daerah wajah. Setelah dianalisis terdapat satu *holes* label yang mempunyai rasio tinggi dan lebar yang wilayahnya lebih luas dari *holes* yang lain.

Setelah menentukan wilayah wajah selanjutnya dilakukan pengambilan atau pemotongan gambar wajah dari citra suatu wajah, dimana citra wajah yang diambil merupakan warna putih atau bernilai piksel 1. Piksel yang di dalam kotak merah Gambar 28 merupakan *holes* yang mempunyai wilayah lebih besar dan diidentifikasi sebagai wajah. Gambar 29 menunjukkan beberapa contoh hasil deteksi wajah.



Gambar 28. Area yang diidentifikasi sebagai wajah

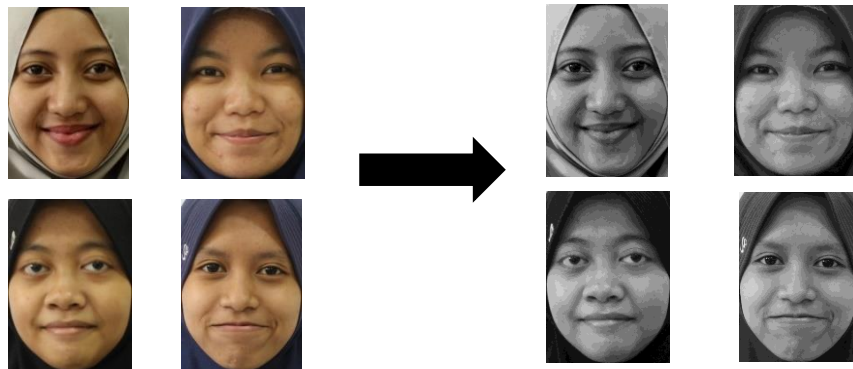


Gambar 29. Beberapa contoh citra hasil deteksi wajah

## B. Pra-Proses

### 1. RGB to *Grayscale*

Konversi *RGB to grayscale* ini banyak digunakan dalam sistem pengenalan wajah karena memiliki persamaan sederhana dan mampu mengurangi kebutuhan memori penghitungan sehingga mempercepat proses yang dilakukan. Gambar 30 adalah beberapa contoh hasil dari konversi citra RGB menjadi *grayscale*.

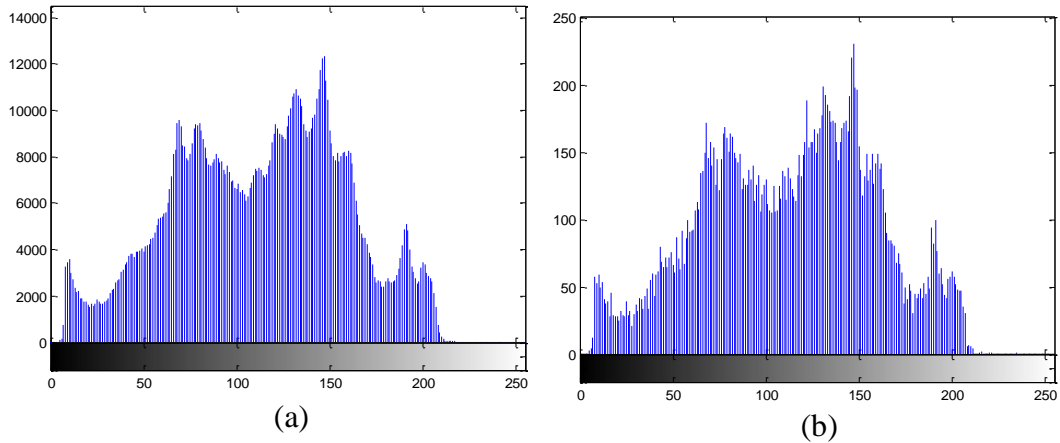


Gambar 30. Konversi citra RGB menjadi citra *grayscale*

### 2. *Resize*

Setiap citra hasil pemotongan memiliki ukuran piksel yang berbeda-beda, maka diperlukan penyeragaman ukuran citra dengan metode Interpolasi *Nearest Neighbor* (INN). Masing-masing citra diperkecil resolusinya menjadi 120 x 170 piksel seperti yang ditunjukkan Gambar 32. Gambar 31 memperlihatkan histogram *downsampling* atau memperkecil ukuran gambar digital artinya menampilkan suatu gambar dengan jumlah piksel yang lebih sedikit dengan proses *sampling* berjarak. Nilai piksel

pada citra rekonstruksi diperoleh dari menghilangkan piksel terdekat pada titik tersebut di sample citra asli.



Gambar 31. Histogram resize dengan metode INN. (a) sebelum *resize* (b) setelah *resize*



Gambar 32. Citra hasil *resize* resolusi 120 x 170 piksel

### C. *Principal Component Analysis*

Algoritma pengenalan wajah ini menggunakan citra latih Gambar 33 yang terdiri dari 5 individu, di mana untuk masing-masing individu terdiri dari 7 citra wajah sehingga jumlah total data latih adalah sebanyak 35 citra wajah. Sedangkan pada citra uji, masing-masing individu terdiri dari 4 citra wajah dan 2 citra selain dari *database* sehingga jumlah total data uji adalah sebanyak 22 citra wajah. Setiap citra wajah berukuran 120x170 piksel dan tingkat keabuan 256 (*grayscale*).



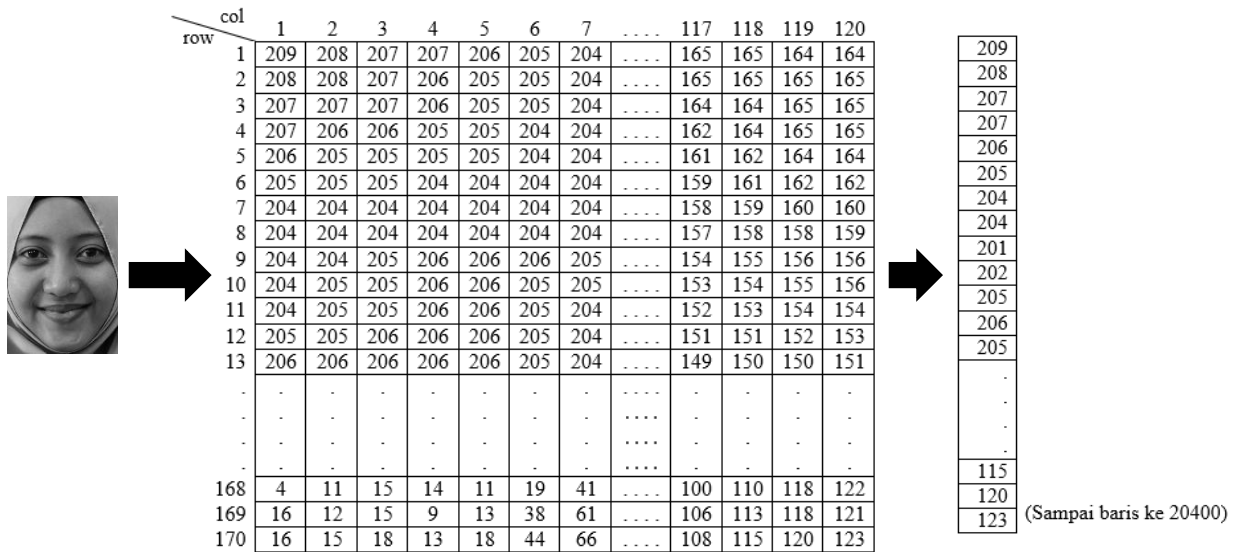
Gambar 33. *Database citra pelatihan*

## 1. Pra-PCA

### a. Reduksi Dimensi

Reduksi dimensi ini bertujuan untuk mempermudah proses selanjutnya yaitu pencarian nilai rata-rata baris. Gambar 34 adalah contoh reduksi dimensi pada salah satu citra pelatihan.





Gambar 34. Reduksi dimensi citra 2D menjadi 1D

Reduksi dimensi citra 2D menjadi 1D dilakukan pada semua citra pelatihan. Selanjutnya adalah menggabungkannya menjadi matriks  $\Gamma$  (augmentasi) Tabel 1.

Tabel 1. Matriks augmentasi citra data pelatihan

Matriks T											
Citra 1	Citra 2	Citra 3	Citra 4	Citra 5	Citra 6	Citra 7	....	Citra 32	Citra 33	Citra 34	Citra 35
183	209	211	229	234	137	193	....	216	213	215	215
183	208	210	229	234	205	193	....	217	218	216	215
183	207	208	229	234	227	193	....	217	218	217	214
183	207	206	229	234	213	192	....	218	214	218	214
183	206	205	229	234	224	192	....	218	214	217	214
183	205	205	229	234	230	191	....	217	219	215	215
183	204	206	229	234	218	191	....	217	219	212	216
183	204	206	229	234	220	191	....	216	215	210	216
180	201	204	229	237	220	190	....	217	222	212	218
180	202	205	229	233	220	191	....	217	207	221	200
180	205	205	229	231	220	191	....	217	208	216	224
181	206	202	229	235	219	192	....	217	216	206	216
181	205	199	229	239	219	192	....	217	211	208	204
.	.	.	.	.	.	.	....	.	.	.	.
.	.	.	.	.	.	.	....	.	.	.	.
.	.	.	.	.	.	.	....	.	.	.	.
.	.	.	.	.	.	.	....	.	.	.	.
172	115	88	162	107	149	153	....	15	21	106	4
181	120	93	162	107	154	152	....	9	20	39	5
191	123	106	165	109	157	155	....	15	19	124	8

(Sampai baris ke 20400)

Pada penelitian ini terdapat 35 citra pelatihan dengan resolusi masing-masing citra 120 x 170 piksel sehingga didapat ukuran matriks augmentasi sebesar 35 x 20400.

b. Rata-rata citra pelatihan

Menentukan citra rata-rata ( $\Psi$ ) ke arah baris dengan menjumlahkan semua citra pelatihan  $\Gamma_1 + \Gamma_2 + \Gamma_3 + \dots + \Gamma_{35}$  dan membaginya dengan jumlah total citra yaitu:  $\Psi = \frac{1}{35} \sum_{k=1}^{35} \Gamma_k$ . Hasilnya merupakan citra rata-rata  $\Psi$  yang menghasilkan citra wajah yang kabur seperti Gambar 35.



Gambar 35. Citra rata-rata pelatihan

c. Normalisasi Citra

Proses selanjutnya adalah mengurangi setiap citra wajah  $\Gamma_k$  dengan rata-rata  $\Psi$  atau  $\Phi_k = \Gamma_k - \Psi$  (deviasi dari rata-rata) sehingga diperoleh vektor  $\Phi_k$  dengan  $k = 1, 2, 3, \dots, 35$ . Dari setiap vektor  $\Phi_k$  bisa dibentuk matriks  $A$  dengan:

$$[\Phi_1 \quad \Phi_2 \quad \dots \quad \Phi_{35}] = A$$



Gambar 36. Citra hasil normalisasi

d. Matriks Kovarian

Matriks Kovarian diperoleh dari perkalian matriks *transpose*  $A$  dengan matriks  $A$ . Komponen utama dari suatu data adalah vektor eigen dari matriks kovarian. Matriks kovarian digunakan untuk memperoleh ruang wajah (*eigenspace*) dari ciri utama masing-masing sampel wajah. Pada penelitian ini matriks kovarian yang terbentuk berupa matriks simetris bujur sangkar yang berukuran  $35 \times 35$  ditunjukkan pada Tabel 2.

$$C = A^T A$$

Tabel 2. Matriks kovarian

Matriks Kovarian (C) × 10 <sup>8</sup>											
C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>4</sub>	C <sub>5</sub>	C <sub>6</sub>	C <sub>7</sub>	....	C <sub>32</sub>	C <sub>33</sub>	C <sub>34</sub>	35
0,7278	0,2534	0,2300	0,3452	0,3093	0,3209	0,4491	....	-0,0201	0,0359	-0,0308	0,1401
0,2534	0,3990	0,2898	0,2684	0,2083	0,3827	0,2335	....	0,1226	0,0553	0,0344	0,0228
0,2300	0,2898	0,3736	0,3158	0,2706	0,3608	0,1973	....	0,0554	0,0257	0,0095	-0,0203
0,3452	0,2684	0,3158	1,3854	0,8902	0,5356	0,3009	....	-0,1680	0,0030	-0,0931	-0,2256
0,3093	0,2083	0,2706	0,8902	0,9249	0,4167	0,2458	....	-0,1587	0,0149	-0,0152	-0,0779
0,3209	0,3827	0,3608	0,5356	0,4167	0,5452	0,2834	....	0,0330	0,0136	-0,0176	-0,0577
.	.	.	.	.	.	.	....	.	.	.	.
.	.	.	.	.	.	.	....	.	.	.	.
.	.	.	.	.	.	.	....	.	.	.	.
0,0359	0,0553	0,0257	0,0030	0,0149	0,0136	0,0314	....	0,2862	0,4332	0,2291	0,0674
-0,0308	0,0344	0,0095	-0,0931	-0,0152	-0,0176	-0,0209	....	0,1927	0,2291	0,4084	0,1084
0,1401	0,0228	-0,0203	-0,2256	-0,0779	-0,0577	0,0914	....	-0,0367	0,0674	0,1084	0,6683

(Sampai baris ke 35)

## 2. Proses PCA

### a. *Eigen Value Decomposition* (EVD)

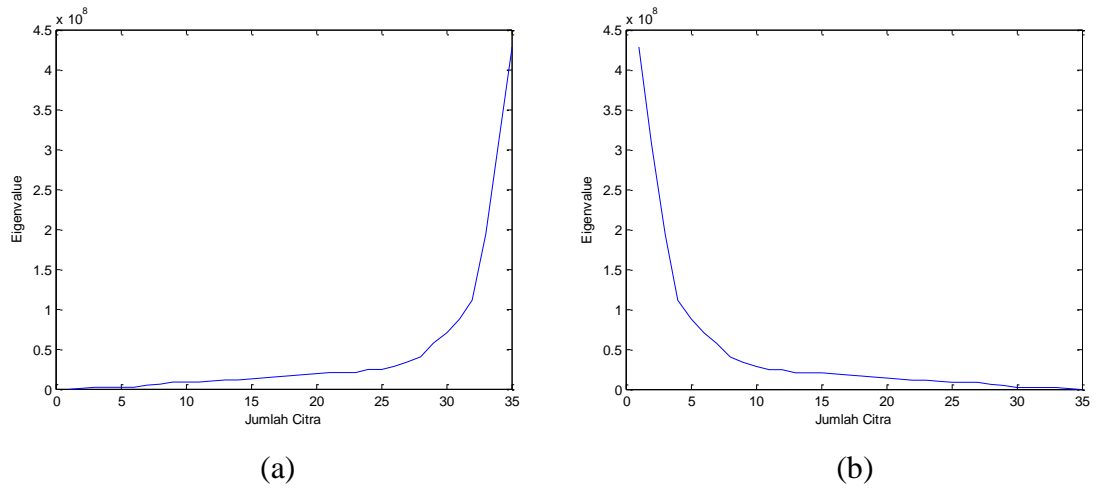
Nilai eigen adalah nilai karakteristik dari suatu matriks bujur sangkar, sedangkan vektor eigen diambil berdasarkan nilai eigen yang lebih besar dari 0. Proses menghitung vektor eigen dan nilai eigen dimaksudkan untuk memperoleh kumpulan hubungan karakteristik dari suatu citra, dalam hal ini citra wajah. Nilai eigen menunjukkan pengaruh terhadap variasi data. Nilai vektor eigen yang bersesuaian dengan nilai eigen kemudian diurutkan mulai dari nilai yang terbesar sampai yang terkecil, untuk mendapatkan fitur utama citra wajah. Nilai eigen terbesar mempunyai pengaruh paling besar terhadap variasi data.

Berikut hasil diagonal nilai eigen dari matriks kovarian;

$\lambda (\times 10^8)$		$\lambda (\times 10^8)$
0,0000		4,2766
0,0078		3,0784
0,0144		1,9389
0,0152		1,1055
0,0198		0,8713
0,0229		0,7014
0,0494		0,5757
0,0605		0,3994
0,0802		0,3298
0,0882		0,2829
0,0904		0,2493
0,0974		0,2440
0,1097		0,2104
0,1135		0,2075
0,1246		0,1981
0,1317		0,1862
0,1509		0,1717
0,1628		0,1628
0,1717		0,1509
0,1862		0,1317
0,1981		0,1246
0,2075		0,1135
0,2104		0,1097
0,2440		0,0974
0,2493		0,0904
0,2829		0,0882
0,3298		0,0802
0,3994		0,0605
0,5757		0,0494
0,7014		0,0229
0,8713		0,0198
1,1055		0,0152
1,9389		0,0144
3,0784		0,0078
4,2766		0,0000



Diurutkan dari yang terbesar  
hingga terkecil



Gambar 37. Spektrum *Eigenvalue*. (a) nilai eigen citra pelatihan (b) nilai eigen citra pelatihan setelah diurutkan dari yang terbesar hingga yang terkecil

b. *Eigenface*

*Eigenfaces* yang terbentuk merupakan nilai-nilai piksel yang merepresentasikan fitur-fitur utama wajah. Gambar 38 diurutkan berdasarkan komponen *eigenface* terpenting.

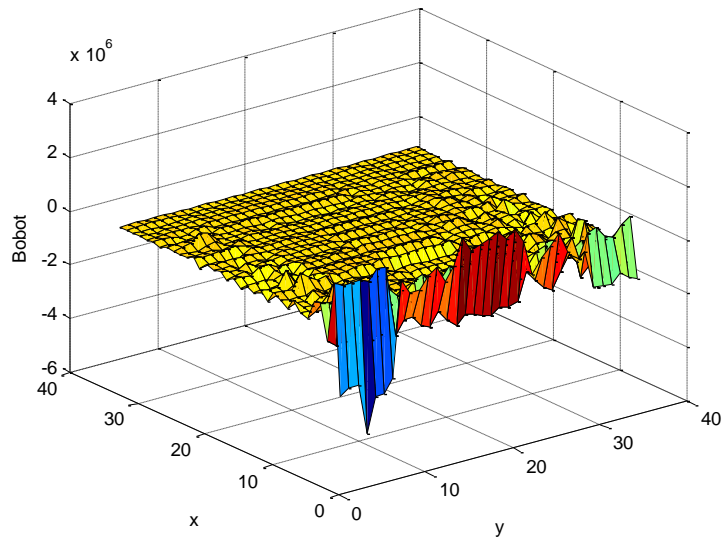


Gambar 38. Citra *eigenfaces*

Langkah berikutnya adalah menentukan jumlah *eigenface* terpilih yang mempunyai nilai signifikan dari *eigenface* lainnya. Nilai *eigenface* terpilih ini akan mewakili seluruh *eigenface* citra pelatihan. Artinya untuk menghitung bobot tidak perlu menggunakan *eigenface* sebesar 100%, cukup sebagian saja. Matriks *eigenface* mempunyai jumlah kolom sebanyak jumlah citra pelatihan dikurangi satu yaitu 34. Dikurangi satu karena terdapat satu vektor eigen yang mempunyai nilai eigen nol sehingga diabaikan. Matriks *eigenface* mempunyai jumlah baris sebanyak  $120 \times 170$  yaitu 20400. Jadi, matriks *eigenface* yang berukuran  $34 \times 20400$  akan direduksi menjadi matriks berukuran  $34 \times 200$ , matriks ini dinamakan komponen terpilih karena data 200 baris pertama memberikan pengaruh yang signifikan terhadap *database*.

#### **D. Tahap Pengenalan**

Pada tahap pengenalan dihitung bobot masing-masing citra pelatihan dengan cara memproyeksikan citra pelatihan ke dalam *eigenfaces*. Ukuran matriks bobot adalah jumlah kolom sebesar jumlah citra pelatihan yaitu 35 dan jumlah baris sebesar 34. Artinya setiap kolom dari matriks bobot adalah bobot suatu citra pelatihan. Gambar 39 menunjukkan bahwa bobot citra pelatihan pertama, kedua, ketiga, sampai ke-35 grafiknya menurun. Artinya bobot citra pelatihan pertama, kedua, ketiga, dan seterusnya memiliki nilai bobot dari yang paling penting hingga tidak penting.



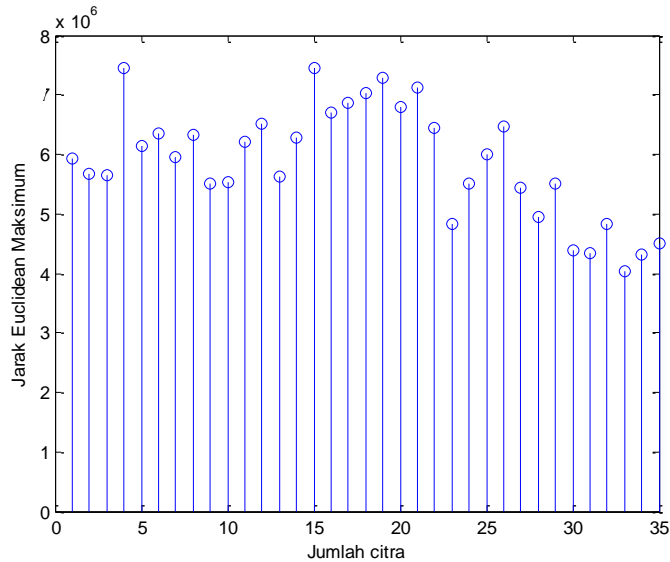
Gambar 39. Bobot citra pelatihan

Matriks bobot yang diperoleh digunakan untuk mencari nilai *threshold*. *Threshold* merupakan batas ambang yang harus dilewati oleh data pengujian, untuk membuktikan bahwa data pengujian merupakan data yang dikenali. Nilai *threshold* dihitung melalui perhitungan jarak tiap bobot masing-masing citra satu per satu menggunakan *euclidean distance*. Kemudian nilai *threshold* diambil dari jarak maksimum yang telah dikalikan dengan nilai indeks. Jarak maksimum yang diperoleh dalam penelitian ini yaitu  $7,4446 \times 10^6$  dan indeks yang digunakan sebesar 0,1 sehingga nilai *threshold* yang didapat adalah sebagai berikut

$$\theta = 0,1 \times 7,4446 \times 10^6 = 7,4446 \times 10^5$$

Grafik jarak maksimum masing-masing bobot citra pelatihan ditunjukkan pada Gambar 40





Gambar 40. Nilai maksimum dari jarak *euclidean* bobot citra pelatihan

### E. Tahap Pengujian

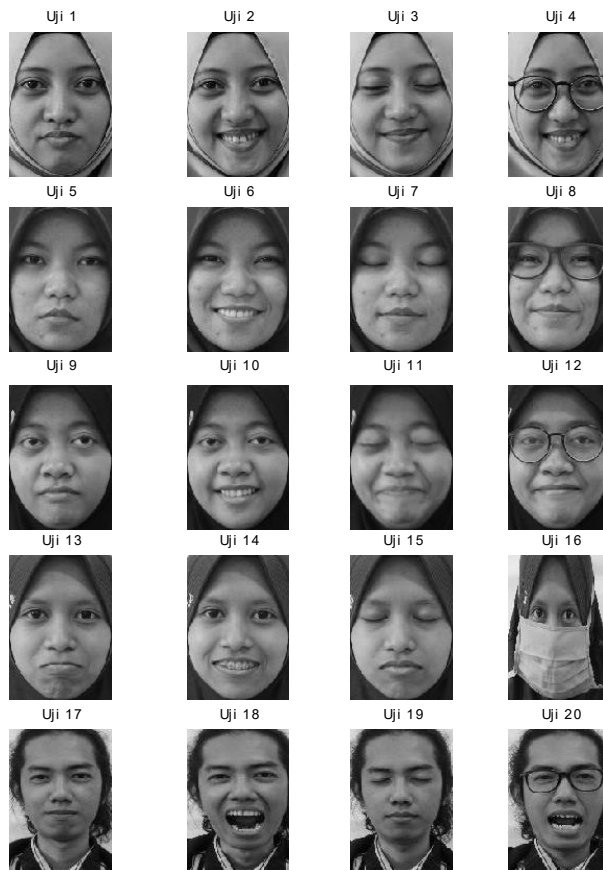
Proses pengujian citra wajah pada penelitian ini dilakukan dengan cara membandingkan jarak antara bobot citra pengujian dengan bobot setiap citra pelatihan pada ruang wajah. Proses ini dilakukan dengan cara menentukan jarak minimum menggunakan *euclidean distance*.

$$\varepsilon^2 = \|\Omega_{test} - \Omega_{train}\|^2$$

Setelah jarak minimum ditemukan maka jarak tersebut akan dibandingkan dengan nilai *threshold* ( $\theta$ ) sebagai indikasi apakah citra uji dikenali atau tidak. Jika jarak minimum citra uji lebih kecil dari *threshold* ( $\theta$ ) maka citra tersebut dikenali sebagai salah satu citra pelatihan, kemudian citra pelatihan tersebut akan ditampilkan. Tetapi jika jarak minimum citra uji lebih besar dari *threshold* ( $\theta$ ) maka citra uji akan dianggap tidak dikenali, kemudian dimunculkan *string* “Citra Tidak Dikenali”.

## 1. Pengujian pertama

Pengujian pertama ini berisi citra yang menggunakan aksesoris dan ekspresi wajah yang berbeda dari *database* citra pelatihan ditunjukkan pada Gambar 41.

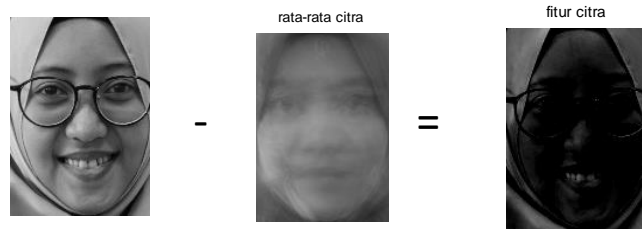


Gambar 41. *Database* citra pengujian

Setelah citra pengujian diinput pada Gambar 42, selanjutnya menghitung fitur dari citra pengujian dengan cara mengurangi piksel citra pengujian dengan rata-rata citra pelatihan ditunjukkan pada Gambar 43.

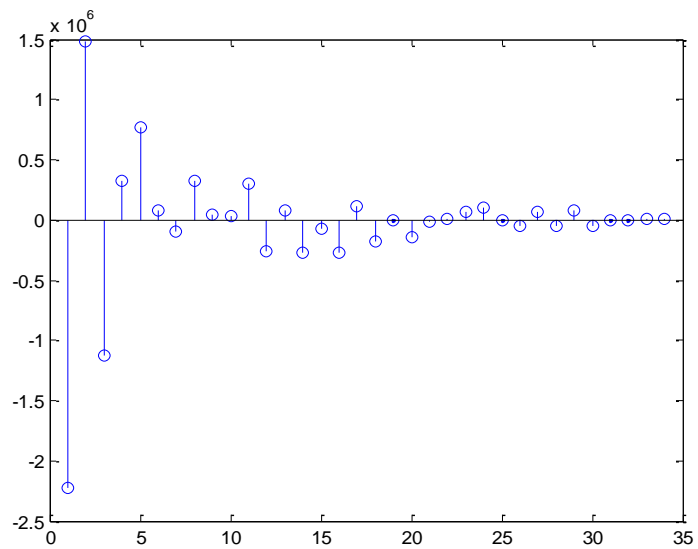


Gambar 42. Sampel Citra pengujian



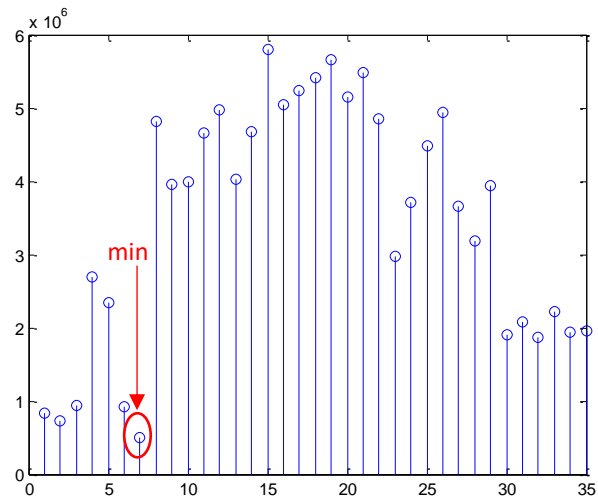
Gambar 43. Contoh fitur citra pengujian

Langkah berikutnya menghitung bobot citra pengujian dengan cara mengalikan fitur citra pengujian dengan *eigenface* yang terpilih. Gambar 44 menunjukkan bobot citra pengujian.



Gambar 44. Bobot sampel citra pengujian

Kemudian bobot citra pengujian dibandingkan dengan setiap kolom matriks bobot citra pelatihan menggunakan *euclidean distance*.



Gambar 45. *Euclidean distance* sampel citra pengujian



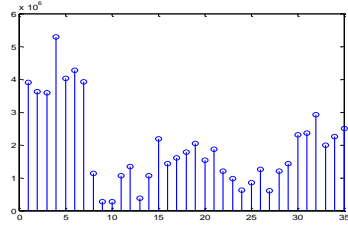
Gambar 45 menunjukkan jarak *Euclidean* minimum pada sampel citra pengujian adalah  $5,0821 \times 10^5$ , karena jarak *euclidean* minimum lebih kecil dari nilai *threshold*  $7,4446 \times 10^5$  maka sampel citra pengujian dikenali mirip citra pelatihan indeks ke-7; ketika dilihat pada *database* citra pelatihan indeks ke-7 adalah seperti pada Gambar 46





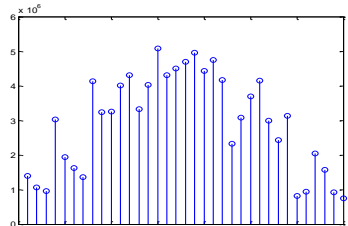
Gambar 46. Citra pengenalan indeks ke-7

Sampel citra pengujian pada Gambar 42 juga dekat dengan citra indeks ke-1, 2, dan 3 karena merupakan orang yang sama; dan jauh dengan citra indeks ke-15.

Berikut hasil pengujian pertama.

Citra pengujian	Citra pengenalan	Grafik jarak <i>Euclidean</i>
 <p data-bbox="501 665 616 696">(Ratna1)</p>		

Citra pengujian di atas diberi nama Ratna1. Nilai jarak *euclidean* minimum citra Ratna1 adalah  $2,6791 \times 10^5$  (terletak pada indeks ke-9) yang lebih kecil dari nilai *threshold* maka citra Ratna1 dapat dikenali sebagai citra indeks ke-9 pada *database* citra pelatihan. Citra Ratna1 juga dekat dengan citra indeks ke-10 dan 13 karena masih orang yang sama. Dan mempunyai jarak paling jauh dengan citra indeks ke-4.

Citra pengujian	Citra pengenalan	Grafik jarak <i>Euclidean</i>
 <p data-bbox="504 1534 612 1565">(Naza1)</p>		

Citra pengujian di atas diberi nama Naza1. Nilai jarak *euclidean* minimum citra Naza1 adalah  $7.3797 \times 10^5$  (terletak pada indeks ke-35) yang lebih kecil dari nilai *threshold*, maka citra Naza1 dapat dikenali sebagai citra indeks ke-35 pada *database* citra pelatihan. Citra Naza1 juga dekat dengan

citra indeks ke-30, 31 dan 34 karena masih orang yang sama. Dan jarak paling jauh dengan citra indeks ke-15.

## 2. Pengujian kedua





Pengujian kedua menggunakan citra selain yang ada di *database*.







































Gambar 47. Citra pengujian selain *database*.  
(a) Udin1 (b) Wuri1

Pada Gambar 47 nilai jarak *euclidean* minimum Udin1 dan Wuri1 masing-masing adalah  $7,5330 \times 10^5$  dan  $14,132 \times 10^5$ . Jarak minimum tersebut lebih besar daripada nilai *threshold* sehingga wajah tidak dikenali.



Tabel 3. Hasil pangujian pada 22 sampel citra

No	Citra Pengujian	Citra Pengenalan	Minimum Jarak <i>Euclidean</i> ( $\times 10^5$ )	kebenaran
1			1.4201	1
2			1.4097	1

3			2.9994	1
4			5.0821	1
5			1.2715	1
6			1.4336	1
7			1.0709	1
8			2.6791	1
9			0.7500	1
10			5.0485	1
11			4.2729	1

12			7.3989	1
13			1.3375	1
14			2.7716	1
15			1.0704	1
16			7.2350	0
17			3.2886	1
18			2.4593	1
19			7.3797	1
20			3.2886	1



21		Tidak dikenali	7,5330	1
22		Tidak dikenali	14,132	1
Jumlah Benar				21

$$akurasi (\%) = \frac{jumlah\ benar}{total} \times 100\% = \frac{21}{22} \times 100\% = 95\%$$

Sistem pengenalan wajah yang telah dibuat mempunyai tingkat keakuratan sebesar 95%. Adapun faktor yang mempengaruhi hasil pengenalan wajah yang teridentifikasi seperti yang ditunjukkan No 16, yaitu ketika wajah tertutupi oleh masker. Citra wajah pada No 16 mempunyai jarak *euclidean* minimum  $7.2350 \times 10^5$  lebih kecil daripada *threshold*  $7.4446 \times 10^5$ , citra tersebut dapat dikenali namun karena beberapa informasi fitur wajah hilang maka hasil identifikasi pengenalan wajah menjadi tidak tepat. Berdasarkan permasalahan tersebut, sistem pengenalan wajah yang telah dibuat saat ini terbatas pada citra wajah yang tidak terhalang oleh suatu benda yang menutupi sebagian besar wajah.

## BAB V

### KESIMPULAN DAN SARAN

#### A. Kesimpulan

1. Pendeteksian wajah dalam citra dengan metode *skin detection* terdiri dari lima tahap yaitu mentransformasikan citra RGB ke model warna YCbCr; melakukan pemisahan warna kulit pada komponen krominan dengan *threshold* yang sudah ditentukan; melakukan perbaikan citra dengan morfologi *opening* yaitu erosi diikuti dilasi; memberikan label pada citra hasil morfologi *opening*; memotong citra yang diidentifikasi sebagai area wajah. Proses ini mampu memisahkan objek wajah dari latar.
2. *Principal Component Analysis* (PCA) dapat menangani sejumlah data yang cukup besar serta kemampuannya menangani dimensi data yang kompleks, seperti fitur-fitur pada citra wajah yang diekstrak berupa dimensi vektor eigen yang berkorelasi dengan nilai eigen tertinggi sehingga diperoleh ruang wajah (*eigenspace*) atau *eigenfaces*. Metode PCA pada penelitian ini dapat mereduksi fitur-fitur wajah yang semula matriksnya berukuran  $34 \times 20400$  menjadi matriks berukuran  $34 \times 200$ , matriks tersebut merupakan komponen terpilih karena memberikan pengaruh yang signifikan terhadap *database*.
3. Pencocokkan wajah dilakukan dengan menghitung jarak minimum antara nilai karakteristik dari piksel citra uji dengan citra pelatihan menggunakan *euclidean distance*.

4. *Principal Component Analysis* dapat diimplementasikan untuk pengenalan wajah dengan tingkat akurasi 95%. Sistem pengenalan wajah ini mampu digunakan untuk mengidentifikasi wajah meskipun wajah dengan ekspresi yang berbeda dan menggunakan aksesoris berupa kacamata.

## **B. Saran**

1. Penelitian ini dibatasi pada satu wajah dalam satu citra. Untuk penelitian lanjutan dapat digunakan beberapa wajah dalam satu citra.
2. Perlu dilakukan penelitian lanjutan untuk membuat program yang dapat mengolah pengenalan wajah secara waktu nyata (*real-time*).

## DDAFTAR PUSTAKA

- Alwakeel, Mohammed., Shaaban, Zyad. (2010). Face Recognition Based on Haar Wavelet Transform and Principal Component Analysis via Levenberg-Marquardt Backpropagation Neural Network. *European Journal of Scientific Research*, 42 (1), 25-31.
- Blanchet, Gerard., Charbit, Maurice. (2014). *Digital Signal and Image Processing using MATLAB®*. London: ISTE.
- Bradski., Ogiuchi, T., Higashikubo, M. (2010). Visual Tracking Algorithm Using Pixel-Pair Feature. *International Confrence on Pattern Recognition*, 4, 1808–1811.
- Chai, Douglas., Ngan, K.N. (1999). Face Segmentation Using Skin-Color Map in Videophone Applications. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, 9 (4), 551-564.
- Chang, Henry., Robles, Ulises. (2000). *Face Detection*, EE Final Project Report, <http://www-cs-students.stanford.edu/~robles/ee368/main.html>.
- Datta, A.K., Datta, M., Banerjee, P.K. (2016). *Face Detecton and Recogniton Theory and Practice*. New York: CRC Press.
- Gonzalez, R.C., Woods, R.E. (2008). *Digital Image Processing Third Edition*. New Jersey: Prentice Hall.
- Kaymak, Sezin. (2003). *Face Detection, Recogniton and reconstruction using eigenfaces*. Diunduh pada tanggal 07 Desember 2018 dari [www.faraday.ee.emu.edu.tr](http://www.faraday.ee.emu.edu.tr).
- Keho, Yaya. (2012). *The Basics of Linear Principal Components Analysis*. Diunduh pada tanggal 18 Januari 2019 dari [www.intechweb.org](http://www.intechweb.org).
- Lu, Xiaoguang. (2008). *Image Analysis for Face Recognition*. Dept. of Computer Science & Engineering, Michigan State University.
- Putra, Darma. (2010). *Pengolahan Citra Digital*. Yogyakarta: Penerbit ANDI.
- Qidwai, Uvais., Chen, C.H. (2009). *Digital Image Processing: An Algorithmic Approach with MATLAB®*. New York: CRC Press.
- Rajaraman, Sivaramakrishnan., Chokkalingam, Arun. (2013). Connected Components Labeling and Extraction Based Interphase Removal from Chromosome Images. *International Journal of Bio-Science and Bio-Technology*, 5 (1), 81-89.

- Sachs, Jonathan. (2001). *Image Resampling*. Diunduh pada tanggal 20 Juli 2019 dari [www.dl-c.com](http://www.dl-c.com).
- Shih, F.Y. (2010). *Image Processing and Pattern Recognition: Fundamentals and Techniques*. New Jersey: IEEE Press.
- Smith, L.I. (2002). *A Tutorial on Principal Components Analysis*. Diunduh pada tanggal 08 April 2019 dari [www.iro.umontreal.ca](http://www.iro.umontreal.ca).
- Sutoyo, T., Mulyanto, Edy., Suhartono, Vincent., Nurhayati, O.Dwi., Wijanarto. (2009). *Teori Pengolahan Citra Digital*. Yogyakarta: Penerbit ANDI.
- Yang, M.H., Kriegman, D.J., Ahuja, Narendra. (2002), Detecting Faces in Images: A Survey. *IEEE Pattern Analysis and Machine Intelligence*, 24 (1), 34-58.
- Zhao, Wenyi., Chellappa, Rama. (2006). *Face Processing Advanced Modeling and Methods*. London: Elsevier Inc.
- Zhou, Hulyu., Zhang, Jianguo., Wu, Jiahua. (2010). *Digital Image Proccesing Part II*. Frederiksberg: Ventus Publishing ApS.

## LAMPIRAN

### Lampiran 1. Matriks kovarian (C) berukuran simetris bujur sangkar $35 \times 35$

Kolom 1-12 ( $\times 10^8$ )

0,7278	0,2534	0,2300	0,3452	0,3093	0,3209	0,4491	0,0587	-0,1789	-0,1852	-0,1834	-0,1508
0,2534	0,3990	0,2898	0,2684	0,2083	0,3827	0,2335	-0,1485	-0,0947	-0,1173	-0,0707	-0,1304
0,2300	0,2898	0,3736	0,3158	0,2706	0,3608	0,1973	-0,1523	-0,0957	-0,0918	-0,0444	-0,0817
0,3452	0,2684	0,3158	1,3854	0,8902	0,5356	0,3009	-0,1929	-0,2441	-0,2281	0,2341	-0,0026
0,3093	0,2083	0,2706	0,8902	0,9249	0,4167	0,2458	-0,1600	-0,1987	-0,1912	0,1062	0,0263
0,3209	0,3827	0,3608	0,5356	0,4167	0,5452	0,2834	-0,1759	-0,1343	-0,1457	0,0063	-0,0760
0,4491	0,2335	0,1973	0,3009	0,2458	0,2834	0,5379	-0,0397	-0,1440	-0,1559	-0,1239	-0,1378
0,0587	-0,1485	-0,1523	-0,1929	-0,1600	-0,1759	-0,0397	0,4396	0,0197	0,0280	-0,0702	-0,0066
-0,1789	-0,0947	-0,0957	-0,2441	-0,1987	-0,1343	-0,1440	0,0197	0,1698	0,1559	-0,0447	0,0264
-0,1852	-0,1173	-0,0918	-0,2281	-0,1912	-0,1457	-0,1559	0,0280	0,1559	0,1839	-0,0295	0,0341
-0,1834	-0,0707	-0,0444	0,2341	0,1062	0,0063	-0,1239	-0,0702	-0,0447	-0,0295	0,6563	0,1783
-0,1508	-0,1304	-0,0817	-0,0026	0,0263	-0,0760	-0,1378	-0,0066	0,0264	0,0341	0,1783	0,3385
-0,1771	-0,0918	-0,1034	-0,2748	-0,2283	-0,1501	-0,1422	0,0238	0,1584	0,1458	-0,0521	0,0112
-0,0208	-0,1458	-0,1616	-0,1872	-0,1669	-0,1724	-0,0552	0,2306	0,0393	0,0380	-0,0304	0,0137
0,1444	-0,1937	-0,1872	-0,3081	-0,2178	-0,2277	-0,0265	0,4191	0,0344	0,0362	-0,1098	-0,0227
-0,2551	-0,1428	-0,1383	-0,3495	-0,2933	-0,2097	-0,2083	-0,0003	0,1474	0,1448	-0,0452	0,0462
-0,2629	-0,1523	-0,1318	-0,3542	-0,3023	-0,2192	-0,2176	-0,0029	0,1362	0,1529	-0,0501	0,0399
-0,2212	-0,0247	-0,0080	0,2616	0,0971	0,0381	-0,1225	-0,1250	-0,0704	-0,0604	0,5642	0,1168
-0,2355	-0,1008	-0,0764	-0,0232	-0,0282	-0,0813	-0,1465	-0,0495	0,0030	0,0040	0,2773	0,2296
-0,2600	-0,1385	-0,1217	-0,3491	-0,2996	-0,2050	-0,2089	-0,0045	0,1373	0,1499	-0,0585	0,0272
-0,1980	-0,1937	-0,1848	-0,3547	-0,2762	-0,2632	-0,1992	0,0911	0,1076	0,1037	-0,0700	0,0071
0,0049	-0,1407	-0,1311	-0,1649	-0,1520	-0,1535	-0,0654	0,2196	0,0295	0,0335	-0,1027	-0,0052
-0,1240	-0,0552	-0,0575	-0,1237	-0,1364	-0,0815	-0,0832	-0,0454	0,0831	0,0773	-0,0288	-0,0080
-0,1477	-0,0785	-0,0716	-0,1966	-0,1916	-0,1163	-0,1119	-0,0319	0,0815	0,0954	-0,0305	-0,0204
-0,1122	-0,0268	-0,0034	0,1848	0,1640	0,0422	-0,0825	-0,0619	-0,0432	-0,0411	0,2787	0,1781
-0,1817	-0,1133	-0,0903	-0,0749	-0,0380	-0,1118	-0,1337	-0,0375	0,0028	0,0075	0,1635	0,2060
-0,1568	-0,0728	-0,0754	-0,2506	-0,2273	-0,1314	-0,1172	-0,0274	0,0859	0,0949	-0,0502	-0,0292
-0,0964	-0,0929	-0,0728	-0,0744	-0,0932	-0,1047	-0,0507	-0,0383	0,0718	0,0782	-0,0263	-0,0170
0,2229	-0,0095	-0,0448	-0,2501	-0,1562	-0,0989	0,1273	0,1980	0,0053	-0,0056	-0,2766	-0,1966
-0,0304	0,0349	0,0059	-0,1162	-0,0187	-0,0260	-0,0145	-0,0606	-0,0205	-0,0314	-0,1796	-0,0930
-0,0137	0,0303	0,0121	-0,1189	-0,0465	-0,0187	0,0445	-0,1070	-0,0338	-0,0381	-0,1402	-0,0561
-0,0201	0,1226	0,0554	-0,1680	-0,1587	0,0330	0,0658	-0,1641	-0,0758	-0,0913	-0,0338	-0,1142
0,0359	0,0553	0,0257	0,0030	0,0149	0,0136	0,0314	-0,0541	-0,0712	-0,0788	-0,1322	-0,0970
-0,0308	0,0344	0,0095	-0,0931	-0,0152	-0,0176	-0,0209	-0,0878	-0,0263	-0,0334	-0,1603	-0,0884
0,1401	0,0228	-0,0203	-0,2256	-0,0779	-0,0577	0,0914	0,1163	-0,0192	-0,0393	-0,3212	-0,1458

Kolom 13-24 ( $\times 10^8$ )

-0,1771	-0,0208	0,1444	-0,2551	-0,2629	-0,2212	-0,2355	-0,2600	-0,1980	0,0049	-0,1240	-0,1477
-0,0918	-0,1458	-0,1937	-0,1428	-0,1523	-0,0247	-0,1008	-0,1385	-0,1937	-0,1407	-0,0552	-0,0785
-0,1034	-0,1616	-0,1872	-0,1383	-0,1318	-0,0080	-0,0764	-0,1217	-0,1848	-0,1311	-0,0575	-0,0716
-0,2748	-0,1872	-0,3081	-0,3495	-0,3542	0,2616	-0,0232	-0,3491	-0,3547	-0,1649	-0,1237	-0,1966
-0,2283	-0,1669	-0,2178	-0,2933	-0,3023	0,0971	-0,0282	-0,2996	-0,2762	-0,1520	-0,1364	-0,1916
-0,1501	-0,1724	-0,2277	-0,2097	-0,2192	0,0381	-0,0813	-0,2050	-0,2632	-0,1535	-0,0815	-0,1163
-0,1422	-0,0552	-0,0265	-0,2083	-0,2176	-0,1225	-0,1465	-0,2089	-0,1992	-0,0654	-0,0832	-0,1119
0,0238	0,2306	0,4191	-0,0003	-0,0029	-0,1250	-0,0495	-0,0045	0,0911	0,2196	-0,0454	-0,0319
0,1584	0,0393	0,0344	0,1474	0,1362	-0,0704	0,0030	0,1373	0,1076	0,0295	0,0831	0,0815
0,1458	0,0380	0,0362	0,1448	0,1529	-0,0604	0,0040	0,1499	0,1037	0,0335	0,0773	0,0954
-0,0521	-0,0304	-0,1098	-0,0452	-0,0501	0,5642	0,2773	-0,0585	-0,0700	-0,1027	-0,0288	-0,0305
0,0112	0,0137	-0,0227	0,0462	0,0399	0,1168	0,2296	0,0272	0,0071	-0,0052	-0,0080	-0,0204
0,1692	0,0422	0,0398	0,1509	0,1406	-0,0723	-0,0065	0,1423	0,1120	0,0309	0,0891	0,0910
0,0422	0,3061	0,2627	0,0323	0,0249	-0,0482	0,0030	0,0277	0,1287	0,1981	-0,0161	-0,0104
0,0398	0,2627	0,7166	-0,0006	-0,0067	-0,1963	-0,0888	-0,0065	0,1452	0,2855	-0,0533	-0,0368
0,1509	0,0323	-0,0006	0,2747	0,2536	-0,0452	0,0489	0,2460	0,1545	0,0269	0,1225	0,1264
0,1406	0,0249	-0,0067	0,2536	0,2910	-0,0478	0,0444	0,2677	0,1557	0,0240	0,1064	0,1442
-0,0723	-0,0482	-0,1963	-0,0452	-0,0478	0,8460	0,3171	-0,0557	-0,1094	-0,1217	-0,0319	-0,0302
-0,0065	0,0030	-0,0888	0,0489	0,0444	0,3171	0,4598	0,0348	0,0154	-0,0361	-0,0121	-0,0160
0,1423	0,0277	-0,0065	0,2460	0,2677	-0,0557	0,0348	0,2794	0,1562	0,0243	0,1067	0,1421
0,1120	0,1287	0,1452	0,1545	0,1557	-0,1094	0,0154	0,1562	0,3796	0,1490	0,0657	0,0872
0,0309	0,1981	0,2855	0,0269	0,0240	-0,1217	-0,0361	0,0243	0,1490	0,3507	0,0002	0,0071
0,0891	-0,0161	-0,0533	0,1225	0,1064	-0,0319	-0,0121	0,1067	0,0657	0,0002	0,1748	0,1386
0,0910	-0,0104	-0,0368	0,1264	0,1442	-0,0302	-0,0160	0,1421	0,0872	0,0071	0,1386	0,2016
-0,0570	-0,0403	-0,1138	-0,0405	-0,0402	0,2849	0,2430	-0,0514	-0,0771	-0,0871	-0,0437	-0,0574
-0,0016	0,0025	-0,0525	0,0386	0,0394	0,1601	0,2773	0,0236	0,0262	-0,0085	0,0023	0,0029
0,0983	-0,0061	-0,0332	0,1408	0,1529	-0,0436	-0,0161	0,1509	0,0941	0,0085	0,1391	0,1804
0,0698	0,0066	-0,0172	0,0708	0,0736	-0,0562	-0,0100	0,0761	0,0981	0,0159	0,0831	0,0854
0,0222	0,0781	0,2849	-0,0528	-0,0567	-0,3929	-0,2884	-0,0433	0,0754	0,1082	-0,0518	-0,0379
-0,0137	-0,0629	-0,0739	-0,0339	-0,0308	-0,1751	-0,1275	-0,0251	-0,0232	-0,0522	-0,0411	-0,0340
-0,0291	-0,0697	-0,1193	-0,0268	-0,0164	-0,1487	-0,1126	-0,0156	-0,0429	-0,0694	-0,0414	-0,0384
-0,0495	-0,1031	-0,2177	-0,0456	-0,0333	0,1431	-0,0043	-0,0357	-0,1119	-0,1307	-0,0163	0,0008
-0,0583	-0,0665	-0,1018	-0,0556	-0,0452	-0,0866	-0,1190	-0,0423	-0,0474	-0,0762	-0,0410	-0,0394
-0,0192	-0,0720	-0,0954	-0,0251	-0,0155	-0,1519	-0,1269	-0,0136	-0,0193	-0,0575	-0,0371	-0,0310
-0,0107	0,0013	0,1383	-0,0567	-0,0612	-0,3832	-0,2520	-0,0572	0,0185	0,0383	-0,0593	-0,0560

Kolom 25-35 ( $\times 10^8$ )

-0,1122	-0,1817	-0,1568	-0,0964	0,2229	-0,0304	-0,0137	-0,0201	0,0359	-0,0308	0,1401
-0,0268	-0,1133	-0,0728	-0,0929	-0,0095	0,0349	0,0303	0,1226	0,0553	0,0344	0,0228
-0,0034	-0,0903	-0,0754	-0,0728	-0,0448	0,0059	0,0121	0,0554	0,0257	0,0095	-0,0203
0,1848	-0,0749	-0,2506	-0,0744	-0,2501	-0,1162	-0,1189	-0,1680	0,0030	-0,0931	-0,2256
0,1640	-0,0380	-0,2273	-0,0932	-0,1562	-0,0187	-0,0465	-0,1587	0,0149	-0,0152	-0,0779
0,0422	-0,1118	-0,1314	-0,1047	-0,0989	-0,0260	-0,0187	0,0330	0,0136	-0,0176	-0,0577
-0,0825	-0,1337	-0,1172	-0,0507	0,1273	-0,0145	0,0445	0,0658	0,0314	-0,0209	0,0914
-0,0619	-0,0375	-0,0274	-0,0383	0,1980	-0,0606	-0,1070	-0,1641	-0,0541	-0,0878	0,1163
-0,0432	0,0028	0,0859	0,0718	0,0053	-0,0205	-0,0338	-0,0758	-0,0712	-0,0263	-0,0192
-0,0411	0,0075	0,0949	0,0782	-0,0056	-0,0314	-0,0381	-0,0913	-0,0788	-0,0334	-0,0393
0,2787	0,1635	-0,0502	-0,0263	-0,2766	-0,1796	-0,1402	-0,0338	-0,1322	-0,1603	-0,3212
0,1781	0,2060	-0,0292	-0,0170	-0,1966	-0,0930	-0,0561	-0,1142	-0,0970	-0,0884	-0,1458
-0,0570	-0,0016	0,0983	0,0698	0,0222	-0,0137	-0,0291	-0,0495	-0,0583	-0,0192	-0,0107
-0,0403	0,0025	-0,0061	0,0066	0,0781	-0,0629	-0,0697	-0,1031	-0,0665	-0,0720	0,0013
-0,1138	-0,0525	-0,0332	-0,0172	0,2849	-0,0739	-0,1193	-0,2177	-0,1018	-0,0954	0,1383
-0,0405	0,0386	0,1408	0,0708	-0,0528	-0,0339	-0,0268	-0,0456	-0,0556	-0,0251	-0,0567
-0,0402	0,0394	0,1529	0,0736	-0,0567	-0,0308	-0,0164	-0,0333	-0,0452	-0,0155	-0,0612
0,2849	0,1601	-0,0436	-0,0562	-0,3929	-0,1751	-0,1487	0,1431	-0,0866	-0,1519	-0,3832
0,2430	0,2773	-0,0161	-0,0100	-0,2884	-0,1275	-0,1126	-0,0043	-0,1190	-0,1269	-0,2520
-0,0514	0,0236	0,1509	0,0761	-0,0433	-0,0251	-0,0156	-0,0357	-0,0423	-0,0136	-0,0572
-0,0771	0,0262	0,0941	0,0981	0,0754	-0,0232	-0,0429	-0,1119	-0,0474	-0,0193	0,0185
-0,0871	-0,0085	0,0085	0,0159	0,1082	-0,0522	-0,0694	-0,1307	-0,0762	-0,0575	0,0383
-0,0437	0,0023	0,1391	0,0831	-0,0518	-0,0411	-0,0414	-0,0163	-0,0410	-0,0371	-0,0593
-0,0574	0,0029	0,1804	0,0854	-0,0379	-0,0340	-0,0384	0,0008	-0,0394	-0,0310	-0,0560
0,4290	0,1842	-0,0680	-0,0483	-0,1970	-0,1064	-0,1148	-0,1182	-0,1020	-0,1062	-0,1484
0,1842	0,3374	0,0082	0,0013	-0,1698	-0,0711	-0,0535	-0,0407	-0,0915	-0,0796	-0,1339
-0,0680	0,0082	0,1951	0,0758	-0,0155	-0,0192	-0,0224	0,0171	-0,0285	-0,0162	-0,0350
-0,0483	0,0013	0,0758	0,2591	0,0171	-0,0433	-0,0086	-0,0822	-0,0754	-0,0552	-0,0201
-0,1970	-0,1698	-0,0155	0,0171	0,8330	0,0507	0,0744	-0,1931	-0,0385	0,0081	0,4702
-0,1064	-0,0711	-0,0192	-0,0433	0,0507	0,4160	0,1885	0,1731	0,1972	0,2759	0,1824
-0,1148	-0,0535	-0,0224	-0,0086	0,0744	0,1885	0,4578	0,1573	0,1553	0,2290	0,1533
-0,1182	-0,0407	0,0171	-0,0822	-0,1931	0,1731	0,1573	0,7980	0,2862	0,1927	-0,0367
-0,1020	-0,0915	-0,0285	-0,0754	-0,0385	0,1972	0,1553	0,2862	0,4332	0,2291	0,0674
-0,1062	-0,0796	-0,0162	-0,0552	0,0081	0,2759	0,2290	0,1927	0,2291	0,4084	0,1084
-0,1484	-0,1339	-0,0350	-0,0201	0,4702	0,1824	0,1533	-0,0367	0,0674	0,1084	0,6683



## Lampiran 2. Eigen vektor

Kolom 1-12

$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	$\lambda_6$	$\lambda_7$	$\lambda_8$	$\lambda_9$	$\lambda_{10}$	$\lambda_{11}$	$\lambda_{12}$
-0,2256	0,2801	0,1660	-0,0581	0,2831	0,0171	-0,1882	0,1836	-0,0835	0,2820	-0,2214	-0,0770
-0,1975	0,0971	-0,1240	0,0859	0,2330	-0,1488	-0,1049	-0,2997	-0,0382	-0,2041	-0,0368	0,0322
-0,2013	0,0607	-0,0847	0,1409	0,1672	-0,1072	-0,1302	-0,2787	-0,0736	-0,1405	-0,0632	0,0043
-0,4880	-0,0952	0,1643	0,2116	-0,1873	0,3118	0,2936	0,1310	0,1327	-0,0665	0,1334	0,3009
-0,3853	-0,0192	0,1184	0,1411	-0,3258	0,1488	-0,0329	0,0126	0,1240	0,1083	-0,0164	-0,2650
-0,2953	0,0499	-0,0428	0,1492	0,1926	-0,0766	-0,1341	-0,3011	-0,0847	-0,1780	0,0395	0,0256
-0,2043	0,1933	0,0364	-0,0253	0,2625	-0,0902	-0,1220	0,3987	-0,1107	0,0866	0,0702	-0,0327
0,1162	0,1015	0,2954	-0,2133	0,0293	0,1666	0,0013	-0,1949	0,0858	0,0948	0,1995	0,1808
0,1355	-0,0319	-0,0078	0,1642	0,0163	0,0071	0,0054	-0,1110	-0,0625	0,0305	0,1665	-0,0776
0,1368	-0,0440	0,0029	0,1769	0,0098	0,0307	0,0223	-0,0976	-0,0833	0,0786	0,1693	-0,0551
-0,0799	-0,3464	0,0992	-0,1981	0,0221	-0,1824	0,3144	-0,0363	-0,2655	0,1838	-0,0751	-0,1364
0,0232	-0,1905	0,0857	-0,0353	-0,1861	-0,0598	-0,3550	0,0795	-0,0517	0,1591	0,1269	0,1128
0,1449	-0,0224	-0,0216	0,1444	0,0405	0,0016	0,0279	-0,0955	-0,0419	0,0407	0,1322	-0,0705
0,1145	0,0248	0,1996	-0,1521	0,0454	0,1930	-0,0140	0,0190	-0,0607	-0,2473	0,0906	0,1180
0,1594	0,1597	0,3912	-0,2656	0,0928	0,2033	-0,0483	-0,1687	-0,1277	0,2332	0,0457	-0,2245
0,1885	-0,0769	-0,0635	0,1844	0,0380	0,0252	-0,0380	-0,0543	-0,0011	0,1221	0,0083	0,1818
0,1926	-0,0778	-0,0781	0,1864	0,0343	0,0370	-0,0301	-0,0476	-0,0006	0,1480	-0,0262	0,2302
-0,1071	-0,3978	-0,0065	-0,2927	0,1721	-0,1224	0,4166	-0,1051	-0,0554	-0,0737	0,0424	-0,0206
0,0175	-0,2915	0,0398	-0,1403	-0,0394	-0,1519	-0,3152	0,1122	0,1023	-0,1708	-0,0882	0,0529
0,1889	-0,0678	-0,0793	0,1947	0,0431	0,0373	-0,0134	-0,0692	-0,0136	0,1070	-0,0337	0,2194
0,2026	-0,0073	0,0614	0,0613	-0,0573	0,1150	0,0527	0,0755	0,0483	-0,4037	-0,4510	-0,0743
0,1147	0,0645	0,2101	-0,0857	0,0264	0,2519	-0,0395	0,0186	0,0160	-0,4607	0,0383	0,0374
0,0835	-0,0448	-0,0536	0,2044	0,1005	0,0472	0,0702	0,1144	0,1069	0,0892	0,0334	-0,1186
0,1125	-0,0443	-0,0680	0,1907	0,1239	0,0401	0,0997	0,0995	0,1118	0,1223	-0,0279	-0,1275
-0,0773	-0,2278	0,0948	-0,0698	-0,1724	-0,2471	-0,1556	-0,1258	0,1489	0,1247	-0,1160	0,0230
0,0389	-0,1960	0,0366	-0,0949	-0,1368	-0,1639	-0,3284	0,2103	0,1274	-0,0839	-0,1038	-0,0328
0,1273	-0,0317	-0,0854	0,1703	0,1250	0,0097	0,0802	0,0746	0,1065	0,0991	-0,0478	-0,0915
0,0744	-0,0218	0,0113	0,2029	0,0035	0,0128	0,1401	0,3771	-0,0300	-0,1334	0,0590	-0,3132
0,0909	0,3620	0,1916	-0,0010	-0,0738	-0,4673	0,3493	0,0679	0,0114	-0,0259	-0,3291	0,1964
0,0089	0,1421	-0,2469	-0,1159	-0,2943	0,0792	0,0005	-0,1883	-0,0071	-0,0312	-0,0157	-0,4117
0,0030	0,1278	-0,2500	-0,0915	-0,2464	-0,0769	-0,0104	0,2568	-0,6096	-0,0926	0,2547	0,2486
-0,0308	0,0126	-0,4592	-0,4028	0,3161	0,1058	0,0317	0,1632	0,3284	-0,0484	0,1852	-0,0185
-0,0389	0,1064	-0,2639	-0,1924	-0,1180	0,2668	0,0539	-0,0249	0,1765	0,2562	-0,3439	0,3390
0,0015	0,1218	-0,2755	-0,1132	-0,2730	0,1602	0,0053	-0,1400	-0,2318	0,0282	-0,2003	-0,1420
0,0551	0,3307	0,0059	-0,0614	-0,2669	-0,3738	0,0947	-0,0559	0,4065	-0,0337	0,4014	-0,0139

Kolom 13-24

$\lambda_{13}$	$\lambda_{14}$	$\lambda_{15}$	$\lambda_{16}$	$\lambda_{17}$	$\lambda_{18}$	$\lambda_{19}$	$\lambda_{20}$	$\lambda_{21}$	$\lambda_{22}$	$\lambda_{23}$	$\lambda_{24}$
-0,1613	0,0073	0,0457	-0,0073	-0,0513	-0,2666	0,3221	0,3140	0,1661	-0,2708	0,1782	-0,0626
0,0354	-0,0904	-0,0128	-0,0991	-0,0039	0,0538	-0,0842	-0,0526	-0,0562	-0,2305	0,0037	-0,2616
-0,0196	0,1100	0,1912	-0,0428	-0,0988	0,1156	-0,0961	0,0555	0,1991	0,2620	0,0462	0,3641
0,2307	-0,2068	0,0593	0,0091	-0,3031	-0,1068	0,1635	-0,1215	-0,0651	-0,1170	-0,0758	0,0491
-0,1660	0,5396	0,0294	0,2362	0,3156	0,1376	-0,1860	-0,0657	0,0437	-0,0068	0,0358	-0,1303
0,0171	-0,0428	0,0578	-0,1584	0,0442	0,0225	-0,0143	-0,0355	-0,0313	0,0116	-0,0384	-0,0535
-0,0634	-0,1111	-0,4797	0,1268	-0,0074	-0,0866	-0,2723	-0,1952	-0,2489	0,3294	-0,1613	0,1331
0,0352	-0,1195	-0,0422	-0,0133	0,0693	0,1264	-0,3004	-0,1305	0,2577	-0,0794	0,2914	0,3406
0,2360	0,1156	-0,1817	-0,0760	0,0774	0,0150	0,0617	0,1431	-0,2131	-0,0649	-0,0392	-0,0581
0,2170	0,1169	-0,1094	-0,0305	0,0600	0,0229	0,0493	0,1393	-0,0780	0,0391	0,0247	0,0775
0,0531	-0,0250	0,0689	-0,3554	0,1394	-0,3283	-0,1019	-0,3054	0,1756	0,1497	0,0910	-0,2758
0,1880	0,1057	0,1613	-0,2083	0,1410	-0,3230	0,0751	0,1800	-0,0338	0,1200	-0,1066	0,3749
0,2233	0,0816	-0,1581	-0,0552	0,0861	0,0139	0,0447	0,1242	-0,2079	-0,0977	-0,0436	-0,0797
-0,0135	-0,0635	-0,3248	0,0421	0,3437	-0,0234	-0,0024	0,0250	0,1867	-0,3368	0,0802	-0,0841
0,0094	0,0952	0,2294	-0,0262	-0,3753	0,2600	0,0550	-0,1656	-0,2215	0,0057	-0,2639	-0,1108
-0,1054	0,1156	-0,0963	0,1371	-0,1174	-0,1443	0,0502	-0,0354	0,0318	-0,0016	-0,0181	-0,1911
-0,1557	0,1106	-0,0333	0,1853	-0,1701	-0,0985	0,0761	-0,0961	0,1803	0,0951	0,0300	-0,0637
-0,2771	0,0788	-0,0660	0,2636	-0,1300	0,0256	-0,0412	0,4519	-0,1711	-0,0593	-0,0332	0,2047
0,1539	0,0328	-0,0665	0,1218	-0,3056	0,0775	-0,2457	-0,0984	-0,0306	-0,0350	0,2304	-0,1984
-0,1290	0,1089	-0,0585	0,1676	-0,1634	-0,0741	0,0358	-0,1024	0,1756	0,0995	0,0091	-0,0747
-0,2121	0,2496	-0,1033	-0,3060	-0,0823	-0,1634	0,0269	-0,2305	-0,1329	-0,2064	-0,1835	0,3118
-0,1026	-0,1286	0,2262	0,1311	0,2112	-0,1364	0,1455	0,1741	-0,0342	0,5335	-0,0406	-0,2693
-0,0334	-0,2107	0,1481	-0,0180	0,1680	0,0224	-0,0740	-0,0089	-0,1785	-0,1055	-0,0692	-0,0399
-0,1142	-0,2370	0,2586	0,0566	0,1560	0,0417	-0,1178	-0,0845	0,0022	-0,0570	-0,0126	0,1316
-0,1799	-0,2784	-0,2485	-0,0114	0,1907	0,3902	0,5191	-0,1627	0,0544	0,0884	-0,1742	0,0627
0,1269	-0,1341	0,2388	0,1067	-0,0631	0,0195	-0,1474	0,1460	-0,0766	-0,1935	-0,0128	-0,0902
-0,1108	-0,2293	0,2370	0,0907	0,1429	0,0413	-0,1440	-0,0665	-0,0199	-0,0887	-0,0075	0,0683
0,0693	0,0057	-0,1104	-0,3007	-0,2366	0,3644	0,1017	0,1927	0,3181	0,1629	0,2745	-0,0210
0,4048	0,1113	0,0961	0,2501	0,1175	0,0683	0,0266	0,0116	0,0435	0,0438	-0,0526	0,0253
0,1426	-0,2528	-0,1687	0,1979	-0,1669	-0,2424	-0,1086	0,0971	0,3970	-0,0316	-0,3819	0,0203
-0,2099	0,0360	0,2041	-0,0227	0,0358	0,2258	-0,0081	-0,0527	0,0534	-0,1805	-0,1661	-0,0161
0,2332	0,2863	0,1177	0,0365	0,0851	0,0228	0,2789	-0,2503	0,0842	-0,0118	-0,0349	0,0404
-0,0132	-0,0291	-0,0927	-0,3679	0,0485	0,1968	-0,2618	0,3209	-0,0671	0,1551	-0,0987	-0,1707
0,0415	-0,1610	-0,0221	0,1681	-0,0123	-0,0662	0,1809	-0,1334	-0,3687	0,0783	0,5825	0,1259
-0,3503	0,0126	0,0052	-0,2281	-0,1448	-0,2042	-0,0070	0,0183	-0,1339	0,0009	0,1371	-0,0785

Kolom 25-35

$\lambda_{25}$	$\lambda_{26}$	$\lambda_{27}$	$\lambda_{28}$	$\lambda_{29}$	$\lambda_{30}$	$\lambda_{31}$	$\lambda_{32}$	$\lambda_{33}$	$\lambda_{34}$	$\lambda_{35}$
0,2604	0,0759	0,0495	0,0186	0,0297	-0,0031	0,0397	-0,0197	0,0062	-0,0058	-0,1690
-0,0597	-0,1193	-0,3084	0,0115	-0,6145	-0,1418	-0,0738	-0,0149	-0,0600	-0,0220	-0,1690
-0,0805	0,2773	0,4154	-0,3651	-0,0887	0,0968	0,0102	0,0178	-0,0234	-0,0160	-0,1690
0,0041	0,0709	0,0702	-0,0095	-0,0624	0,0381	-0,0102	-0,0140	0,0160	-0,0039	-0,1690
0,0060	-0,0319	-0,0714	0,0105	-0,0461	-0,0012	0,0107	0,0036	0,0075	0,0032	-0,1690
-0,1122	-0,2237	-0,1342	0,2890	0,6731	0,0312	0,0146	0,0315	0,0537	0,0517	-0,1690
-0,0804	0,0745	-0,0593	0,0171	-0,0347	0,0059	-0,0101	0,0009	0,0010	0,0001	-0,1690
0,2719	0,0829	-0,3893	-0,0557	0,0538	0,0242	0,0325	0,0003	-0,0054	-0,0122	-0,1690
0,1698	0,1446	0,0322	0,0036	0,0418	0,1625	0,0871	0,0288	-0,0271	-0,7699	-0,1690
0,1420	0,2851	0,1362	0,2829	0,0062	-0,5319	-0,3895	-0,2215	-0,0186	0,2561	-0,1690
-0,0040	0,1636	0,0396	-0,0435	0,0023	0,0074	0,0127	0,0017	0,0062	-0,0101	-0,1690
-0,1670	-0,4237	-0,0804	0,0225	-0,1944	0,0025	0,0001	-0,0023	0,0289	-0,0022	-0,1690
0,1462	0,1209	0,0036	-0,0574	-0,0714	0,4121	0,3621	0,2157	0,0834	0,5672	-0,1690
-0,4458	-0,0823	0,4072	-0,0485	0,0117	-0,0164	-0,0263	0,0202	-0,0080	-0,0008	-0,1690
-0,2020	-0,0916	0,0993	0,0030	-0,0286	-0,0136	-0,0232	-0,0016	0,0055	0,0023	-0,1690
-0,0409	-0,0862	-0,1695	-0,3781	0,1404	0,3590	-0,5267	-0,2653	-0,1586	0,0795	-0,1690
-0,1179	0,0635	-0,0598	0,1360	-0,0272	-0,1719	-0,0696	0,7315	0,1392	-0,0666	-0,1690
-0,0343	-0,0966	-0,0908	0,0240	0,0021	-0,0211	0,0094	0,0051	-0,0058	-0,0032	-0,1690
0,4067	-0,2789	0,3493	0,0538	0,0293	-0,0179	-0,0104	0,0203	0,0029	0,0145	-0,1690
-0,1331	-0,0101	-0,0221	0,1303	-0,0238	-0,1524	0,6033	-0,4921	-0,0426	-0,0228	-0,1690
0,1324	0,0697	-0,0935	0,0281	0,0234	-0,0267	-0,0137	-0,0042	0,0084	0,0119	-0,1690
0,1908	0,0480	-0,0817	0,0208	-0,0705	0,0158	0,0063	-0,0016	0,0002	-0,0006	-0,1690
0,0964	-0,2004	-0,0072	-0,5927	0,1909	-0,4726	0,1685	0,1046	-0,0361	0,0134	-0,1690
0,0231	-0,0719	0,1665	0,3553	-0,0797	0,2042	-0,0353	0,0802	-0,6419	0,0016	-0,1690
0,1072	0,0480	-0,0034	-0,0214	-0,0255	0,0108	-0,0140	-0,0241	0,0040	0,0069	-0,1690
-0,3395	0,5359	-0,2499	-0,0095	0,1353	-0,0004	0,0366	-0,0169	-0,0434	-0,0046	-0,1690
0,0070	-0,0444	0,1147	0,1643	-0,1005	0,1927	-0,1427	-0,1843	0,7219	-0,0635	-0,1690
-0,1759	-0,1625	-0,1851	-0,0057	-0,0277	0,0383	-0,0203	0,0007	0,0200	0,0000	-0,1690
-0,0316	-0,1345	-0,0198	-0,0031	0,0462	-0,0160	-0,0177	0,0113	-0,0160	-0,0092	-0,1690
0,0868	-0,0624	-0,0154	-0,0065	0,0327	-0,0129	-0,0015	0,0137	0,0026	0,0003	-0,1690
0,1928	0,0331	0,0084	-0,0030	0,0152	0,0109	0,0054	-0,0010	-0,0057	0,0039	-0,1690
-0,0029	0,0276	-0,0392	-0,0088	0,0310	-0,0024	-0,0079	-0,0179	0,0016	-0,0056	-0,1690
-0,0185	0,0054	0,0764	0,0379	0,0146	-0,0016	-0,0187	0,0041	-0,0079	-0,0017	-0,1690
-0,1457	-0,0443	-0,0247	-0,0240	0,0089	0,0029	0,0036	-0,0046	-0,0094	-0,0015	-0,1690
-0,0516	0,0377	0,1363	0,0232	0,0072	-0,0113	0,0086	-0,0062	0,0004	0,0098	-0,1690

### Lampiran 3. Bobot citra pelatihan

Kolom 1-12 ( $\times 10^6$ )

$\Omega_1$	$\Omega_2$	$\Omega_3$	$\Omega_4$	$\Omega_5$	$\Omega_6$	$\Omega_7$	$\Omega_8$	$\Omega_9$	$\Omega_{10}$	$\Omega_{11}$	$\Omega_{12}$
-2,5155	-2,2911	-2,2944	-4,1463	-2,9012	-2,8775	-2,3766	1,2199	0,6116	0,6356	1,1008	1,3463
1,6057	1,3994	1,4044	1,4511	1,2693	1,6679	1,6542	-0,9879	-0,4797	-0,4803	-0,8790	-1,0095
-0,7166	-0,9417	-0,8763	-0,6803	-0,3515	-1,0621	-0,9778	0,6530	0,3166	0,3165	0,7194	0,8434
0,6346	0,5157	0,5755	0,9347	0,8358	0,6296	0,4652	-0,2837	-0,1206	-0,1277	-0,1925	-0,2462
0,5528	0,4331	0,2304	-0,4118	-0,9870	0,5168	0,6254	-0,0514	-0,0672	-0,1245	0,0997	0,0394
-0,2090	0,0061	-0,0262	0,8005	0,2037	0,0705	-0,1569	0,1530	0,0884	0,0862	0,0156	-0,0314
-0,1008	-0,1309	-0,0914	0,2696	0,1644	-0,1233	-0,0310	0,0248	0,0527	0,0663	0,0575	0,0349
0,2592	-0,0872	-0,1263	0,1223	0,0067	-0,0277	0,4975	-0,0919	-0,0700	-0,0665	0,0106	0,0716
-0,1706	-0,0576	-0,0969	0,2399	0,0816	-0,0716	-0,1113	0,0774	0,0433	0,0423	-0,0184	-0,0176
0,0350	-0,0378	-0,0378	-0,0581	0,0903	-0,0617	-0,0032	0,0067	0,0167	0,0202	-0,0047	-0,0013
0,1665	0,1443	0,1219	0,6902	0,4251	0,2396	0,3000	-0,0989	-0,0290	-0,0251	-0,1227	-0,1236
-0,2450	-0,1262	-0,1149	0,0876	-0,1727	-0,1416	-0,2667	0,2015	0,0766	0,0654	0,1504	0,1655
0,0496	0,0070	-0,0180	0,5457	0,0449	0,0686	0,1020	0,0774	0,0872	0,0776	0,1530	0,0992
-0,1425	-0,2216	-0,1808	-0,1082	0,2612	-0,2522	-0,2796	0,1361	0,1243	0,1346	0,1189	0,1436
-0,0064	-0,0021	0,0128	-0,0297	0,0461	-0,0231	-0,1357	0,0212	-0,0023	-0,0038	-0,0033	0,0125
-0,2393	-0,2410	-0,2019	-0,2395	0,1031	-0,3128	-0,2391	0,1027	0,0475	0,0617	0,0553	0,1086
0,0264	-0,0128	-0,0478	0,1886	0,2008	0,0053	0,0390	0,0353	0,0179	0,0181	-0,0155	0,0098
-0,1307	-0,0655	-0,0132	-0,1929	0,0543	-0,1176	-0,1288	-0,0204	-0,0140	-0,0038	-0,0612	-0,0372
0,0646	-0,0313	-0,0557	0,1296	-0,0456	-0,0165	-0,0215	0,0133	0,0080	0,0016	0,0342	0,0384
0,0168	-0,0362	-0,0073	-0,2055	-0,0543	-0,0635	-0,1404	-0,0038	0,0229	0,0232	0,0043	-0,0024
0,1223	0,0203	0,0427	0,0555	0,0252	0,0352	-0,0083	-0,0092	-0,0187	-0,0191	0,0387	0,0307
-0,1478	-0,0989	-0,0767	-0,1838	-0,2431	-0,1015	0,0843	0,0178	0,0193	0,0212	0,0507	0,0391
0,1485	0,0783	0,0841	0,1015	0,0864	0,1016	0,0491	-0,0558	-0,0133	-0,0131	-0,0215	-0,0491
0,0447	-0,0014	0,0232	0,0340	-0,0650	0,0221	0,1303	-0,0092	-0,0218	-0,0200	0,0182	0,0252
0,0404	-0,0311	-0,0355	0,0660	0,1200	-0,0269	-0,0129	0,0077	0,0359	0,0371	0,0029	0,0046
-0,0047	0,0573	0,1318	0,4152	0,3394	0,0541	-0,0491	-0,0200	0,0159	0,0242	-0,0709	-0,1012
0,0375	0,0595	0,1128	0,0607	0,0086	0,0619	0,0090	-0,0235	-0,0060	-0,0029	-0,0372	-0,0644
-0,0451	-0,0694	-0,1214	-0,0608	-0,0679	-0,0450	-0,0578	0,0801	0,0538	0,0502	0,0948	0,0984
0,0578	0,0464	0,0367	-0,0599	-0,0430	0,0897	0,0680	-0,0473	-0,0323	-0,0337	-0,0510	-0,0497
-0,0052	-0,0172	-0,0046	0,0444	-0,0088	-0,0119	-0,0392	0,0353	0,0184	0,0101	0,0442	0,0389
0,0093	-0,0168	-0,0272	-0,0003	-0,0141	-0,0061	-0,0035	0,0184	0,0085	0,0041	0,0224	0,0280
-0,0008	-0,0062	0,0043	-0,0297	-0,0211	-0,0006	-0,0015	0,0037	0,0030	0,0027	0,0090	0,0031
0,0188	0,0167	0,0144	0,0676	0,0412	0,0194	0,0199	-0,0170	-0,0128	-0,0142	-0,0242	-0,0193
0,0140	0,0063	0,0069	-0,0091	-0,0050	0,0090	0,0072	-0,0048	-0,0095	-0,0060	-0,0045	-0,0059

Kolom 13-24 ( $\times 10^6$ )

$\Omega_{13}$	$\Omega_{14}$	$\Omega_{15}$	$\Omega_{16}$	$\Omega_{17}$	$\Omega_{18}$	$\Omega_{19}$	$\Omega_{20}$	$\Omega_{21}$	$\Omega_{22}$	$\Omega_{23}$	$\Omega_{24}$
0,6893	1,2002	2,1875	1,5958	1,7323	1,8553	2,0443	1,6664	1,9589	1,3750	0,0720	0,6196
-0,5104	-0,9156	-1,2399	-0,8933	-0,9893	-1,1101	-1,2257	-0,9653	-1,0729	-0,9638	0,0892	-0,2963
0,3324	0,4944	0,8630	0,5444	0,6316	0,7229	0,8727	0,6257	0,7344	0,4960	-0,2912	0,0812
-0,1446	-0,3509	-0,4231	-0,3085	-0,3445	-0,4029	-0,3952	-0,3306	-0,3793	-0,3857	-0,1057	-0,1954
0,0088	0,1274	-0,0059	-0,0337	-0,0329	0,0151	0,0287	0,0105	-0,0634	0,0169	0,1995	0,2837
0,0697	0,1863	-0,1838	-0,1238	-0,1095	-0,0668	-0,1407	-0,1057	-0,1400	0,1769	-0,0258	-0,0136
0,0337	-0,0106	-0,0420	-0,0594	-0,0414	-0,0102	-0,0249	-0,0458	-0,0095	0,0273	-0,0825	-0,0574
-0,0640	-0,0663	-0,1573	-0,1622	-0,1697	-0,1361	-0,1045	-0,1537	-0,1087	-0,0517	0,0992	0,0177
0,0406	0,0948	-0,0779	-0,0181	-0,0398	-0,0339	-0,0582	-0,0400	-0,0728	0,0972	0,1037	0,0332
0,0201	-0,0234	0,0122	0,0226	0,0122	-0,0032	0,0049	0,0056	-0,0414	-0,0186	0,0446	0,0261
-0,0524	-0,0850	-0,3224	-0,2660	-0,2685	-0,2429	-0,2858	-0,2560	-0,3150	-0,0879	0,0021	-0,0825
0,0842	0,1855	0,1608	0,1279	0,1417	0,1522	0,1653	0,1407	0,1046	0,1666	-0,0570	0,0146
0,0788	0,0664	-0,1086	-0,1065	-0,0954	-0,0581	-0,0481	-0,0814	-0,1031	0,0654	-0,0475	-0,0181
0,1106	0,0684	0,1672	0,0736	0,1109	0,1441	0,1570	0,1025	0,1396	0,0723	-0,1421	-0,0128
-0,0009	0,0205	0,0303	0,0424	0,0281	0,0132	0,0239	0,0259	0,0025	-0,0067	0,0188	0,0023
0,0437	0,0393	0,1736	0,1415	0,1379	0,1346	0,1532	0,1261	0,1296	0,0637	-0,0071	0,0069
0,0148	0,0328	-0,0908	-0,0524	-0,0686	-0,0687	-0,0651	-0,0669	-0,0969	0,0098	0,0972	0,0213
-0,0190	-0,0377	0,0521	0,0631	0,0505	0,0138	0,0241	0,0358	0,0183	-0,0229	0,0351	-0,0317
0,0134	0,0416	-0,0506	-0,0347	-0,0424	-0,0306	-0,0214	-0,0361	-0,0291	0,0273	0,0385	-0,0096
0,0249	-0,0085	0,0811	0,0551	0,0640	0,0668	0,0599	0,0559	0,0549	-0,0033	-0,0573	-0,0163
-0,0161	0,0112	-0,0092	-0,0247	-0,0180	-0,0147	0,0014	-0,0160	-0,0123	-0,0260	-0,0463	-0,0216
0,0162	0,0185	0,0685	0,0350	0,0519	0,0612	0,0629	0,0483	0,0560	0,0743	0,0050	0,0395
-0,0156	-0,0410	-0,0729	-0,0691	-0,0628	-0,0554	-0,0657	-0,0609	-0,0585	-0,0565	-0,0480	-0,0246
-0,0192	0,0027	-0,0228	-0,0296	-0,0244	-0,0115	-0,0100	-0,0218	0,0163	-0,0041	0,0138	0,0166
0,0300	-0,0186	-0,0225	-0,0252	-0,0207	-0,0156	-0,0153	-0,0215	-0,0114	-0,0060	-0,0075	-0,0106
-0,0008	-0,0532	-0,1360	-0,0658	-0,0835	-0,1147	-0,1328	-0,0961	-0,1289	-0,0512	-0,0181	-0,0855
-0,0033	-0,0017	-0,0383	-0,0246	-0,0215	-0,0303	-0,0475	-0,0287	-0,0482	-0,0222	-0,0014	0,0046
0,0487	0,0653	0,0943	0,0290	0,0580	0,0946	0,0973	0,0673	0,0787	0,0726	-0,0894	0,0286
-0,0296	-0,0311	-0,0375	-0,0216	-0,0274	-0,0359	-0,0396	-0,0279	-0,0260	-0,0327	0,0455	0,0049
0,0258	0,0294	0,0327	0,0203	0,0236	0,0301	0,0344	0,0247	0,0192	0,0254	-0,0540	0,0004
0,0174	0,0194	0,0133	-0,0031	0,0064	0,0162	0,0187	0,0092	0,0106	0,0153	-0,0019	0,0054
0,0097	0,0058	0,0132	0,0005	0,0108	0,0126	0,0111	0,0065	0,0115	0,0029	-0,0060	0,0034
-0,0129	-0,0187	-0,0386	-0,0131	-0,0256	-0,0350	-0,0346	-0,0253	-0,0342	-0,0147	0,0191	-0,0181
0,0053	-0,0006	-0,0007	-0,0017	-0,0010	-0,0010	-0,0019	-0,0020	-0,0008	-0,0064	0,0039	0,0006

Kolom 25-35 ( $\times 10^6$ )

$\Omega_{25}$	$\Omega_{26}$	$\Omega_{27}$	$\Omega_{28}$	$\Omega_{29}$	$\Omega_{30}$	$\Omega_{31}$	$\Omega_{32}$	$\Omega_{33}$	$\Omega_{34}$	$\Omega_{35}$
0,9600	1,3503	0,5704	0,1189	0,4199	-1,0183	-1,0237	-1,0938	-0,7370	-0,9565	-1,0978
-0,6772	-0,8885	-0,2534	0,2569	0,0490	0,9080	1,0281	0,6290	0,4652	0,8458	1,1147
0,5117	0,6625	0,0100	0,1379	0,7039	-0,8795	-0,5242	-1,3817	-0,9618	-0,9134	-0,7156
-0,1123	-0,1950	-0,1710	0,0786	0,2938	0,1119	0,2919	-0,3498	-0,1302	0,0720	0,2560
-0,3335	-0,3031	0,1976	0,0004	-0,2515	-0,3607	-0,4724	0,8697	0,0381	-0,3042	-0,4909
-0,1423	-0,1539	-0,0479	-0,4062	-0,8050	0,1057	-0,3382	0,7976	0,5250	0,1718	-0,2304
-0,0529	-0,0426	-0,0935	0,1362	0,0717	0,0520	0,0456	-0,0359	-0,0506	0,0325	0,0674
-0,0724	-0,0922	0,0047	0,3982	0,2664	-0,0622	0,1915	0,2043	-0,2731	-0,1120	0,1060
0,0367	0,0170	0,0677	-0,2484	-0,4037	0,0251	-0,2049	0,5149	0,2706	0,0531	-0,0976
0,0540	0,0464	0,0473	-0,0935	0,0156	-0,0286	-0,0330	-0,0029	0,0112	-0,0271	-0,0154
-0,1806	-0,2270	-0,1090	0,0467	-0,2050	0,1454	0,1729	0,4674	0,1448	0,1247	0,1940
0,1257	0,1502	0,0241	-0,1947	-0,1133	-0,2391	-0,2183	-0,0964	0,0006	-0,2083	-0,2970
-0,1050	-0,0914	-0,0602	0,0252	-0,0765	-0,1338	-0,2349	0,2556	-0,0443	-0,1306	-0,2420
0,0006	0,0579	-0,0814	0,0165	0,1501	-0,1485	-0,0483	-0,2752	-0,1944	-0,1709	-0,0313
0,0734	0,0819	0,0328	-0,0620	0,0156	-0,0543	0,0054	-0,0757	-0,0316	-0,0532	-0,0187
0,1865	0,2161	0,0417	-0,0851	0,1673	-0,0695	-0,0084	-0,3485	-0,1420	-0,0772	-0,0290
0,0439	0,0205	0,0446	-0,1603	-0,1459	-0,0600	-0,0790	0,2791	0,0519	-0,0500	-0,0764
0,1467	0,1253	0,0180	0,0257	0,1110	0,0303	0,1559	-0,1935	-0,0156	0,0196	0,1259
0,0282	0,0026	-0,0027	0,0793	-0,0279	-0,0403	-0,0305	0,1314	-0,0390	-0,0405	-0,0458
0,0081	0,0308	-0,0179	0,0679	0,1043	0,0132	0,0464	-0,2065	-0,0361	0,0039	0,0551
-0,0174	-0,0164	-0,0412	0,1464	0,1073	-0,0421	0,0360	-0,1038	-0,1247	-0,0703	0,0033
-0,0376	-0,0268	0,0137	0,0910	0,0063	0,0060	-0,0466	0,0759	0,0193	0,0112	-0,0304
-0,0992	-0,0969	-0,0529	0,1030	0,0180	0,0613	0,0441	0,0174	-0,0067	0,0668	0,0792
-0,0352	-0,0356	-0,0008	0,0562	0,0420	-0,0244	-0,0051	0,0284	-0,0585	-0,0216	-0,0318
-0,0155	-0,0225	-0,0250	-0,0186	0,0093	-0,0034	0,0093	0,0110	-0,0102	-0,0107	0,0128
0,0415	0,0198	-0,0355	-0,1641	-0,1085	0,0814	0,0326	0,0218	0,1495	0,0898	0,0461
-0,0562	-0,0597	-0,0072	-0,0662	-0,1049	0,0465	-0,0062	0,1106	0,1145	0,0584	0,0180
-0,0589	-0,0196	-0,0225	0,0204	0,0039	-0,0864	-0,1020	-0,0503	-0,0645	-0,0857	-0,0893
-0,0089	-0,0224	0,0152	0,0031	-0,0184	0,0499	0,0498	0,0680	0,0406	0,0519	0,0479
-0,0111	0,0089	-0,0078	-0,0018	0,0108	-0,0560	-0,0560	-0,0650	-0,0301	-0,0521	-0,0564
-0,0023	-0,0052	-0,0090	0,0089	0,0007	-0,0336	-0,0218	0,0044	-0,0295	-0,0339	-0,0282
-0,0130	-0,0141	-0,0128	0,0110	0,0061	-0,0045	-0,0023	-0,0071	-0,0041	-0,0039	-0,0034
0,0277	0,0245	0,0205	-0,0165	-0,0064	0,0142	0,0183	0,0134	0,0178	0,0138	0,0137
-0,0026	-0,0045	-0,0018	0,0023	0,0025	0,0010	0,0042	0,0016	-0,0011	0,0011	0,0048