



TESIS - IF 185401

Pengenalan Wajah Menggunakan Metode Deep Neural Networks dengan Perpaduan Metode Discrete Wavelet Transform, Stationary Wavelet Transform, dan Discrete Cosine Transform

AFRIZAL LAKSITA AKBAR
05111550010005

Dosen Pembimbing
Dr. Eng. Chastine Fatichah, S.Kom, M.Kom
Dr. Ahmad Saikhu, S.Si., MT.

Departemen Teknik Informatika
Fakultas Teknologi Elektro Dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN TESIS

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom)

di

Institut Teknologi Sepuluh Nopember

Oleh:

AFRIZAL LAKSITA AKBAR

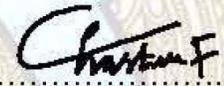
NRP: 05111550010005

Tanggal Ujian : 20 Juli 2020

Periode Wisuda : 2020 Ganjil

Disetujui oleh :
Pembimbing:

1. Dr. Eng. Chastine Fatichah, S.Kom., M.Kom.
NIP. 197512202001122002

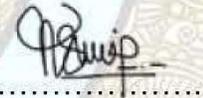


2. Dr. Ahmad Saikhu S.Si., MT.
NIP. 19710718 2006041001



Penguji:

1. Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
NIP. 19710428 199422000



2. Dr. Eng. Radityo Anggoro, S.Kom., M.Sc.
NIP. 19841016 2008121002



3. Shintami Chusnul Hidayati, S. Kom., M.Sc., Ph.D.
NIP. 1987202012004



Kepala Departemen Teknik Informatika
Fakultas Teknologi Elektro Dan Informatika Cerdas



Dr. Eng. Chastine Fatichah, S.Kom., M.Kom.

NIP. 197512202001122002

[Halaman ini sengaja dikosongkan]

PENGENALAN WAJAH MENGGUNAKAN METODE DEEP NEURAL
NETWORKS DENGAN PERPADUAN METODE DISCRETE WAVELET
TRANSFORM, STATIONARY WAVELET TRANSFORM, DAN DISCRETE
COSINE TRANSFORM

Nama mahasiswa : Afrizal Laksita Akbar
NRP : 05111550010005
Pembimbing : Dr. Eng. Chastine Fatichah S.Kom, M.Kom
Dr. Ahmad Saikhu S.Si., MT.

ABSTRAK

Metode pengenalan identitas dilakukan dengan menggunakan wajah, sidik jari, telapak tangan, retina mata, atau suara yang umum dikenal dengan metode *biometric*. Wajah adalah organ tubuh manusia yang paling sering dijadikan indikasi pengenalan seseorang. Dalam pengembangan sistem pengenalan wajah terdapat beberapa isu yang harus diperhatikan, karena dalam proses pengenalan wajah terdapat beberapa faktor yang mempengaruhi, yaitu faktor pencahayaan, ekspresi wajah dan perubahan atribut wajah antara lain dagu, kumis, dan aksesoris yang digunakan misalnya kacamata atau syal.

Pada penelitian ini, diusulkan penggabungan metode *Discrete Wavelet Transform* dan *Stationary Wavelet Transform* untuk meningkatkan kualitas citra khususnya pada gambar berukuran kecil. Sedangkan metode *Histogram Equalization* dapat memperbaiki citra pada kondisi citra dengan kelebihan atau kekurangan intensitas cahaya. Metode *Discrete Cosine Transform* digunakan untuk mengubah citra wajah ke dalam bentuk citra frekuensi untuk ekstraksi fitur pada metode klasifikasi *Deep Neural Networks*.

Pengujian dilakukan dengan *10 fold Cross Validation*. Hasil penelitian menunjukkan bahwa penggabungan 4 metode yang diusulkan diperoleh tingkat akurasi yang paling baik sebesar 92.73% dibandingkan dengan metode *Histogram Equalization* 80.73%, *Discrete Wavelet Transform* 85.85%, *Stationary Wavelet Transform* 64.27%, *Discrete Cosine Transform* 89.50%, penggabungan *Discrete Wavelet Transform* dan *Stationary Wavelet Transform* 86.89%, penggabungan *Histogram Equalization*, *Discrete Wavelet Transform*, dan *Stationary Wavelet Transform* 69.77%, dan *Stationary Wavelet Transform*, *Discrete Wavelet Transform*, dan *Histogram Equalization* 77.39%.

Kata kunci : Pengenalan Wajah, *Histogram Equalization*, *Discrete Wavelet Transform*, *Stationary Wavelet Transform*, *Discrete Cosine Transform*, *Deep Neural Networks*

[Halaman ini sengaja dikosongkan]

FACE RECOGNITION USING DEEP NEURAL NETWORKS WITH THE
COMBINATION OF DISCRETE WAVELET TRANSFORM, STATIONARY
WAVELET TRANSFORM, AND DISCRETE COSINE TRANSFORM

By : Afrizal Laksita Akbar
Student Identity Number : 05111550010005
Supervisor : Dr. Eng. Chastine Fatichah S.Kom, M.Kom
Dr. Ahmad Saikhu S.Si., MT.

ABSTRACT

Personal identification can be done by using face, fingerprint, palm prints, eye's retina, or voice recognition which commonly called as biometric methods. The face recognition is the most popular and widely used among those biometric methods. However, there are some issues in the implementation of this method. They are lighting factor, facial expression and attributes (chin, mustache, or wearing some accessories).

In this study, we propose a combination method of Discrete Wavelet Transform and Stationary Wavelet Transform that able to improve the image quality, especially in the small-sized image. Moreover, we also use Histogram Equalization in order to correct noises such as over or under exposure, Discrete Cosine Transform in order to transform the image into frequency domain, and Deep Neural Networks in order to perform the feature extraction and classify the image. A 10-fold cross-validation method was used in this study.

As the result, the proposed method showed the highest accuracy up to 92.73% compared to Histogram Equalization up to 80.73%, Discrete Wavelet Transform up to 85.85%, Stationary Wavelet Transform up to 64.27%, Discrete Cosine Transform up to 89.50%, the combination of Histogram Equalization, Discrete Wavelet Transform, and Stationary Wavelet Transform up to 69.77%, and the combination of Stationary Wavelet Transform, Discrete Wavelet Transform, and Histogram Equalization up to 77.39%.

Keywords : Face Recognition, Discrete Wavelet Transform, Stationary Wavelet Transform, Discrete Cosine Transform, Deep Neural Networks

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

LEMBAR PENGESAHAN TESIS	iii
ABSTRAK.....	v
ABSTRACT	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	5
1.6 Kontribusi Penelitian.....	5
BAB 2 KAJIAN PUSTAKA	7
2.1 Klasifikasi Citra Wajah	7
2.2 <i>Preprocessing</i>	8
2.3 <i>Histogram Equalization (HE)</i>	8
2.4 <i>Discrete Wavelet Transform (DWT)</i>	9
2.5 <i>Stationary Wavelet Transform (SWT)</i>	11
2.6 <i>Discrete Cosine Transform (DCT)</i>	12
2.7 <i>Artificial Neural Networks (ANN)</i>	13
2.8 <i>K-fold Cross-Validation</i>	17
2.9 Metode Evaluasi.....	18
2.10 Keras.....	18
BAB 3 METODE PENELITIAN	21
3.1 Studi Literatur dan Analisa Awal.....	21
3.2 Desain Model Sistem.....	21
3.2.1 Data.....	22

3.2.2	<i>Preprocessing</i>	23
3.2.3	Peningkatan Resolusi	23
3.2.4	Perbaikan Citra	26
3.2.5	Transformasi Citra	27
3.2.6	Klasifikasi	28
3.3	Implementasi	30
3.4	Skenario Uji Coba	30
BAB 4 HASIL DAN PEMBAHASAN		33
4.1	Lingkungan Implementasi	33
4.2	Pembuatan <i>Dataset</i> Klasifikasi <i>10-fold Cross-Validation</i>	33
4.3	Uji Coba.....	34
4.3.1	Uji Coba Jumlah <i>Epoch</i>	34
4.3.2	Uji Coba Data Normal	35
4.3.3	Uji Coba Data <i>Histogram Equalization (HE)</i>	36
4.3.4	Uji Coba Data <i>Discrete Wavelet Transform (DWT)</i>	37
4.3.5	Uji Coba Data <i>Stationary Wavelet Transform (SWT)</i>	38
4.3.6	Uji Coba Data <i>Discrete Cosine Transform (DCT)</i>	39
4.3.7	Uji Coba Data DWT dan SWT.....	40
4.3.8	Uji Coba Data HE, DWT, dan SWT.....	41
4.3.9	Uji Coba Data DWT, SWT, dan HE.....	42
4.3.10	Uji Coba Data DWT, SWT, HE, dan DCT	43
4.4	Pembahasan	45
BAB 5 KESIMPULAN DAN SARAN		49
5.1	Kesimpulan	49
5.2	Saran.....	49
DAFTAR PUSTAKA		51
LAMPIRAN		56
BIODATA PENULIS.....		63

DAFTAR GAMBAR

Gambar 2.1 Dekomposisi Wavelet Tiga Tingkat	10
Gambar 2.2 Citra Dekomposisi DWT	11
Gambar 2.3 Distribusi Frekuensi Pada Blok DCT	12
Gambar 2.4 Arsitektur CNN	16
Gambar 2.5 <i>K-fold Cross Validarion</i>	18
Gambar 3.1 Tahapan Penelitian.....	21
Gambar 3.2 Diagram Alir Sistem Usulan	22
Gambar 3.3 Citra <i>Dataset</i>	22
Gambar 3.4 Citra Hasil <i>Grayscale</i>	23
Gambar 3.5 Alur Perbaikan Citra	24
Gambar 3.6 Citra Hasil HE	27
Gambar 3.7 Citra Hasil DCT.....	27
Gambar 3.8 Alur Transformasi Citra	28
Gambar 3.9 Arsitektur <i>Deep Neural Networks</i> (DNNs)	29
Gambar 3.10 Alur proses <i>K-Fold Cross Validation</i>	31
Gambar 4.1 Gambar Asli	34
Gambar 4.2 Gambar <i>Preprocessing</i>	35
Gambar 4.3 Gambar HE.....	36
Gambar 4.4 Gambar DWT	37
Gambar 4.5 Gambar SWT.....	38
Gambar 4.6 Gambar DCT	39
Gambar 4.7 Gambar DWT - SWT.....	40
Gambar 4.8 Gambar HE - DWT - SWT	41
Gambar 4.9 Gambar DWT - SWT - HE	42
Gambar 4.10 Gambar DWT - SWT - HE - DCT.....	44
Gambar 4.11 Grafik Perbandingan Hasil Akurasi Setiap Metode.....	48

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 3.1 <i>Pseudocode</i> gabungan DWT dan SWT	25
Tabel 3.2 <i>Pseudocode</i> HE	27
Tabel 3.3 <i>Pseudocode</i> DCT.....	28
Tabel 3.4 <i>Pseudocode</i> DNNs	29
Tabel 4.1 Hasil Uji Jumlah <i>Epoch</i>	34
Tabel 4.2 Hasil Uji Coba Data Normal.....	35
Tabel 4.3 Hasil Uji Coba Data <i>Histogram Equalization</i> (HE).....	36
Tabel 4.4 Hasil Uji Coba Data <i>Discrete Wavelet Transform</i> (DWT).....	37
Tabel 4.5 Hasil Uji Coba Data <i>Stationary Wavelet Transform</i> (SWT)	38
Tabel 4.6 Hasil Uji Coba Data <i>Discrete Cosine Transform</i> (DCT)	39
Tabel 4.7 Hasil Uji Coba Data DWT dan SWT	40
Tabel 4.8 Hasil Uji Coba Data HE, DWT, dan SWT	42
Tabel 4.9 Hasil Uji Coba Data DWT, SWT, dan HE	43
Tabel 4.10 Hasil Uji Coba Data DWT, SWT, HE, dan DCT.....	44
Tabel 4.11 Perbandingan Waktu Proses Klasifikasi	45
Tabel 4.12 Perbandingan Hasil Akurasi Setiap Metode	47

[Halaman ini sengaja dikosongkan]

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Beberapa tahun ini perkembangan di bidang teknologi semakin cepat. Dalam perkembangan tersebut memudahkan orang dalam melakukan segala kegiatan melalui perangkat lunak atau gawai. Kecenderungan peningkatan tersebut mengakibatkan banyaknya terjadi kejahatan pada dunia digital. Pada kegiatan yang berisiko tinggi akan membutuhkan metode tambahan untuk pengamanan data. Metode pengamanan data melalui identifikasi pengguna dilakukan dengan menggunakan pengenalan wajah, sidik jari, telapak tangan, retina mata, atau suara yang dikenal dengan metode *biometric*. Metode *biometric* sendiri memiliki kelebihan dimana tiap-tiap orang memiliki variasi yang tidak mungkin ditiru dan selalu di bawa oleh manusia tersebut.

Wajah adalah organ tubuh manusia yang paling umum dijadikan indikasi pengenalan seseorang atau *face recognition* (Marti, 2010). Pengenalan wajah dilakukan dengan perbandingan citra wajah seseorang yang akan dikenali dengan citra wajah yang sudah ada. Kualitas citra pada saat melakukan pengenalan wajah dapat menurunkan kinerja. Kualitas citra berkurang dapat disebabkan karena jarak dari kamera atau kualitas citra gambar (Gao, 2020). Metode pengenalan wajah bergantung pada kondisi pencahayaan (Zhou, 2019). Aktivitas manusia yang berbeda dapat memengaruhi kinerja pengenalan wajah (Manju, 2019). Di era *big data* dibutuhkan metode yang dapat mengatasi perubahan data yang sering berubah-ubah, tingkat pengenalan wajah yang rendah, peningkatan konsumsi waktu, dan tidak dapat beradaptasi dengan berbagai kondisi pencahayaan (Qu, 2018). Dalam pengenalan wajah memiliki tingkat kesulitan tertentu diantaranya ekspresi atau wajah setelah operasi plastik dan gambar wajah menggunakan aksesoris kaca mata (Mulyono, 2019).

Tahap dalam proses pengenalan wajah terdiri atas ekstraksi fitur dan proses klasifikasi. Ekstraksi fitur adalah metode paling mendasar sebelum melakukan

pengenalan gambar/citra. Ekstraksi fitur dibagi menjadi dua kategori yaitu, yang berdasarkan tata letak dan statistik (Wang, 2010). Ekstraksi fitur berdasarkan pada tata letak adalah metode dengan menggunakan jarak, sudut, dan letak komponen wajah antara lain, mata, hidung, dan mulut. Sedangkan pada ekstraksi fitur berdasarkan statistik diperoleh dari hasil perhitungan algoritma. Algoritma yang digunakan untuk ekstraksi fitur antara lain, *Discrete Wavelet Transform* (DWT), *Stationary Wavelet Transform* (SWT), dan *Discrete Cosine Transform* (DCT), Metode *Discrete Wavelet Transform* (DWT) merupakan metode yang membantu dalam mengubah gambar skala abu-abu ke domain spasial dan frekuensi pada saat yang sama (Vishwakarma, 2018). Metode *Stationary Wavelet Transform* (SWT) merupakan metode yang dirancang untuk mengatasi *translation invariance* dari DWT dikarenakan tidak ada *downsampling* (Syahdeini, 2015). Sedangkan *Discrete Cosine Transform* (DCT) merupakan metode yang mengubah sinyal apa pun dalam komponen frekuensi dasar (Krasmla, 2017).

Tahap selanjutnya adalah proses klasifikasi yang bertujuan untuk pengelompokan atau membuat segmentasi mengenai data yang saling berkaitan atau homogen. Hasil dari tahap ini adalah ekstraksi fitur dan dilakukan klasifikasi. Algoritma klasifikasi digunakan untuk proses pengenalan menggunakan data dengan target/kelas/label berupa nilai kategori. Metode klasifikasi pada pengenalan wajah antara lain *Principle Component Analysis* (Mohammed, 2018), *Support Vector Machine* (Senthilkumar, 2017), *Partial face recognition* (Reshma, 2018), *Neural Network* (Khan, 2019).

Sistem Pengenalan wajah digunakan di banyak bidang, antara lain memverifikasi kartu kredit, mengidentifikasi pelaku kejahatan, untuk identifikasi data pribadi, sampai dengan keamanan pintu masuk (Mahmud, 2015). Dalam pengembangan sistem pengenalan wajah ada beberapa hal yang harus diperhatikan, di antaranya adalah faktor pencahayaan, ekspresi wajah dan perubahan atribut wajah antara lain dagu, kumis, dan kacamata (Ravi, 2013). Untuk mengatasi masalah pada faktor pencahayaan dilakukan perbaikan citra menggunakan metode *Histogram Equalization*, metode *Histogram Equalization* dapat digunakan untuk meningkatkan kontras citra dan kualitas citra, sehingga informasi yang ada pada

citra yang telah diperbaiki akan terlihat lebih jelas (Mau, 2016). Untuk mengatasi kualitas citra pada proses pengenalan wajah dapat menggunakan gabungan metode DWT dan SWT, dimana hasil dari kombinasi metode tersebut dapat menghasilkan citra resolusi tinggi (Hemalatha, 2017). Sedangkan untuk proses pengenalan wajah diperlukan sistem yang efisien dan bisa beradaptasi dengan cara melakukan pembelajaran mandiri.

Deep Learning (DL) adalah metode pembelajaran mesin yang memungkinkan komputer dapat menjalankan aktivitas seperti sifat manusia. Metode DL digunakan untuk mengolah sistem informasi berbasis non-linier untuk melakukan ekstraksi fitur, pengenalan pola, dan klasifikasi dengan menggunakan *hidden layer* (Deng, 2014). Dengan menggunakan metode DL akan meminimalisasi masalah ketika terjadi perubahan pada atribut wajah karena metode ini akan terus belajar mengenali setiap fitur.

Deep Neural Networks (DNNs) dan *Convolutional Neural Network* (CNN) yang merupakan pengembangan dari metode *Artificial neural networks* (ANN) banyak digunakan pada penelitian terdahulu karena memiliki tingkat akurasi yang relatif tinggi dan memiliki hasil yang signifikan dalam pengenalan citra. Menurut penelitian pada pengenalan wajah dari kombinasi DWT dan CNN didapatkan hasil akurasi terbaik sebesar 93%, dibandingkan dengan menggunakan metode *Principal Component Analysis* (PCA) – DWT dan *Linear Discriminant Analysis* (LDA) – DWT dengan nilai akurasi terbaik sebesar 84% dan 85% (Tabassuma, 2020). Pada penelitian lain dengan menggunakan citra kanker menggunakan metode DCT dengan CNN menghasilkan nilai akurasi 94% dibandingkan dengan kombinasi metode *Cell CNN* (C-CNN) yang menghasilkan akurasi 84,8% (Gehlot, 2020). Sedangkan dalam penelitian tentang pengklasifikasian citra otak *Magnetic Resonance Imaging* (MRI) untuk menentukan tingkatan penyakit *astrocitoma* didapatkan hasil akurasi 96%, sedangkan metode *Radial Basis Function Neural Network* (RBF NN) menghasilkan nilai akurasi 86,6% (Suja, 2018).

Dalam penelitian ini diusulkan sebuah pengembangan metode untuk klasifikasi citra wajah dengan kombinasi metode DWT, SWT, dan DCT pada metode klasifikasi DNNs. Metode ini digunakan untuk memperbaiki kualitas citra

dan mengekstrak fitur citra menggunakan domain frekuensi dalam mengelompokkan wajah. Metode ini diharapkan hasil akurasi yang lebih tepat dan dalam waktu singkat dibanding dengan metode lainnya.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya maka terdapat beberapa permasalahan yang diangkat dalam penelitian ini antara lain:

1. Apakah gabungan metode Discrete Wavelet Transform (DWT) dan Stationary Wavelet Transform (SWT) dapat meningkatkan kemampuan pendeteksian citra wajah?
2. Bagaimana menempatkan metode *Histogram Equalization* (HE) untuk memperbaiki citra wajah?
3. Apakah menggabungkan metode *Histogram Equalization* (HE), *Discrete Wavelet Transform* (DWT), *Stationary Wavelet Transform* (SWT), *Discrete Cosine Transform* (DCT), dan *Deep Neural Networks* (DNNs) dapat menghasilkan akurasi yang lebih baik?

1.3 Batasan Masalah

Mengingat permasalahan pengenalan wajah merupakan suatu hal yang cukup kompleks maka penelitian yang dilakukan dibatasi pada beberapa poin berikut:

1. Penelitian ini hanya diperuntukkan untuk pengenalan wajah.
2. Citra wajah yang digunakan pada penelitian ini adalah *AR Face Database*.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk menerapkan metode *Deep Neural Networks* (DNNs) untuk mengklasifikasi citra wajah menggunakan kombinasi *Discrete Wavelet Transform* (DWT), *Stationary Wavelet Transform* (SWT), dan *Discrete Cosine Transform* (DCT).

1.5 Manfaat Penelitian

Manfaat yang didapat dari penelitian ini yaitu untuk menentukan metode yang dapat digunakan untuk memperbaiki dan mengekstraksi fitur dari citra wajah. Metode ini dapat digunakan untuk bidang keamanan.

1.6 Kontribusi Penelitian

Kontribusi pada penelitian ini adalah untuk meningkatkan kualitas gambar/citra sebelum dilakukan ekstraksi fitur dan mempercepat proses pembelajaran menggunakan kompresi citra digital ke dalam bentuk frekuensi.

[Halaman ini sengaja dikosongkan]

BAB 2

KAJIAN PUSTAKA

Pada bagian ini akan dipaparkan konsep dasar tentang teori-teori yang dipakai sebagai pegangan dalam melakukan penelitian ini.

2.1 Klasifikasi Citra Wajah

Citra didefinisikan sebagai fungsi $f(x, y)$ dimana x dan y adalah sebuah koordinat bidang sedangkan f merupakan tingkat keabuan dari suatu citra berdasarkan koordinat. Jika nilai x , y , dan intensitas f tersebut bernilai diskrit, berhingga, citra tersebut dinamakan citra digital (Gonzalez, 2008). Nilai fungsi $f(x, y)$ pada koordinat citra (x, y) merupakan besar dari intensitas atau warna dari pixel pada titik itu (Sutoyo, 2009). Penulisan $f(x, y)$ bisa dilihat pada Persamaan 2.1

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0, M - 1) \\ f(1,0) & f(0,2) & \cdots & f(1, M - 1) \\ \vdots & \vdots & \ddots & \vdots \\ f(N - 1,0) & f(N - 1,1) & \cdots & f(N - 1, M - 1) \end{bmatrix} \quad (2.1)$$

Dimana, N merupakan jumlah baris dengan rentang $0 \leq k \leq N - 1$, sedangkan M merupakan jumlah kolom dengan rentang $0 \leq k \leq M - 1$.

Klasifikasi citra digital adalah pengelompokan citra ke dalam kelas berdasarkan jenis yang saling keterkaitan atau homogen. Secara umum klasifikasi digital biasa dibagi menjadi 2, yaitu klasifikasi tidak terbimbing (*Unsupervised Classification*) dan klasifikasi terbimbing (*Supervised Classification*) (Jaya, 2002). Klasifikasi tidak terbimbing adalah proses pengelompokan citra digital berdasarkan piksel menjadi beberapa kelas dengan cara analisa klaster. Sedangkan untuk klasifikasi terbimbing adalah pengelompokan piksel-pikasel berdasarkan klaster yang sudah ditetapkan. Sebelum melakukan klasifikasi maka dibutuhkan fungsi ekstraksi fitur. Dimana fungsi tersebut akan akan mendapatkan nilai piksel setiap citra digital.

Pengenalan wajah dilakukan dengan perbandingan citra wajah seseorang yang ingin dikenali dengan citra wajah yang sudah ada. Tahapan dalam proses pengenalan wajah dibagi menjadi dua langkah yaitu, ekstraksi fitur dan proses klasifikasi. Ekstraksi fitur adalah metode paling mendasar sebelum melakukan pengenalan gambar/citra. Ekstraksi fitur dibagi menjadi dua kategori yaitu, yang berdasarkan tata letak dan statistik (Wang, 2010). Ekstraksi fitur berdasarkan pada tata letak menggunakan jarak, sudut, dan letak komponen wajah seperti, mata hidung, mulut, dll. Sedangkan pada ekstraksi fitur berdasarkan statistik didapat dari hasil perhitungan algoritma.

2.2 Preprocessing

Citra wajah akan mengalami proses *preprocessing* sebelum dilakukan proses klasifikasi. Tahap pertama yang dilakukan adalah mengambil citra wajah, dimana pada tahap ini citra digital masih dalam format RGB (*Red, Green, Blue*) atau berwarna. Tahap selanjutnya adalah mengubah citra digital menjadi format *grayscale* atau format abu-abu dimana hanya terdapat nilai hitam dan putih. Pada tahap ini nilai *Red, Green, Blue* setiap piksel pada citra akan dijumlah dan dibagi menjadi 3, dimana hal tersebut hanya akan menghasilkan satu dimensi dengan rentang nilai dari piksel antara hitam atau putih. Penulisan rumus bisa dilihat pada Persamaan 2.2.

$$f(x, y) = \frac{f^R(x,y)+f^G(x,y)+f^B(x,y)}{3} \quad (2.2)$$

2.3 Histogram Equalization (HE)

Histogram Equalization (HE) adalah teknik paling dasar untuk menyamakan gambar dengan memetakan rentang tingkat intensitas yang sempit ke rentang tingkat intensitas yang lebih luas yang tersedia (Gonzalez, 2008). Metode *Histogram Equalization* dapat digunakan untuk meningkatkan kontras citra dan kualitas citra, sehingga informasi yang ada pada citra yang telah diperbaiki akan terlihat lebih jelas (Mau, 2016). Tujuan dari metode HE ini adalah untuk meratakan citra abu-abu menjadi seragam dengan cara mengubah derajat keabuan sebuah piksel (r) dengan derajat keabuan yang baru (s) dengan fungsi transformasi. Pada

penerapannya dapat ditulis secara matematis dengan persamaan $s = T(r)$ sedangkan nilai r didapat dari proses transformasi *inverse* dari s dengan persamaan $r = T^{-1}(s)$ dimana $0 \leq s \leq 1$. Persamaan yang digunakan untuk menghitung nilai HE dituliskan pada Persamaan 2.3 dan 2.4.

$$P_r(r_k) = \frac{n_k}{n} \quad (2.3)$$

$$r_k = \frac{k}{L-1}, 0 \leq k \leq L-1 \quad (2.4)$$

Dimana n_k adalah derajat keabuan pada piksel k , dan n adalah jumlah seluruh piksel pada citra. Pada persamaan tersebut dapat derajat keabuan (k) dibatasi dengan nilai dari derajat keabuan ($L-1$) hal tersebut dapat diartikan bahwa jika nilai $r_k = 0$, maka menyatakan nilai 0 begitu juga dengan sebaliknya jika nilai $r_k = 1$, maka pada piksel tersebut berwarna putih dalam skala keabuan.

2.4 Discrete Wavelet Transform (DWT)

Dasar dari Transformasi Wavelet Diskrit atau *Discrete Wavelet Transform* (DWT) dimulai pada tahun 1976 dimana teknik untuk mendekomposisi sinyal waktu diskrit ditemukan (Sripathi, 2003). DWT membagi sebuah dimensi sinyal menjadi dua bagian yakni frekuensi tinggi (*highpass filter*) dan frekuensi rendah (*lowpass filter*), yang disebut dengan dekomposisi (Terzija, 2006). Di dalam metode DWT untuk mendapatkan nilai sinyal dalam skala waktu digunakan teknik filterisasi. Dimana dalam prosesnya dua sinyal dengan skala frekuensi yang berbeda untuk di analisa pada filter. Dalam proses filterisasi DWT membagi sebuah dimensi sinyal menjadi dua bagian, yaitu frekuensi tinggi (*highpass filter*) dan frekuensi rendah (*lowpass filter*) (Gani, 2018).

Proses dekomposisi dimulai dengan melewati sinyal asal melewati *highpass filter* dan *lowpass filter*. Misalkan sinyal asal ini memiliki rentang frekuensi dari 0 sampai dengan $\pi \text{ rad/s}$. Dalam melewati *highpass filter* dan *lowpass filter*, rentang frekuensi pada *subsample* dibagi menjadi dua, sehingga rentang frekuensi tertinggi pada masing-masing *subsample* menjadi $\pi/2 \text{ rad/s}$. Setelah penyaringan, setengah dari *sample* atau salah satu *subsample* dapat dieliminasi

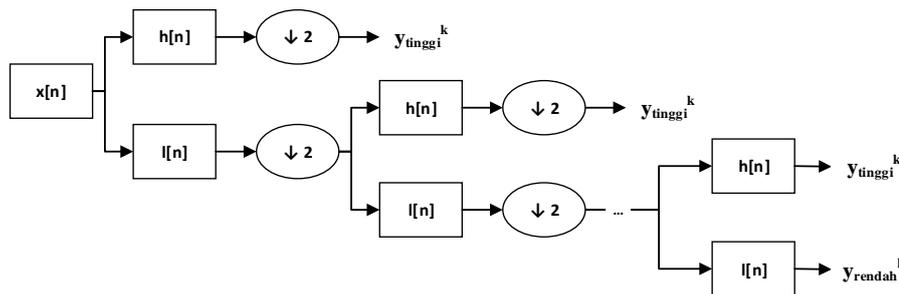
berdasarkan aturan Nyquist (Terzija, 2006). Sehingga sinyal dapat selalu di-*subsample* oleh 2 ($\downarrow 2$) dengan cara mengabaikan setiap *sample* yang kedua.

Proses dekomposisi ini dapat melalui satu atau lebih tingkatan. Dekomposisi satu tingkat ditulis dengan ekspresi matematika pada Persamaan 2.5 dan 2.6.

$$y_{tinggi}^{[k]} = \sum_n x[n] h[2k - n] \quad (2.5)$$

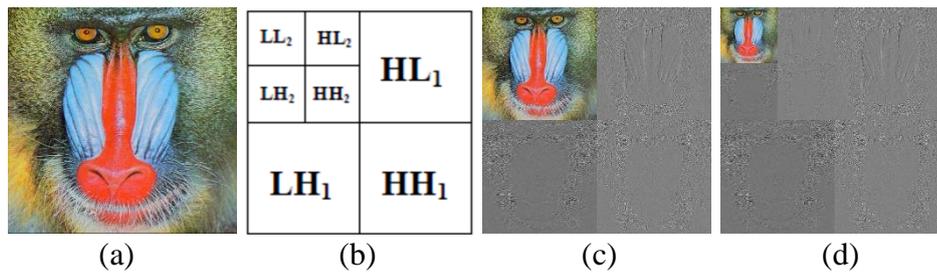
$$y_{rendah}^{[k]} = \sum_n x[n] l[2k - n] \quad (2.6)$$

$y_{tinggi}^{[k]}$ dan $y_{rendah}^{[k]}$ adalah hasil dari *highpass filter* dan *lowpass filter*, $x(n)$ merupakan sinyal asal, $h(n)$ adalah *highpass filter*, dan $l(n)$ adalah *lowpass filter*. Untuk dekomposisi lebih dari satu tingkat, maka prosedur pada Persamaan 2.5 dan 2.6 dapat digunakan pada masing-masing tingkatan. Contoh penggambaran dekomposisi dipaparkan pada Gambar 2.1 dengan menggunakan dekomposisi tiga tingkat.



Gambar 2.1 Dekomposisi Wavelet Tiga Tingkat

Pada Gambar 2.1, $y_{tinggi}^{[k]}$ dan $y_{rendah}^{[k]}$ yang merupakan hasil dari *highpass filter* dan *lowpass filter* disebut sebagai koefisien DWT. $y_{tinggi}^{[k]}$ merupakan detail informasi sinyal, sedangkan $y_{rendah}^{[k]}$ merupakan taksiran kasar dari fungsi pelaksanaan. $y_{tinggi}^{[k]}$ dapat mensaikan dua informasi rentang frekuensi HH (*high-high frequency*), HL (*high-low frequency*), dan LH (*low-high frequency*), sedangkan $y_{rendah}^{[k]}$ mensaikan dua informasi rentang frekuensi LL (*low-low frequency*).



Gambar 2.2 (a) Citra Baboon asli, (b) Struktur wavelet dua tingkat, (c) Dekomposisi Baboon menggunakan DWT, (d) Dekomposisi Baboon menggunakan DWT dua tingkat.

Filter yang digunakan pada DWT, antara lain *Haar* dan *Daubechies*. *Haar* merujuk pada suatu fungsi matematika (*Haar Wavelet*) yang berbentuk kotak, prinsipnya sama seperti pada fungsi *Fourier* (Sulistiyo, 2014). *Haar* memproses gambar dalam bentuk kotak-kotak, dimana dalam satu kotak terdapat beberapa piksel. Pada setiap kotak akan di proses dan menghasilkan perbedaan nilai (*threshold*) dari hasil nilai tersebut yang nantinya dijadikan dasar dalam *image processing*. Sedangkan *Daubechies* adalah pengembangan dari *wavelet Haar* (Suma'inna, 2013). *Daubechies 1* (db1) dengan panjang filter 2 merupakan *wavelet Haar*. *Daubechies 2* disingkat (db2) adalah *wavelet Daubechies* dengan banyak filter 4 dan seterusnya. Dari perbandingan filter *wavelet Haar* dan *wavelet Daubechies*, didapatkan hasil filter *wavelet Haar* memiliki nilai lebih tinggi dibandingkan dengan menggunakan filter *wavelet Daubechies* (Sari, 2019).

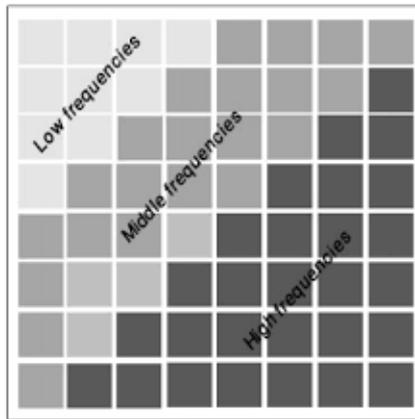
2.5 Stationary Wavelet Transform (SWT)

Stationary Wavelet Transform (SWT) memiliki cara kerja yang sama dengan DWT. Pada SWT metode *downsampling* yang terjadi pada metode DWT ditiadakan sehingga jumlah semua koefisiennya pada semua level dekomposisinya tidak berubah (Syahdeini, 2015). DWT dan SWT bekerja pada gelombang kecil yang memiliki frekuensi yang berbeda. Sama dengan DWT membagi sebuah dimensi sinyal menjadi dua bagian yakni frekuensi tinggi (*highpass filter*) dan frekuensi rendah (*lowpass filter*), frekuensi tinggi (*highpass filter*) menghasilkan frekuensi HL (*high-low frequency*), LH (*low-high frequency*), dan HH (*high-high frequency*) sedangkan frekuensi rendah (*lowpass filter*) menghasilkan frekuensi LL

(*low-low frequency*). Perbedaannya adalah citra pada *subband* LL menghasilkan citra yang sama dengan citra asli tidak ada *downsampling*. Maka dari itu metode SWT merupakan bentuk transformasi *wavelet* yang dirancang untuk mengatasi kekurangan pada metode *Discrete Wavelet Transform* (DWT) (Umar, 2019).

2.6 Discrete Cosine Transform (DCT)

Transformasi *Cosinus* Diskrit atau *Discrete Cosine Transform* (DCT) adalah sebuah teknik yang mengubah sinyal ke dalam komponen frekwensi dasar adalah algoritma yang digunakan untuk membuat kompresi (*lossy compression*) yaitu penempatan data dimana tidak ada satu *byte* pun data yang hilang sehingga data tersebut utuh dan disimpan sesuai dengan aslinya (Krasmla, 2017).



Gambar 2.3 Distribusi Frekuensi Pada Blok DCT (Faruqi, 2015)

Pada Gambar 2.3 distribusi frekuensi pada citra 8x8 blok dari domain spasial ke domain DCT. DCT dibagi menjadi 2, yaitu DCT – 1 Dimensi dan DCT-2 dimensi. DCT -1 dimensi adalah sederet bilangan real $d(x)$, dimana $x = 1, 2, \dots, n-1$, dapat dirumuskan pada Persamaan 2.7 (Watson, 1994).

$$d(x) = \left(\frac{2}{N}\right)^{\frac{1}{2}} \sum_{x=0}^{n-1} d(u) C(u) \cos\left(\frac{(2x+1)u}{2n}\right) \quad (2.7)$$

dimana $u = 0, 1, 2, 3, \dots, n-1$, sedangkan $C(u) = 2 - \frac{1}{2}$ untuk $u = 0$

Sedangkan DCT-2 dimensi dapat diperoleh dengan rumus berikut, dapat dirumuskan pada Persamaan 2.8 (Faruqi, 2015).

$$C(u, v) = \frac{2}{\sqrt{MN}} \alpha(u) \alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{M-1}$$

$$f(x, y) \cos\left(\frac{\pi(2x+1)u}{2N}\right) \cos\left(\frac{\pi(2x+1)u}{2M}\right) \quad (2.8)$$

dimana C adalah koefisien pada indeks ke- u . M merupakan ukuran tinggi matriks, sedangkan N merupakan ukuran lebar matriks. Sedangkan x adalah indeks baris dan y adalah indeks kolom dari citra 2 dimensi yang dicari nilainya.

2.7 Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) atau Jaringan Saraf Tiruan (JST) adalah model algoritma yang terinspirasi dari cara kerja neuron pada otak manusia. Dimana tiap neuron pada otak manusia saling terhubung dan informasi mengalir ke setiap neuron. Pada JST, neuron-neuron akan dikumpulkan dalam lapisan-lapisan yang disebut dengan lapisan (*layer*) neuron. Neuron pada prinsipnya dihubungkan pada lapisan sebelum atau sesudahnya, akan tetapi tidak bisa terhubung dengan masuk dan keluar.

JST memiliki 3 (tiga) komponen yang berperan penting (Sivanandam, 2009), yaitu arsitektur jaringan, algoritma pembelajaran yang berfungsi untuk pembobotan pada jaringan yang terbentuk antara neuron, dan fungsi aktivasi yang digunakan. Pada pemodelan JST sendiri juga dibagi menjadi 3 (tiga) bagian, yaitu *Synapsis* (Jalur Penghubung) antara neuron, *Summing Unit* untuk melakukan perhitungan terhadap total *output*, dan aktivasi dalam pembatasan nilai *output*.

Dalam proses pembuatan model JST diperlukan arsitektur yang digunakan untuk pengelompokan lapisan-lapisan (*layers*). Arsitektur JST dibagi menjadi 3 (Lesnussa, 2017), yaitu Jaringan Saraf Dengan Lapisan Tunggal (*Single Layer Net*), Jaringan Saraf Dengan Banyak Lapisan (*Multi Layer Net*), Jaringan Saraf Dengan Lapisan Kompetitif (*Competitive Layer*). Dimana Jaringan Saraf Dengan Lapisan Tunggal hanya terdiri dari satu lapisan dengan bobot yang terhubung, Jaringan Saraf Dengan Banyak Lapisan terdiri dari beberapa lapisan dengan bobot yang terhubung dengan kelebihan dapat mengatasi masalah yang lebih kompleks, sedangkan Jaringan Saraf Dengan Lapisan Kompetitif merupakan antara neuron akan saling terhubung atau bisa disebut *feedback loop* dikarenakan nilai *output* (keluaran) akan memberikan informasi terhadap tiap *input* (masukan).

Fungsi aktivasi bertujuan mengubah nilai *input* menjadi satu *output* yang telah bertransformasi. Dimana pada prosesnya akan melalui proses fungsi pembatasan. Fungsi aktivasi yang sering digunakan adalah fungsi linier, sigmoid, fungsi Tanh, dan *Rectified Linear Unit* (ReLU). Fungsi linear atau sering disebut fungsi identitas adalah fungsi yang secara *default* (standar) dalam fungsi aktivasi nilai *output* akan sama dengan nilai *input*. Fungsi sigmoid atau sering disebut fungsi berbentuk kurva *s* adalah fungsi aktivasi yang berguna karena penggunaan hubungan sederhana antar nilai fungsi pada sebuah poin dengan nilai *derivatif*, yang dapat mengurangi hambatan komputasi selama percobaan, persamaan fungsi sigmoid dapat dirumuskan pada Persamaan 2.9.

$$s(x) = \frac{1}{1+e^{-x}} \quad (2.9)$$

dimana pada sigmoid akan menerima angka tunggal dan mengubah nilai x menjadi sebuah nilai dengan rentang mulai dari 0 – 1. Fungsi Tanh adalah sebuah fungsi yang memiliki rentang nilai dari -1 sampai 1. Sama dengan fungsi Sigmoid, Tanh yaitu dapat menghapus nilai gradien, persamaan fungsi Tanh dapat dirumuskan pada Persamaan 2.10.

$$\text{Tanh}(x) = 2\sigma(2x) - 1 \quad (2.10)$$

Rectified Linear Unit (ReLU) adalah fungsi yang akhir-akhir ini sering digunakan karena membuat pembatas pada bilangan 0, persamaan fungsi ReLU dapat dirumuskan pada Persamaan 2.11.

$$f(x) = \max(0, x) \quad (2.11)$$

dimana apabila nilai $x \leq 0$ maka nilai $x = 0$, sedangkan apabila $x \geq 0$ maka nilai $x = x$. Pada aktivasi ReLU ini memiliki kelebihan yaitu, mempercepat proses konvergensi jika dibandingkan dengan Sigmoid atau Tanh dan ReLU bisa diimplementasikan hanya dengan membuat batasan (*threshold*) pada bilangan 0. Akan tetapi ReLU memiliki kekurangan yaitu jika pada proses pelatihan akan dapat mematikan unit.

Fungsi pembobotan adalah pemilihan bobot dan bias yang berfungsi mencari nilai eror minimum global atau eror minimum lokal. Pada pembobotan ini

sangat penting dalam menentukan nilai bias yang mengakibatkan turunnya fungsi aktivasi. Metode *Backpropagation* adalah sebuah metode statistik JST yang menggunakan algoritma pembelajaran terawasi (*Supervised Learning*) dimana digunakan untuk mengubah bobot pada lapisan tersembunyi (*hidden layers*). Ada 3 langkah untuk menjalankan *Backpropagation* yaitu, langkah maju (*feedforward*) dari masukan, langkah mundur (*backpropagation*) dari eror, kemudian dilakukan penyesuaian bobot.

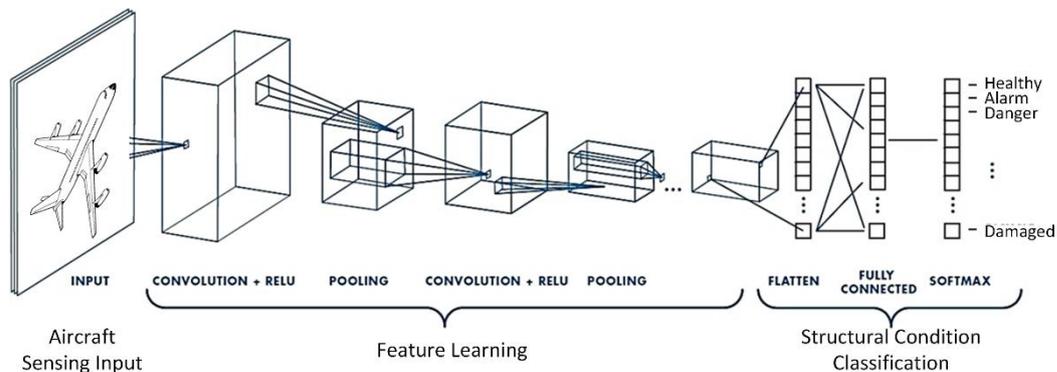
Arsitektur dari *Backpropagation* dimana terdapat neuron yang terhubung dengan lapisan tersembunyi. Neuron tersebut terhubung pada lapisan *input* (*input layer*) dengan lapisan tersembunyi, kemudian neuron pada lapisan tersembunyi terhubung pada lapisan *output* (*output layer*). Dimana *input layer* merupakan 1 lapis neuron (*input neuron*) mulai dari *input* pertama sampai dengan *input* ke-*n*. *Hidden layer* terdiri dari beberapa neuron tersembunyi awal sampai ke-*n*. *Output Layer* adalah akhir dari alur neuron dimana terdiri dari nilai *output* neuron dari *output* awal sampai *output* ke-*n*.

Deep Neural Networks (DNNs) adalah *Neural Networks* (NN) dengan banyak lapisan (*layer*) atau bisa disebut dengan *Multilayer Perceptron* (MPL). DNNs merupakan salah satu algoritma berbasis jaringan saraf yang dapat digunakan untuk pengambilan keputusan. DNNs adalah teknik dalam *neural network* yang menggunakan teknik tertentu seperti *Restricted Boltzmann Machine* (RBM) untuk mempercepat proses pembelajaran dalam *Neural Network* yang menggunakan banyak lapisan (Tanaka, 2014). Lapisan pada DNNs terdiri atas tiga bagian yaitu *input layer*, *hidden layer* dan *output layer*.

Pada *hidden layer* dapat dibuat berlapis, hal tersebut dapat mengurangi nilai eror pada hasil *output*. Maka dari itu penambahan lebih banyak lapisan menjadikan model yang dihasilkan semakin mewakili citra tersebut. Dengan semakin bertambah banyaknya lapisan yang digunakan akan mempengaruhi besar data, kecepatan, dan akurasi. Pada prinsipnya semakin metode DNNs semakin sering digunakan untuk melakukan penelitian untuk menyelesaikan masalah tentang data besar.

Dengan berkembangnya waktu hal tersebut dapat memicu berkembangnya DNNs dimana dalam satu model dapat berdiri dari beberapa *layer* untuk melakukan transformasi data sebelum data diolah menggunakan metode klasifikasi. Maka dari itu jumlah layer pada DNNs tidak memiliki aturan universal dan berlaku, hal tersebut tergantung dengan *dataset* yang digunakan. Karena hal tersebut, jumlah layer pada jaringan serta jumlah neuron pada masing-masing *layer* dianggap sebagai *hyperparameter* dan di optimasi menggunakan pendekatan *searching*.

Metode *Convolutional Neural Network* (CNN) adalah pengembangan dari model DNNs yang memiliki arsitektur yang sudah ditetapkan. Arsitektur dari CNN dibagi menjadi 2 bagian besar, *Feature Extraction Layer* dan *Fully-Connected Layer* (MLP). Penerapannya bisa dilihat pada Gambar 2.4



Gambar 2.4 Arsitektur CNN (Tabian, 2019)

Feature Learning adalah proses yang dilakukan untuk mengubah citra menjadi bentuk angka yang merepresentasikan citra tersebut atau disebut *Feature Extraction*. *Feature extraction* sendiri terdiri dari dua bagian yaitu *Convolutional Layer* dan *Pooling Layer*. *Convolutional Layer* bertujuan menghitung nilai output dari neuron yang terhubung ke daerah lokal dalam *input*, masing-masing akan menghitung bobot antara wilayah kecil yang terhubung ke dalam volume *input*. Sedangkan *pooling layer* adalah lapisan yang mengurangi dimensi dari *feature map* atau lebih dikenal dengan langkan untuk *downsampling*, sehingga dapat mempersingkat komputasi parameter yang harus di *update* semakin sedikit dan mengatasi *overfitting*.

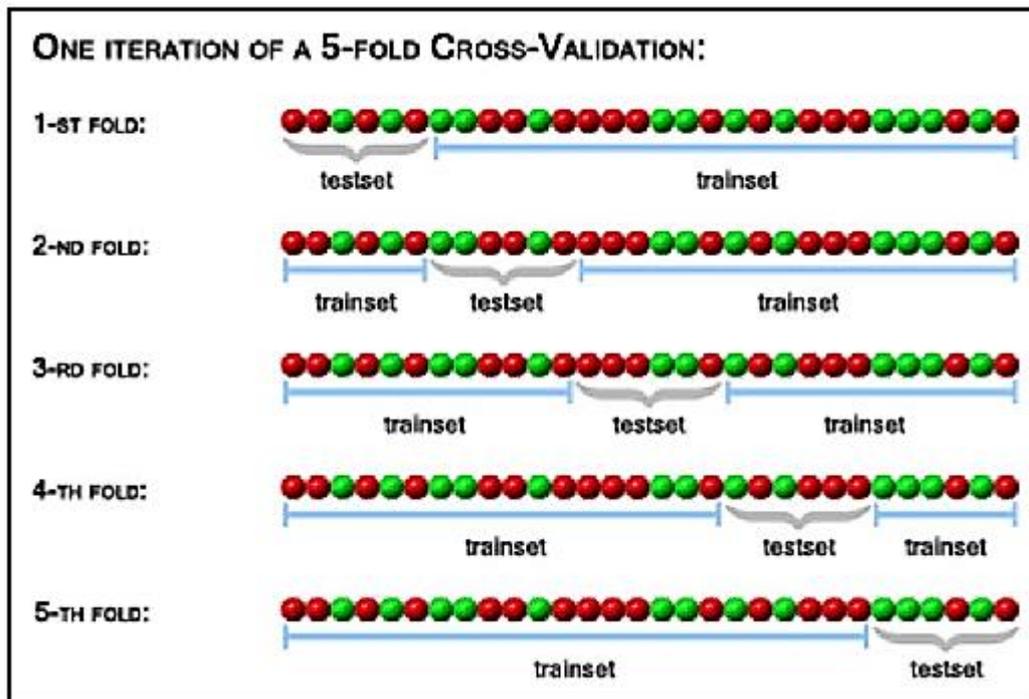
Classification Lapisan ini berguna untuk mengklasifikasikan tiap neuron yang telah diekstraksi fitur pada sebelumnya. Terdiri dari *Flatten*, *Fully-connected*,

dan *Softmax*. Dimana *Flatten* bertujuan membentuk ulang fitur (*reshape feature map*) menjadi sebuah *vector* agar bisa kita gunakan sebagai *input* dari *fully-connected layer*. Sedangkan *fully connected layer* (yaitu terhubung sepenuhnya) bertujuan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasi secara linear. Seperti Jaringan Saraf biasa dimana setiap neuron dalam lapisan ini akan terhubung ke neuron. Tiap neuron pada *convolution layer* perlu ditransformasi menjadi data satu dimensi terlebih dahulu sebelum dapat dimasukkan ke dalam sebuah *fully connected layer*. Fungsi *Softmax* ditujukan untuk menghitung nilai probabilitas dari setiap kelas target atas semua kelas target yang memungkinkan dan akan membantu untuk menentukan kelas target untuk *input* yang diberikan. Kelebihan utama dari penggunaan *softmax* adalah memiliki rentang antara 0 hingga 1. Sedangkan untuk hasil *output* dari fungsi *softmax* adalah rasio eksponensial dari nilai *input* dan jumlah nilai eksponensial adalah *output*.

Metode optimasi pada yang terdapat pada *Deep Learning* antara lain adalah *Adam*, *Adam* adalah algoritma yang digunakan untuk mengoptimalkan *gradient* pada *neural network*. Metode *Adam* lebih mudah diimplementasikan, efisien secara komputasi, memerlukan kebutuhan memori yang kecil, dan sesuai untuk masalah yang memiliki banyak data maupun parameter (Kingma, 2015).

2.8 K-fold Cross-Validation

Cross Validation merupakan metode yang digunakan untuk evaluasi hasil kinerja. *Cross Validation* bisa disebut juga dengan *K-fold Cross Validation* dimana langkah pertama yang ditentukan adalah nilai *K* dimana nilai *K* merupakan jumlah iterasi yang akan dilakukan untuk proses *training* dan *testing*. Dimana untuk *fold* ke-*x* maka jumlah data total dibagi sebanyak *K* kemudian nilai tersebut akan menjadi jumlah data *test* (*m*) sedangkan jumlah data *training* adalah jumlah data (*n*) dikurangi jumlah data *testing* (*m*), maka jumlah data *training* adalah (*n-m*). dan proses tersebut diulang sebanyak *K* kali dengan data testing indeks ke *K*. Contoh *Cross Validation* bisa dilihat di Gambar 2.5 dimana 30 data dilakukan *K-fold Cross Validation*



Gambar 2.5 K-fold Cross Validarion (Chemchem, 2019)

2.9 Metode Evaluasi

Metode evaluasi yang digunakan pada penelitian ini adalah akurasi. Dimana jumlah hasil prediksi yang benar dibagi dengan jumlah data uji, bisa dilihat pada Persamaan 2.12.

$$Akurasi = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ data\ uji} \times 100\% \quad (2.12)$$

Sedangkan untuk mengetahui seberapa lama waktu yang digunakan dalam melakukan proses klasifikasi pada setiap data yang digunakan melalui waktu proses berjalan dikurangi waktu ketika proses berhenti dan dibagi 60 untuk menunjukkan dalam menit, bisa dilihat pada Persamaan 2.13.

$$Waktu = \frac{(Waktu\ mulai - Waktu\ berakhir)}{60} \text{ Menit} \quad (2.13)$$

2.10 Keras

Keras merupakan *interface library* yang dibangun untuk mensederhanakan implementasi algoritma-algoritma *Deep Learning* di atas *TensorFlow*. *TensorFlow*

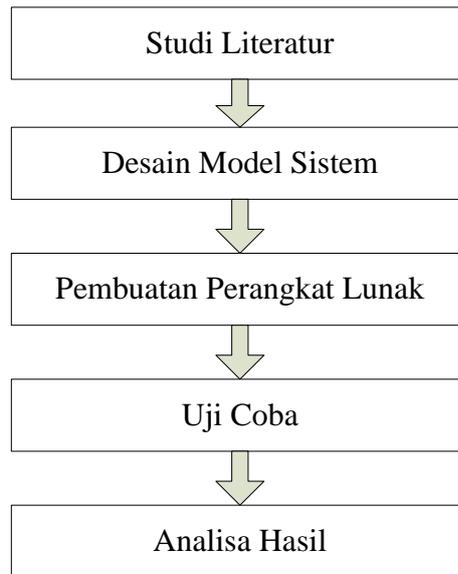
merupakan *computational framework* untuk membangun model *machine learning* seperti *neural networks*. *Tensorflow* dikembangkan oleh tim Google Brain untuk penggunaan Google internal dan dirilis di bawah lisensi *open source* Apache 2.0. *Tensorflow* adalah koleksi *software open source* untuk komputasi numerik yang menggunakan grafik aliran data (Toleubay, 2020).

[Halaman ini sengaja dikosongkan]

BAB 3

METODE PENELITIAN

Adapun tahapan penelitian yang akan dilakukan pada penelitian ini dapat digambarkan sebagai berikut.



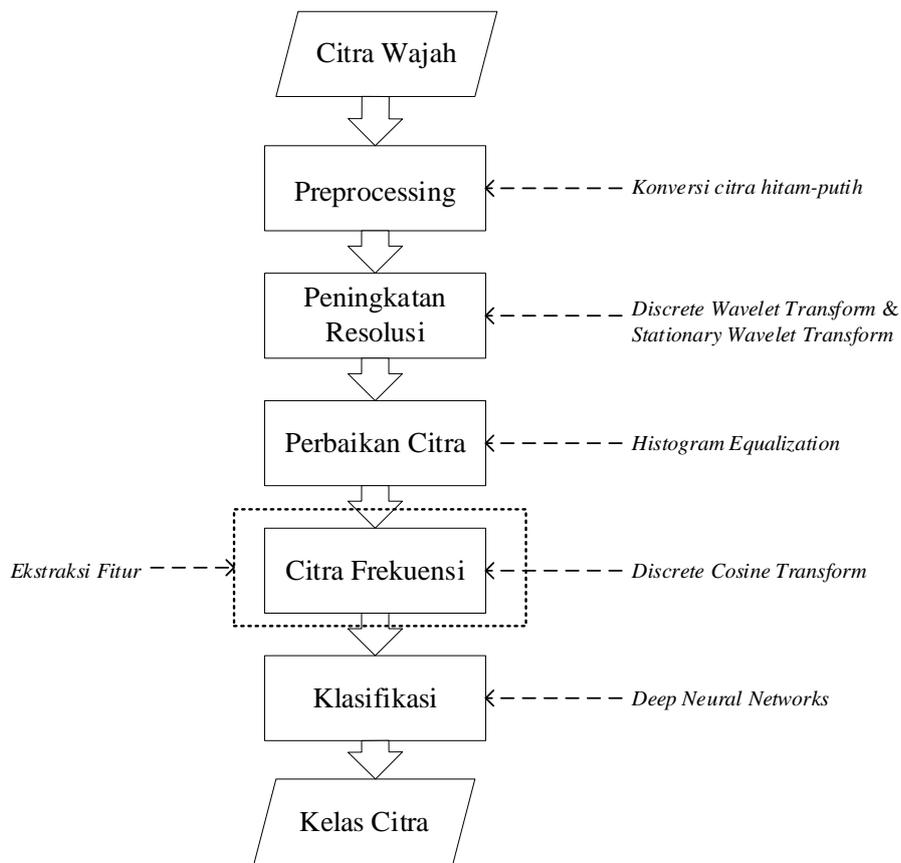
Gambar 3.1 Tahapan Penelitian

3.1 Studi Literatur dan Analisa Awal

Dalam melakukan suatu penelitian, tahapan studi literatur dan analisa awal ini merupakan hal yang mutlak diperlukan. Hal ini berkaitan dengan suatu pemahaman detail baik dari sisi konsep/dasar teori yang dipakai maupun teknis dari tiap tahapan suatu penelitian. Pada tahap ini akan dipelajari tentang segala informasi dan sumber pustaka yang disesuaikan dengan konteks penelitian yang dilakukan. Dalam penelitian ini, literatur yang dikaji secara garis besar meliputi konsep-konsep dasar yang terkait dengan ekstraksi fitur dan klasifikasi citra digital.

3.2 Desain Model Sistem

Sistem yang akan dibangun adalah sistem pengenalan citra wajah. Alur kerja dari sistem ini terdiri dari 5 tahapan seperti yang dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram Alir Sistem Usulan

3.2.1 Data

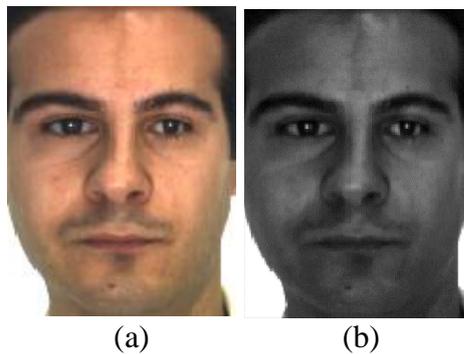
Data yang digunakan pada penelitian ini adalah *AR Face Database* dimana data tersebut memiliki total 100 orang yang berbeda. Dimana terdapat 26 citra yang berbeda untuk setiap orang, dengan total citra wajah sebanyak 2600. Pada *AR Face Database* memiliki jenis citra yang berbeda, misalnya: intensitas cahaya, ekspresi wajah, dan ada aksesoris yang digunakan dapat dilihat pada Gambar 3.3 (Martinez, 1998)(Martinez, 2001).



Gambar 3.3 Citra *Dataset* (a) Wajah Normal, (b) Bereksresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

3.2.2 Preprocessing

Citra wajah akan mengalami proses *preprocessing* sebelum dilakukan proses klasifikasi. Tahap pertama yang dilakukan adalah mengambil citra wajah, dimana pada tahap ini citra digital masih dalam format RGB (*Red, Green, Blue*) atau berwarna. Tahap selanjutnya adalah mengubah citra digital menjadi format *grayscale* atau format abu-abu dimana hanya terdapat nilai hitam dan putih. Perbedaan hasil dari gambar setelah dan sebelum dilakukan *preprocessing* dapat dilihat pada Gambar 3.4.

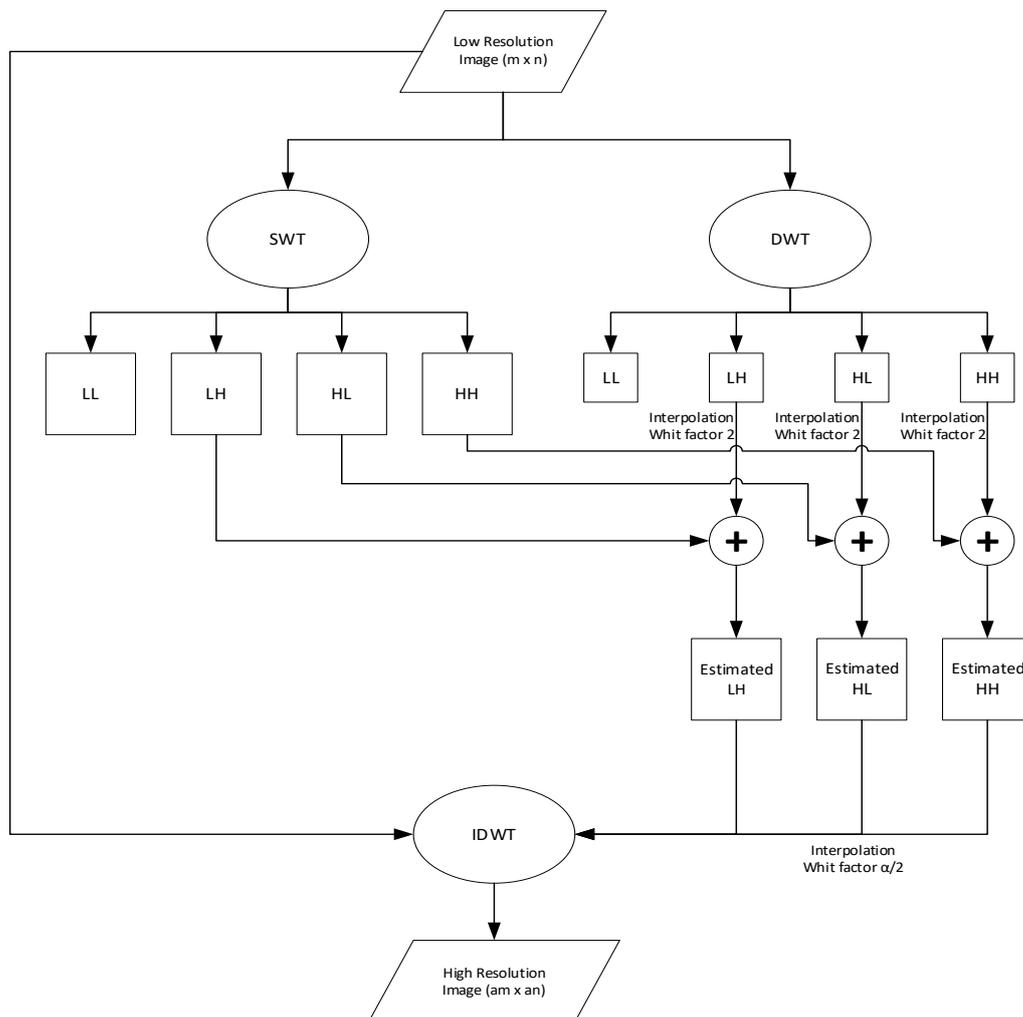


Gambar 3.4 (a) Citra Normal, (b) Citra Hasil *Grayscale*

3.2.3 Peningkatan Resolusi

Peningkatan resolusi pada citra wajah adalah dengan menggunakan gabungan metode DWT dengan SWT. Kelemahan dari metode DWT adalah adanya *downsampling* pada sebuah citra. *Downsampling* pada masing-masing sub-DWT menyebabkan hilangnya informasi pada masing-masing *sub-sample*. Sedangkan pada metode SWT *downsampling* yang terjadi pada DWT ditiadakan sehingga jumlah semua koefisiennya pada semua level dekomposisinya tidak berubah. Akan tetapi untuk menggabungkan hasil dari DWT dengan SWT diperlukan metode interpolasi dikarenakan dimensi hasil dari SWT dan DWT berbeda. Langkah pertama adalah melakukan interpolasi 2 pada frekuensi tinggi (*highpass filter*), yaitu frekuensi HL (*high-low frequency*), LH (*low-high frequency*), dan HH (*high-high frequency*). Kelemahan dari tahap interpolasi adalah garis tepi pada citra akan memudar. Oleh karenanya diperlukan penambahan frekuensi tinggi (*highpass filter*) pada SWT akan memperbaiki memudarnya garis tepi pada hasil interpolasi 2. Setelah melakukan penambahan maka akan dilakukan interpolasi $\alpha/2$, dimana α adalah berapa kali dilakukan interpolasi pada tahap sebelumnya. Sedangkan untuk

frekuensi rendah (*lowpass filter*) tidak digunakan, tetapi digunakan citra asli dan kemudian dilakukan interpolasi $\alpha/2$. Selanjutnya dilakukan *Inverse DWT* atau mengembalikan metode DWT ke dalam bentuk citra awal. Dengan frekuensi rendah (*lowpass filter*) dari citra asli yang di interpolasi $\alpha/2$, dan pada frekuensi tinggi (*highpass filter*) adalah hasil penambahan HL (*high-low frequency*), LH (*low-high frequency*), dan HH (*high-high frequency*) dari DWT dan SWT kemudian dilakukan interpolasi $\alpha/2$. Alur dalam peningkatan resolusi citra bisa dilihat pada Gambar 3.5.



Gambar 3.5 Alur Perbaikan Citra (Demirel, 2011)

Dimana *Low Resolution Image* adalah citra *grayscale*. Sedangkan untuk filter yang digunakan pada DWT dan SWT adalah *wavlet Haar*. *Pseudocode* alur kerja pada proses perbaikan citra menggunakan gabungan metode DWT dengan SWT dapat dilihat pada Tabel 3.1.

Tabel 3.1 *Pseudocode* gabungan DWT dan SWT

Input	train_data, test_data
Output	train_data, test_data
1	# DWT
2	TrainDWT = dwt(train_data, 'haar')
3	TrainLL, (TrainLH, TrainHL, TrainHH) = TrainDWT
4	TestDWT = dwt(test_data, 'haar')
5	TestLL, (TestLH, TestHL, TestHH) = TestDWT
6	
7	# SWT
8	TrainSWT = swt(train_data, 'haar', 1)
9	sTrainLL, (sTrainLH, sTrainHL, sTrainHH) =
10	TrainSWT
11	TestSWT = swt(test_data, 'haar', 1)
12	sTestLL, (sTestLH, sTestHL, sTestHH) = TestSWT
13	
14	# Interpolasi 2
15	Inter = 2
16	rTrainLL = resize(TrainLL, axisX = Inter, axisY =
17	Inter, interpolation = INTER_CUBIC)
18	rTrainLH = resize(TrainLH, axisX = Inter, axisY =
19	Inter, interpolation = INTER_CUBIC)
20	rTrainHL = resize(TrainHL, axisX = Inter, axisY =
21	Inter, interpolation = INTER_CUBIC)
22	rTrainHH = resize(TrainHH, axisX = Inter, axisY =
23	Inter, interpolation = INTER_CUBIC)
24	
25	rTestLL = resize(TestLL, axisX = Inter, axisY =
26	Inter, interpolation = INTER_CUBIC)
27	rTestLH = resize(TestLH, axisX = Inter, axisY =
28	Inter, interpolation = INTER_CUBIC)
29	rTestHL = resize(TestHL, axisX = Inter, axisY =
30	Inter, interpolation = INTER_CUBIC)
31	rTestHH = resize(TestHH, axisX = Inter, axisY =
32	Inter, interpolation = INTER_CUBIC)
33	
34	# Penambahan
35	rsTrainLH = sTrainLH + rTrainLH
36	rsTrainHL = sTrainHL + rTrainHL
37	rsTrainHH = sTrainHH + rTrainHH
38	
39	rsTestLH = sTestLH + rTestLH
40	rsTestHL = sTestHL + rTestHL
41	rsTestHH = sTestHH + rTestHH
42	

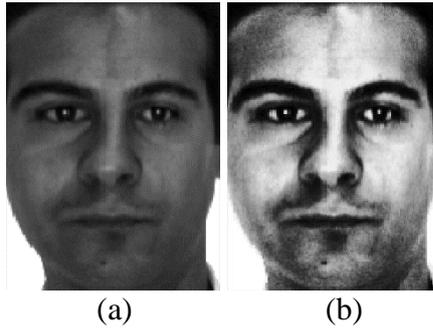
```

43 # Interpolasi a/2
44 a = 1
45 Inter2 = a/2
46 TrainImg = resize(train_data, axisX = Inter2,
47 axisY = Inter2, interpolation = INTER_CUBIC)
48 iTrainLH = resize(rsTrainLH, axisX = Inter2,
49 axisY = Inter2, interpolation = INTER_CUBIC)
50 iTrainHL = resize(rsTrainHL, axisX = Inter2,
51 axisY = Inter2, interpolation = INTER_CUBIC)
52 iTrainHH = resize(rsTrainHH, axisX = Inter2,
53 axisY = Inter2, interpolation = INTER_CUBIC)
54
55 TestImg = resize(Test_data, axisX = Inter2, axisY
56 = Inter2, interpolation = INTER_CUBIC)
57 iTestLH = resize(rsTestLH, axisX = Inter2, axisY
58 = Inter2, interpolation = INTER_CUBIC)
59 iTestHL = resize(rsTestHL, axisX = Inter2, axisY
60 = Inter2, interpolation = INTER_CUBIC)
61 iTestHH = resize(rsTestHH, axisX = Inter2, axisY
62 = Inter2, interpolation = INTER_CUBIC)
63
64 # Inverse DWT
65 TrainIDWT = TrainImg, (iTrainLH, iTrainHL,
66 iTrainHH)
67 train_data = idwt(TrainIDWT, 'haar')
68 TestIDWT = TestImg, (iTestLH, iTestHL, iTestHH)
69 test_data = idwt(TestIDWT, 'haar')
70
71 return train_data, test_data

```

3.2.4 Perbaikan Citra

Setelah melakukan Peningkatan resolusi pada citra wajah maka selanjutnya dilakukan perbaikan kontras citra dan kualitas citra dengan menggunakan metode *Histogram Equalization* (HE). Proses ini digunakan untuk memperbaiki citra wajah dengan intensitas cahaya yang kurang dengan meratakan derajat keabuan citra tersebut. Perbedaan hasil dari gambar setelah dan sebelum dilakukan HE dapat dilihat pada Gambar 3.6.



Gambar 3.6 (a) Citra *Grayscale*, (b) Citra Hasil HE

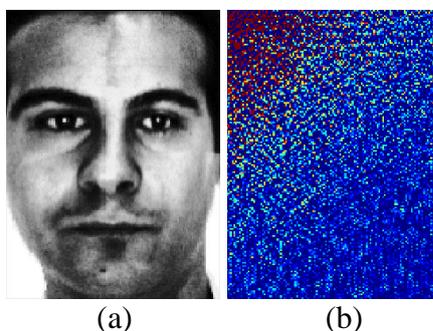
Hal tersebut dilakukan dengan meratakan nilai piksel dengan rentang tingkat intensitas yang sempit ke rentang tingkat intensitas yang lebih luas. *Pseudocode* alur kerja pada proses ekstraksi fitur menggunakan metode HE dapat dilihat pada Tabel 3.2.

Tabel 3.2 *Pseudocode* HE

Input	train_data, test_data
Output	train_data, test_data
1	x = train_data
2	y = test_data
3	
4	EH_dataTrain = equalizeHist(x)
5	EH_dataTest = equalizeHist(y)
6	
7	return EH_dataTrain, EH_dataTest

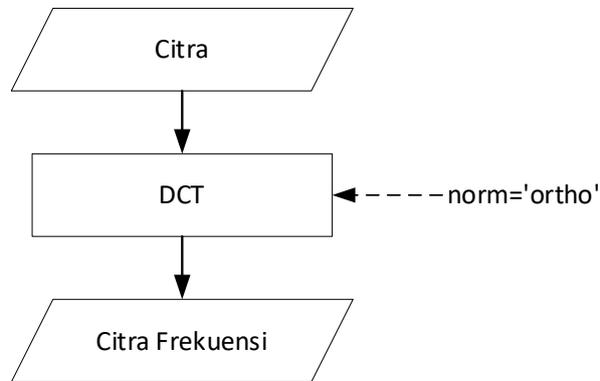
3.2.5 Transformasi Citra

Pada tahapan ini citra hasil dari *Histogram Equalization* (HE) akan di rubah menjadi bentuk frekuensi 2 dimensi menggunakan metode DCT. Perbedaan hasil dari gambar setelah dan sebelum menjadi bentuk frekuensi dapat dilihat pada Gambar 3.7.



Gambar 3.7 (a) Citra Hasil HE, (b) Citra Hasil DCT

Sedangkan untuk alur proses pada tahap transformasi citra, dimana citra wajah di ubah menjadi citra frekuensi menggunakan metode DCT bisa dilihat pada Gambar 3.8.



Gambar 3.8 Alur Transformasi Citra

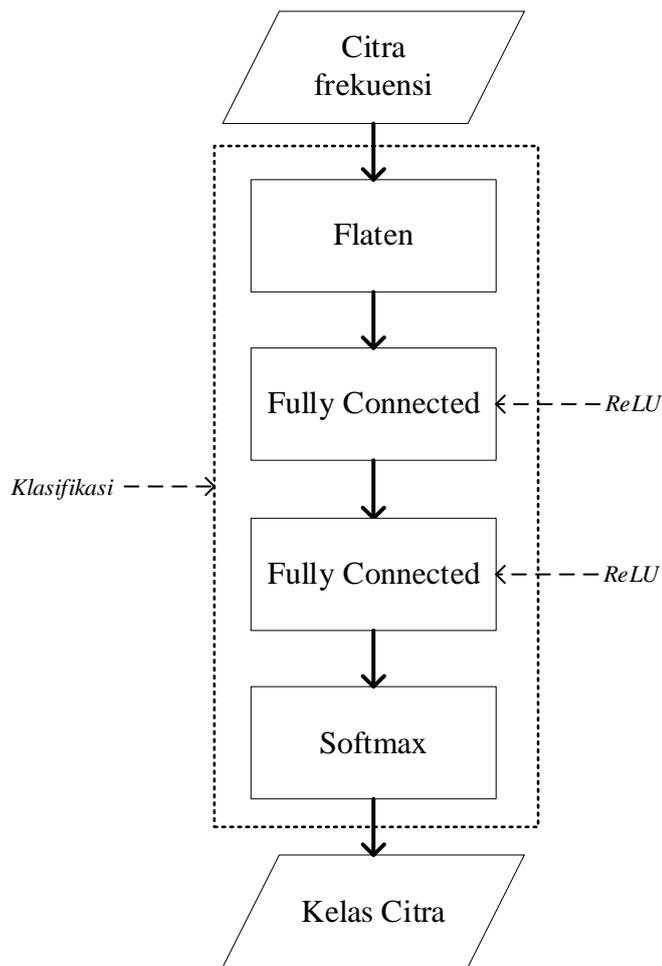
Pseudocode alur kerja pada proses perubahan citra menjadi bentuk frekuensi menggunakan metode DCT dapat dilihat pada Tabel 3.3.

Tabel 3.3 *Pseudocode* DCT

Input	train_data, test_data
Output	train_data, test_data
1	x = train_data
2	y = test_data
3	
4	def dct2(a):
5	return dct(dct(a.T, norm='ortho').T,
6	norm='ortho')
7	
8	imageDctTrain = dct2(x)
9	imageDctTest = dct2(y)
10	
11	return imageDctTrain, imageDctTest

3.2.6 Klasifikasi

Pada tahap klasifikasi metode yang digunakan adalah *Deep Neural Networks* (DNNs), dimana tahap - tahapan yang di lakukan adalah *flatten*, *fully connected*, dan *softmax*.



Gambar 3.9 Arsitektur *Deep Neural Networks* (DNNs)

Pada Gambar 3.9 dapat dilihat bahwa aktivasi yang digunakan adalah ReLU. Di mana jumlah *fully connected* yang digunakan bisa lebih dari satu dan jumlah neuron yang terhubung setiap *layer* bisa berbeda. *Pseudocode* alur kerja pada proses DNNs dapat dilihat pada Tabel 3.4, dengan arsitektur 2 *fully connected* yang terdiri dari 256 dan 128 neuron, dan menggunakan metode optimasi *Adam*.

Tabel 3.4 *Pseudocode* DNNs

Input	train_data, test_data, epochs
Output	akurasi
1	x = train_data
2	y = test_data
3	
4	train_x, test_x = x(i, j)
5	rain_y, test_y = y(i, j)
6	
7	img width, img height = train_x.shape

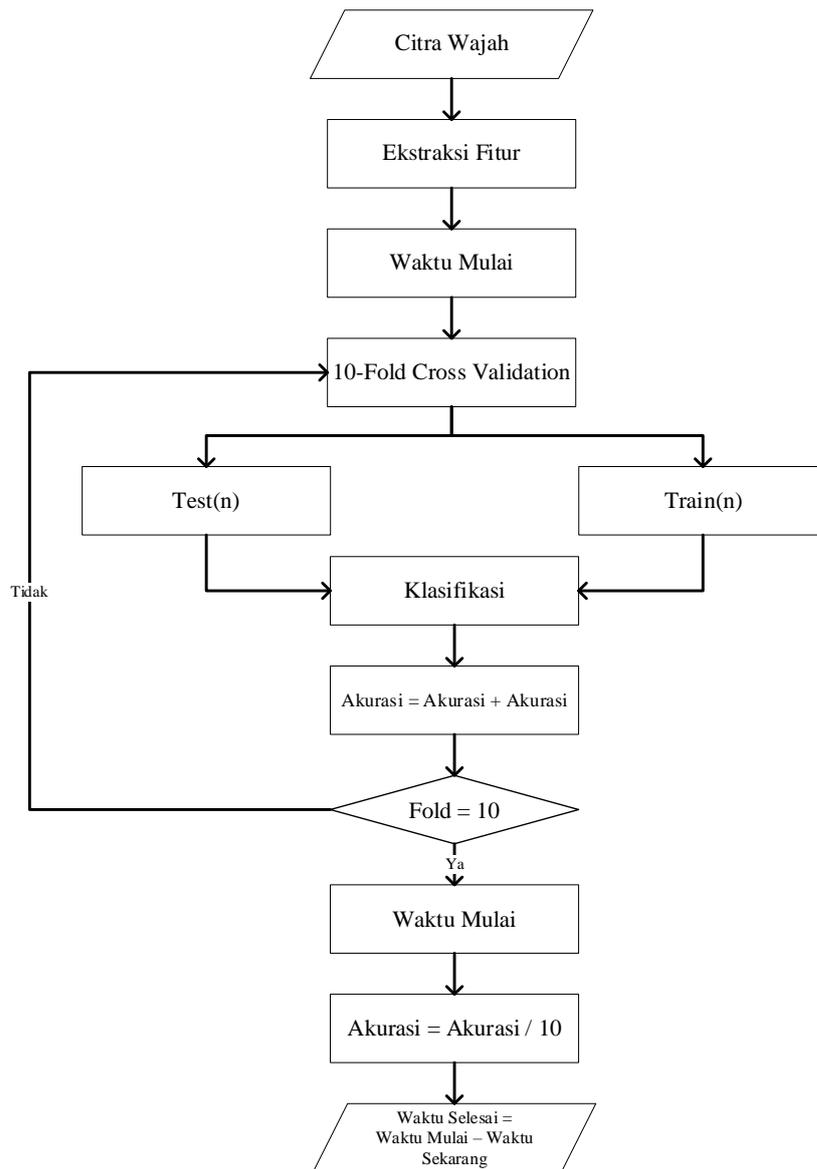
```
8
9  model = Sequential([
10     Flatten(input_shape=(img_width, img_height)),
11     Dense(256, activation='relu'),
12     Dense(128, activation='relu'),
13     Dense(num_classes, activation='softmax')
14 ])
15
16 model.compile(optimizer='adam')
17
18 model.fit(train_x, train_y, epochs=epochs)
19 test_acc = model.evaluate(test_x, test_y)
20
21 return test_acc
```

3.3 Implementasi

Dalam tahap ini, diimplementasikan rancangan yang telah dibuat pada proses sebelumnya. Sistem akan dikembangkan untuk lingkungan desktop dengan menggunakan bahasa pemrograman python. Untuk *preprocessing*, *library tensorflow*, dan meletakkan semua *dataset* citra wajah dalam folder yang sama.

3.4 Skenario Uji Coba

Untuk menguji keakuratan dari produk yang dihasilkan sistem, maka dilakukan pengujian *K-Fold Cross Validation* sebanyak 10 *fold*, di mana pada setiap *fold* akan dilakukan perulangan sebanyak *epoch* pada saat proses klasifikasi. Sedangkan untuk menentukan jumlah *epoch* yang optimal maka akan di uji dengan jumlah *epoch* sebanyak 5, 10, 15, 20, 25, 30, 35, 40, 45, dan 50 kali dengan menggunakan data uji *fold* ke-1.



Gambar 3.10 Alur proses K-Fold Cross Validation

Pada Gambar 3.10 dapat dilihat alur proses pengujian metode ekstraksi fitur menggunakan 10-Fold Cross Validation. Pada tahap ekstraksi fitur akan dilakukan uji coba pada setiap metode yang digunakan, antara lain *Histogram Equalization* (HE), *Discrete Wavelet Transform* (DWT), *Stationary Wavelet Transform* (SWT), *Discrete Cosine Transform* (DCT), dan kombinasi metode DWT-SWT, DWT-SWT-HE, HE-DWT-SWT, dan DWT-SWT-HE-DCT. Dari beberapa metode yang digunakan akan dicari nilai akurasi dari keseluruhan *fold* dan waktu yang digunakan dalam proses klasifikasi.

[Halaman ini sengaja dikosongkan]

BAB 4

HASIL DAN PEMBAHASAN

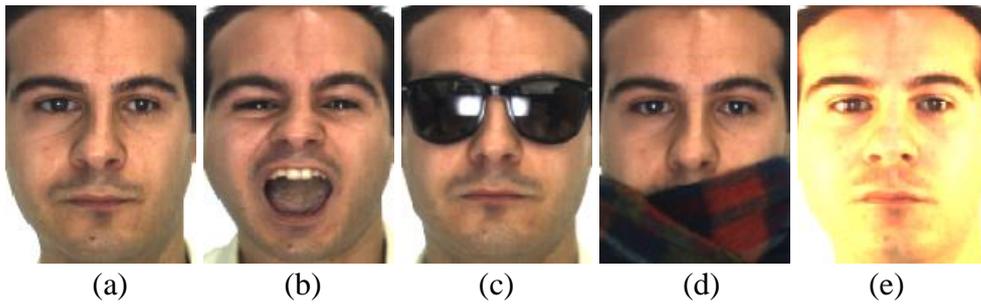
Bab ini akan membahas pemaparan hasil penelitian serta analisa terhadap hasil percobaan yang telah dilakukan.

4.1 Lingkungan Implementasi

Lingkungan implementasi sistem yang digunakan antara lain : (a) perangkat keras CPU Intel (R) Core (TM) i7-3770 @ 3.40 GHz dan RAM 8 GB, (b) Sistem Operasi *Windows 10*, (c) bahasa pemrograman Python, (d) *Interface Library* Keras.

4.2 Pembuatan *Dataset* Klasifikasi 10-fold Cross-Validation

Dataset dibagi menjadi 10 *fold* (lipatan/bagian), dimana data *test* yang digunakan per-*fold* dibagi sebagai berikut *fold* ke-1 berisi wajah normal, *fold* ke-2 berisi wajah dengan intensitas cahaya sebagian dan wajah berekspresi, *fold* ke-3 berisi wajah dengan intensitas cahaya penuh dan wajah dengan aksesoris kacamata, *fold* ke-4 berisi wajah dengan aksesoris kacamata beserta intensitas cahaya sebagian dan wajah dengan aksesoris syal (*scarf*), *fold* ke-5 berisi wajah dengan aksesoris syal beserta intensitas cahaya sebagian dan tapa intensitas cahaya, *fold* ke-6 berisi wajah normal dan wajah berekspresi, *fold* ke-7 berisi wajah berekspresi dan wajah dengan intensitas cahaya sebagian, *fold* ke-8 berisi wajah dengan intensitas cahaya penuh dan wajah dengan aksesoris kacamata, *fold* ke-9 berisi wajah dengan aksesoris kacamata beserta intensitas cahaya sebagian dan wajah dengan aksesoris syal, *fold* ke-10 berisi wajah dengan aksesoris syal beserta intensitas cahaya sebagian. Contoh data latih bisa dilihat pada Gambar 4.1.



Gambar 4.1 Gambar Asli (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk data *training* setiap *fold* yang digunakan adalah data yang tidak digunakan pada data *test*. Data *training* setiap *fold* berjumlah 2,340 sedangkan jumlah data *test* tiap *fold* adalah 260, dengan jumlah keseluruhan data yang digunakan adalah 2,600. Dari jumlah 2,600 terdapat 100 orang yang berbeda dan 26 citra yang berbeda untuk setiap orang.

4.3 Uji Coba

Pada tahap uji coba ini akan dilakukan pengujian jumlah *epoch* yang optimal dan pengujian citra wajah yang menggunakan metode dan yang tidak menggunakan metode ekstraksi fitur. Metode-metode yang digunakan pada pengujian ini adalah *Histogram Equalization* (HE), *Discrete Wavelet Transform* (DWT), *Stationary Wavelet Transform* (SWT), *Discrete Cosine Transform* (DCT), dan kombinasi metode DWT-SWT, DWT-SWT-HE, HE-DWT-SWT, dan DWT-SWT-HE-DCT.

4.3.1 Uji Coba Jumlah *Epoch*

Pada pengujian jumlah *epoch* yang akan di uji adalah 5, 10, 15, 20, 25, 30, 35, 40, 45, dan 50 hasil akurasi yang didapatkan bisa dilihat pada Tabel 4.1.

Tabel 4.1 Hasil Uji Jumlah *Epoch*

Epoch	Akurasi (%)
5	31.92
10	74.62
15	75
20	82.69

25	86.15
30	86.54
35	90
40	87.69
45	89.62
50	65

Pada uji coba menggunakan data normal didapatkan akurasi tertinggi berada pada jumlah *epoch* sebanyak 35 sebesar 90%, kemudian akurasi tersebut berkurang menjadi 87.69%.

4.3.2 Uji Coba Data Normal

Dataset yang digunakan pada uji coba hanya dilakukan *preprocessing* untuk setiap orang tanpa menggunakan metode ekstraksi fitur. Citra hasil dari proses *preprocessing* bisa dilihat pada Gambar 4.2.



Gambar 4.2 Gambar *Preprocessing* (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* untuk setiap *fold* bisa dilihat pada Tabel 4.2.

Tabel 4.2 Hasil Uji Coba Data Normal

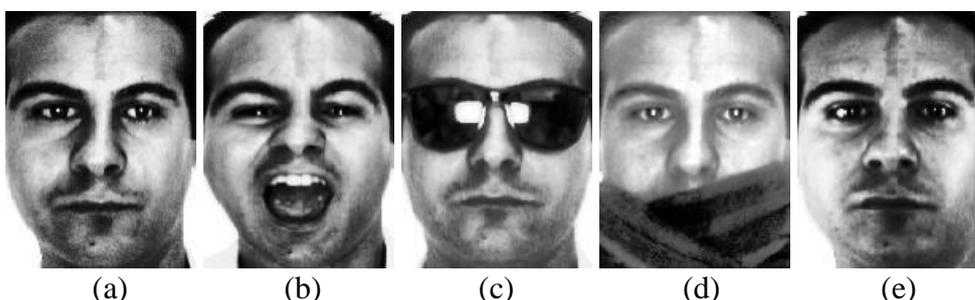
Fold	Akurasi (%)
1	91.54
2	86.92
3	98.08
4	83.08
5	29.23

6	95.38
7	85
8	97.31
9	90.77
10	3.85
Rata-rata	76.12

Pada uji coba menggunakan data normal didapatkan akurasi tertinggi berada pada *fold* ke-3 sebesar 98.08%, sedangkan hasil terendah terdapat pada *fold* ke-10 sebesar 3.85%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 34:01 Menit.

4.3.3 Uji Coba Data *Histogram Equalization* (HE)

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan metode *Histogram Equalization* (HE) untuk perbaikan citra. Citra hasil dari proses *preprocessing* dan HE bisa dilihat pada Gambar 4.3.



Gambar 4.3 Gambar HE (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan HE untuk setiap *fold* bisa dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Uji Coba Data *Histogram Equalization* (HE)

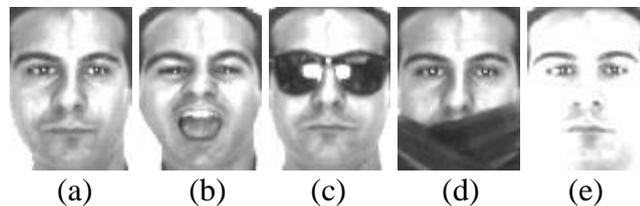
Fold	Akurasi (%)
1	94.62
2	93.08
3	98.85
4	81.54

5	59.23
6	96.15
7	92.31
8	57.69
9	88.46
10	45.38
Rata-rata	80.73

Pada uji coba menggunakan data *Histogram Equalization* (HE) didapatkan akurasi tertinggi berada pada *fold* ke-3 sebesar 98.85%, sedangkan hasil terendah terdapat pada *fold* ke-10 sebesar 45.38%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:51 Menit.

4.3.4 Uji Coba Data *Discrete Wavelet Transform* (DWT)

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan metode *Discrete Wavelet Transform* (DWT) untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan DWT bisa dilihat pada Gambar 4.4.



Gambar 4.4 Gambar DWT (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan DWT untuk setiap *fold* bisa dilihat pada Tabel 4.4.

Tabel 4.4 Hasil Uji Coba Data *Discrete Wavelet Transform* (DWT)

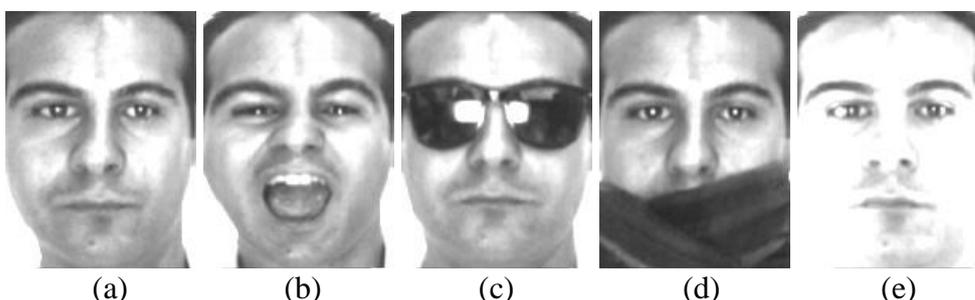
Fold	Akurasi (%)
1	89.62
2	87.69
3	96.92
4	86.15
5	66.92

6	94.23
7	90.38
8	97.31
9	86.92
10	62.31
Rata-rata	85.85

Pada uji coba menggunakan data *Discrete Wavelet Transform* (DWT) didapatkan akurasi tertinggi berada pada *fold* ke-8 sebesar 97.31%, sedangkan hasil terendah terdapat pada *fold* ke-10 sebesar 62.31%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 08:36 Menit.

4.3.5 Uji Coba Data *Stationary Wavelet Transform* (SWT)

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan metode *Stationary Wavelet Transform* (SWT) untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan SWT bisa dilihat pada Gambar 4.5.



Gambar 4.5 Gambar SWT (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan SWT untuk setiap *fold* bisa dilihat pada Tabel 4.5.

Tabel 4.5 Hasil Uji Coba Data *Stationary Wavelet Transform* (SWT)

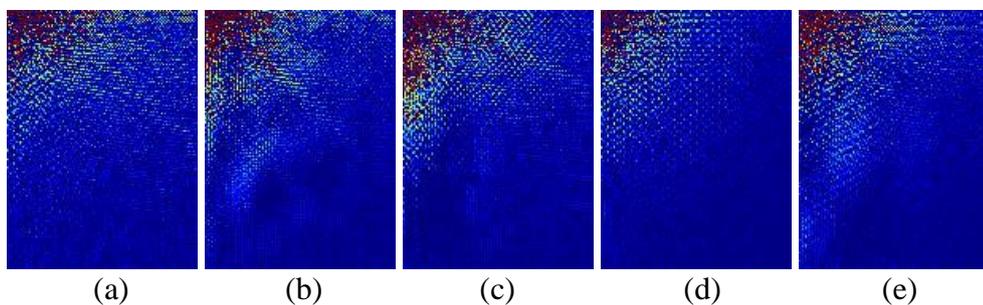
Fold	Akurasi (%)
1	90.77
2	81.54
3	17.69
4	59.23

5	61.15
6	76.92
7	91.15
8	94.23
9	0.77
10	69.23
Rata-rata	64.27

Pada uji coba menggunakan data *Stationary Wavelet Transform* (SWT) didapatkan akurasi tertinggi berada pada *fold* ke-8 sebesar 94.23%, sedangkan hasil terendah terdapat pada *fold* ke-9 sebesar 0.77 %. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:48 Menit.

4.3.6 Uji Coba Data *Discrete Cosine Transform* (DCT)

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan metode *Discrete Cosine Transform* (DCT) untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan DCT bisa dilihat pada Gambar 4.6.



Gambar 4.6 Gambar DCT (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan DCT untuk setiap *fold* bisa dilihat pada Tabel 4.6.

Tabel 4.6 Hasil Uji Coba Data *Discrete Cosine Transform* (DCT)

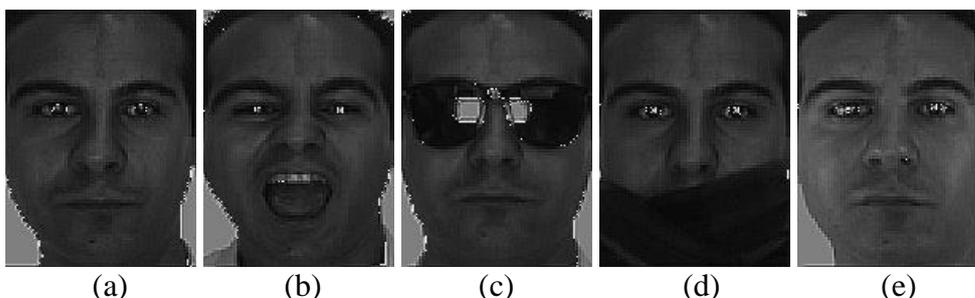
Fold	Akurasi (%)
1	90.38
2	80.38
3	98.46

4	87.31
5	85.77
6	95.77
7	87.69
8	95.38
9	89.23
10	84.62
Rata-rata	89.5

Pada uji coba menggunakan data *Discrete Cosine Transform* (DCT) didapatkan akurasi tertinggi berada pada *fold* ke-3 sebesar 98.46%, sedangkan hasil terendah terdapat pada *fold* ke-2 sebesar 80.38%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:50 Menit.

4.3.7 Uji Coba Data DWT dan SWT

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan kombinasi metode DWT dan SWT untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan kombinasi metode DWT dan SWT bisa dilihat pada Gambar 4.7.



Gambar 4.7 Gambar DWT - SWT (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan kombinasi metode DWT dan SWT untuk setiap *fold* bisa dilihat pada Tabel 4.7.

Tabel 4.7 Hasil Uji Coba Data DWT dan SWT

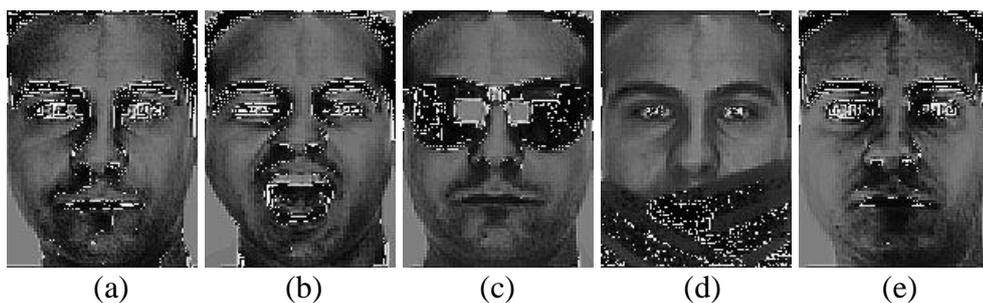
Fold	Akurasi (%)
1	92.31

2	87.69
3	97.31
4	80.77
5	73.08
6	94.62
7	92.31
8	96.15
9	82.69
10	71.92
Rata-rata	86.89

Pada uji coba menggunakan data DWT dan SWT didapatkan akurasi tertinggi berada pada *fold* ke-3 sebesar 97.31%, sedangkan hasil terendah terdapat pada *fold* ke-10 sebesar 71.92%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:42 Menit.

4.3.8 Uji Coba Data HE, DWT, dan SWT

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan kombinasi metode HE, DWT, dan SWT untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan kombinasi metode HE, DWT, dan SWT bisa dilihat pada Gambar 4.8.



Gambar 4.8 Gambar HE - DWT - SWT (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan kombinasi metode HE, DWT, dan SWT untuk setiap *fold* bisa dilihat pada Tabel 4.8.

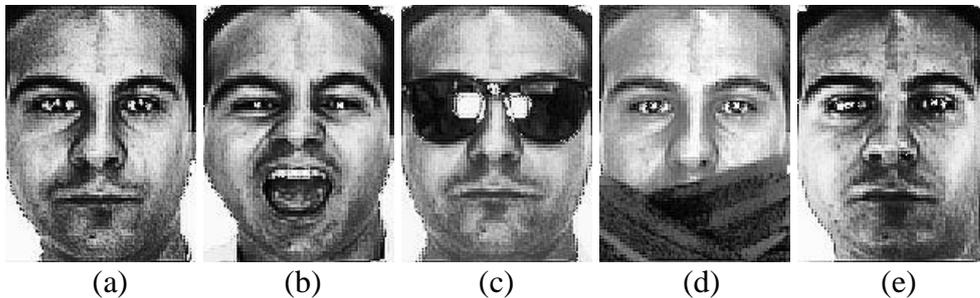
Tabel 4.8 Hasil Uji Coba Data HE, DWT, dan SWT

Fold	Akurasi (%)
1	85
2	86.92
3	91.15
4	10.77
5	37.31
6	96.54
7	87.69
8	90
9	70.38
10	41.92
Rata-rata	69.77

Pada uji coba menggunakan data HE, DWT, dan SWT didapatkan akurasi tertinggi berada pada *fold* ke-6 sebesar 96.54%, sedangkan hasil terendah terdapat pada *fold* ke-4 sebesar 10.77%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:03 Menit.

4.3.9 Uji Coba Data DWT, SWT, dan HE

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan kombinasi metode DWT, SWT, dan HE untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan kombinasi metode DWT, SWT, dan HE bisa dilihat pada Gambar 4.9.



Gambar 4.9 Gambar DWT - SWT - HE (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan kombinasi metode DWT, SWT, dan HE untuk setiap *fold* bisa dilihat pada Tabel 4.9.

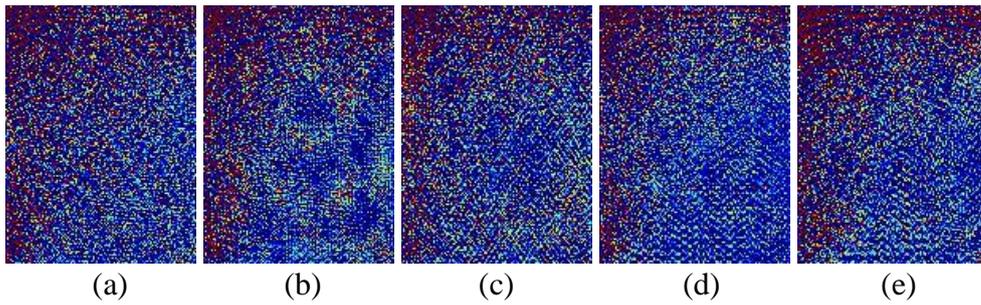
Tabel 4.9 Hasil Uji Coba Data DWT, SWT, dan HE

Fold	Akurasi (%)
1	95.77
2	88.08
3	99.23
4	85
5	59.62
6	96.15
7	20.38
8	97.31
9	67.31
10	65
Rata-rata	77.39

Pada uji coba menggunakan data DWT, SWT, dan HE didapatkan akurasi tertinggi berada pada *fold* ke-3 sebesar 99.23%, sedangkan hasil terendah terdapat pada *fold* ke-7 sebesar 20.38%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:08 Menit.

4.3.10 Uji Coba Data DWT, SWT, HE, dan DCT

Dataset yang digunakan pada uji coba dilakukan *preprocessing* dan menggunakan kombinasi metode DWT, SWT, HE, dan DCT untuk ekstraksi fitur. Citra hasil dari proses *preprocessing* dan kombinasi metode DWT, SWT, HE, dan DCT bisa dilihat pada Gambar 4.10.



Gambar 4.10 Gambar DWT - SWT - HE - DCT (a) Wajah Normal, (b) Berekspresi, (c) Aksesoris Kacamata, (d) Aksesoris Syal, dan (e) Intensitas Cahaya

Sedangkan untuk hasil akurasi data yang telah di *preprocessing* dan kombinasi metode DWT, SWT, HE, dan DCT untuk setiap *fold* bisa dilihat pada Tabel 4.10.

Tabel 4.10 Hasil Uji Coba Data DWT, SWT, HE, dan DCT

Fold	Akurasi (%)
1	93.85
2	89.23
3	99.62
4	93.08
5	85.38
6	96.54
7	90.77
8	98.46
9	93.08
10	87.31
Rata-rata	92.73

Pada uji coba menggunakan data DWT, SWT, HE, dan DCT didapatkan akurasi tertinggi berada pada *fold* ke-3 sebesar 99.62 %, sedangkan hasil terendah terdapat pada *fold* ke-5 sebesar 85.38%. Untuk waktu yang digunakan untuk proses klasifikasi adalah 33:56 Menit.

4.4 Pembahasan

Pada uji coba jumlah *epoch* didapatkan jumlah *epoch* yang optimal adalah 35 dengan akurasi sebesar 90%. Dan untuk jumlah *epoch* lebih dari 35 akan mengakibatkan *overfitting*. Jumlah *epoch* dengan nilai akurasi tertinggi yang akan digunakan pada tahap pengujian selanjutnya.

Pada uji coba data normal hasil terendah terdapat pada *fold* ke-10 sebesar 3.85% ini terjadi dikarenakan tidak dapat mengenali wajah dengan menggunakan aksesoris yang menutup separuh wajah. Beberapa faktor yang dapat mempengaruhinya di antaranya adalah faktor pencahayaan, ekspresi wajah dan perubahan atribut wajah seperti janggut, kumis, dan kacamata (Ravi, 2013).

Pada uji coba data HE didapatkan hasil akurasi terbaik pada pengujian ini terdapat pada *fold* 1, 2, 3, 6, dan 7 yaitu sebesar 94.62%, 93.08%, 98.85%, 96.15%, dan 92.31% hal ini lebih baik jika dibandingkan dengan data normal sebesar 91.54%, 86.92%, 98.08%, 95.38%, dan 85%. Hal tersebut dikarenakan metode HE dapat digunakan untuk meningkatkan kontras citra dan kualitas citra, sehingga informasi yang ada pada citra yang telah diperbaiki akan terlihat lebih jelas dibandingkan dengan citra asli sebelum diperbaiki (Mau, 2016).

Pada uji coba data DWT waktu yang digunakan untuk proses klasifikasi adalah 08:36 menit atau waktu tercepat jika dibandingkan dengan menggunakan metode lain. Hal tersebut terjadi dikarenakan DWT mengalami *downsampling* (Syahdeini, 2015). Perbandingan lama waktu yang digunakan untuk proses klasifikasi bisa dilihat pada Tabel 4.11.

Tabel 4.11 Perbandingan Waktu Proses Klasifikasi

Metode	Waktu (Menit)	Waktu Per Fold (Menit)	Waktu Per Citra (Milidetik)
Normal	34:01	03:24	785
HE	33:51	03:23	781
DWT	08:36	00:52	198
SWT	33:48	03:23	780

DCT	33:50	03:23	781
DWT-SWT	33:42	03:22	778
HE-DWT-SWT	33:03	03:18	763
DWT-SWT-HE	33:08	03:19	765
DWT-SWT-HE-DCT	33:56	03:15	749

Pada uji coba data SWT waktu yang digunakan untuk proses klasifikasi adalah 33:48 menit atau lebih lama dibandingkan dengan menggunakan metode DWT. Hal tersebut dikarenakan metode SWT merupakan metode yang dirancang untuk mengatasi *translation invariance* dari DWT dikarenakan tidak adanya *downsampling* (Syahdeini, 2015).

Pada uji coba data DCT dihasilkan akurasi yang menghasilkan akurasi yang lebih baik dibandingkan dengan citra normal yaitu sebesar 89.5% sedangkan akurasi citra normal sebesar 76.12%. Sedangkan waktu yang digunakan untuk proses klasifikasi adalah 33:50 menit atau lebih cepat dibandingkan dengan menggunakan data normal yaitu 34:01 menit. Hal tersebut dikarenakan metode DCT merupakan algoritma yang digunakan untuk membuat kompresi (*lossy compression*) yaitu penempatan data dimana tidak ada satu *byte* pun data yang hilang sehingga data tersebut utuh dan disimpan sesuai dengan aslinya (Krasmla, 2017).

Pada uji coba data DWT dan SWT menghasilkan akurasi yang lebih baik dibandingkan dengan citra normal yaitu sebesar 86.89% sedangkan akurasi citra normal sebesar 76.12%. Hal ini terjadi dikarenakan gabungan metode DWT dan SWT dapat menghasilkan citra resolusi tinggi (Hemalatha, 2017).

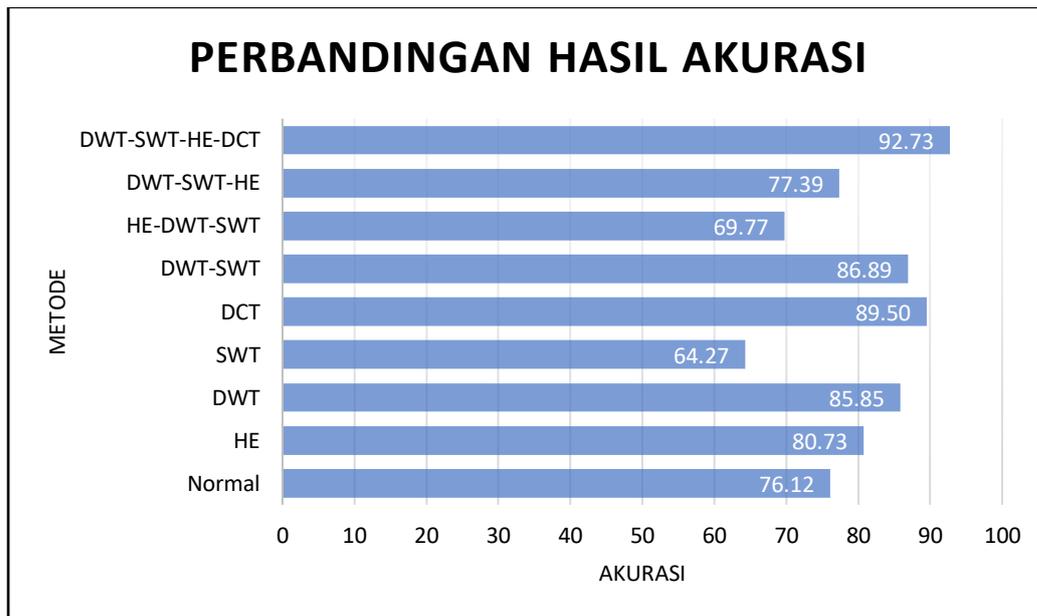
Pada uji coba penempatan metode HE dilakukan perbandingan hasil akurasi pada penempatan metode HE sebelum dan sesudah dilakukan penggabungan metode DWT dan SWT. Dimana hasil dari penempatan metode HE setelah dilakukan penggabungan metode DWT dan SWT menghasilkan akurasi yang lebih tinggi 77.39%, jika dibandingkan dengan menggunakan metode HE di awal yaitu sebesar 69.77%. Hal ini dikarenakan metode HE digunakan untuk

meningkatkan kontras citra dan kualitas citra, sehingga terjadinya perubahan informasi yang ada pada citra yang telah diperbaiki (Mau, 2016).

Pada uji coba dengan metode yang diusulkan terjadinya perbaikan hampir di seluruh *fold*. Akan tetapi masih terdapat 4 *fold* yang memiliki akurasi yang kurang baik dibandingkan dengan metode lain yaitu, pada *fold* 1, 2, 5, dan 7. Pada *fold* 1 gabungan metode DWT, SWT, dan HE dipengaruhi oleh hasil DCT. Pada *fold* 2 metode HE dipengaruhi oleh hasil DCT. Pada *fold* 5 metode DCT dipengaruhi oleh hasil gabungan metode DWT, SWT, dan HE. Pada *fold* 7 metode HE dan gabungan metode DWT dan SWT dipengaruhi oleh hasil gabungan metode DWT, SWT, dan HE. Perbandingan akurasi setiap *fold* pada keseluruhan metode yang diuji bisa dilihat pada Tabel 4.12 dan Gambar 4.11.

Tabel 4.12 Perbandingan Hasil Akurasi Setiap Metode

Fold	Normal	HE	DWT	SWT	DCT	DWT SWT	HE DWT SWT	DWT SWT HE	DWT SWT HE DCT
1	91.54	94.62	89.62	90.77	90.38	92.31	85	95.77	93.85
2	86.92	93.08	87.69	81.54	80.38	87.69	86.92	88.08	89.23
3	98.08	98.85	96.92	17.69	98.46	97.31	91.15	99.23	99.62
4	83.08	81.54	86.15	59.23	87.31	80.77	10.77	85	93.08
5	29.23	59.23	66.92	61.15	85.77	73.08	37.31	59.62	85.38
6	95.38	96.15	94.23	76.92	95.77	94.62	96.54	96.15	96.54
7	85	92.31	90.38	91.15	87.69	92.31	87.69	20.38	90.77
8	97.31	57.69	97.31	94.23	95.38	96.15	90	97.31	98.46
9	90.77	88.46	86.92	0.77	89.23	82.69	70.38	67.31	93.08
10	3.85	45.38	62.31	69.23	84.62	71.92	41.92	65	87.31
Jumlah	76.12	80.73	85.85	64.27	89.50	86.89	69.77	77.39	92.73



Gambar 4.11 Grafik Perbandingan Hasil Akurasi Setiap Metode

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dalam sub bab ini akan dipaparkan kesimpulan yang dapat diambil berdasarkan hasil rangkaian uji coba dan analisa penelitian yang dilakukan terhadap metode yang diusulkan. Kesimpulan tersebut adalah sebagai berikut:

1. Gabungan metode *Discrete Wavelet Transform* (DWT) dan *Stationary Wavelet Transform* (SWT) dapat meningkatkan kualitas citra wajah hal tersebut bisa dilihat akurasi pada data normal dihasilkan sebesar 76.12%, sedangkan dengan akurasi gabungan metode DWT dan SWT dihasilkan sebesar 86.89%.
2. Penempatan metode *Histogram Equalization* (HE) baik dilakukan setelah melakukan peningkatan resolusi menggunakan metode gabungan metode DWT dan SWT. Hal tersebut terbukti dari perbandingan hasil akurasi pada penempatan metode HE sebelum dan sesudah dilakukan penggabungan metode DWT dan SWT. Dimana hasil dari penempatan metode HE setelah dilakukan penggabungan metode DWT dan SWT menghasilkan akurasi yang lebih tinggi 77.39%, jika dibandingkan dengan menggunakan metode HE di awal yaitu sebesar 69.77%.
3. Metode yang diusulkan pada penelitian ini dapat menghasilkan peningkatan akurasi yaitu sebesar 92.73%, dibanding menggunakan metode HE sebesar 80.73%, DWT sebesar 85.85%, SWT sebesar 64.27%, DCT sebesar 89.5%, dan gabungan metode DWT dan SWT sebesar 86.89%.

5.2 Saran

Setelah dilakukannya pengujian, penulis memberikan beberapa saran untuk pengembangan penelitian selanjutnya, antara lain:

1. Perlu dilakukan penelitian lebih lanjut dengan menggunakan perbandingan ukuran citra melalui pemilihan nilai interpolasi dalam mempengaruhi hasil klasifikasi pada tahap peningkatan resolusi.

2. Perlu dilakukan penelitian lebih lanjut pada tahap peningkatan resolusi yang dimana metode penggabungan DWT dan SWT yang sebelumnya menggunakan metode *Inverse* DWT, bagaimana jika menggunakan metode *Inverse* SWT.
3. Perlu dilakukan penelitian lebih lanjut pada tahap perbaikan citra jika menggunakan metode selain HE.
4. Perlu dilakukan pengembangan lebih lanjut untuk bisa diterapkan pada bentuk *mobile* atau *website*.

DAFTAR PUSTAKA

- Chemchem, A., Alin, F., dan Krajecki, M. (2019), "Combining SMOTE Sampling and Machine Learning for Forecasting Wheat Yields in France", *Proceedings of Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, hal. 9- 4, Sardinia, Italia.
- Demirel, H., dan Anbarjafari, G. (2011), "IMAGE Resolution Enhancement by Using Discrete and Stationary Wavelet Decomposition", *Transactions on Image Processing*, VOL. 20, NO. 5, hal. 1458- 1460, New York, USA.
- Deng, L., dan Yu, D. (2014), "Deep Learning: Methods and Applications", *Foundations and Trends in Signal Processing*, Vol.7, No.3-4, hal 197-387, USA.
- Faruqi, A.A., dan Rozi, I.F. (2015), "Implementasi Steganography Menggunakan Algoritma Discrete Cosine Transform", *Jurnal Informatika Polinema*, Vol. 2, No. 1, hal. 35-39, Malang, Indonesia.
- Gani, S., dan Setiyono, B. (2018), "Teknik Invisible Watermarking Digital Menggunakan Metode DWT (Discrete Wavelet Tarnsform)", *Jurnal Sains Dan Seni ITS*. Vol. 7, No. 2, hal. 24-30, Surabaya, Indonesia.
- Gao, G., Yu, Y., Yang, M., Huang, P., Ge, Q., dan Yue. D. (2020), "Multi-scale patch based representation feature learning for low-resolution face recognition", *Applied Soft Computing Journal*, Vol. 90, hal. 1-17.
- Gehlot, S., Gupta, A., dan Gupta, R. (2020), "SDCT-AuxNet θ : DCT augmented stain deconvolutional CNN with auxiliary classifier for cancer diagnosis", *Medical Image Analysis*, Vol. 61, hal. 1-15, UK.
- Gonzalez, R.C., dan Woods R.E. (2008), *Digital Image Processing*, 3rd edition, Prentice-Hall. Inc, New Jersey.
- Hemalatha, M., Varadarajan, S., dan Babu, Y.M.M. (2017), "Comparison of DWT, DWT-SWT, and DT-CWT for low resolution satellite images enhancement", *International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET)*, hal. 1- 5, Chennai, India

- Jaya, I.N.S. 2002. *Penginderaan Jauh Satelit untuk kehutanan*. Laboratorium Inventarisasi Hutan”, IPB Press, Bogor.
- Krasmala, R., Budimansyah, A., dan Lenggana, U.T. (2017), “Kompresi Citra Dengan Menggabungkan Metode Discrete Cosine Transform (DCT) dan Algoritma Huffman”, *Jurnal Online Informatika*. Vol 2, No 1, hal 1-9, Bandung.
- Khan, S., Ahmed, E., Javed, M.H., Shah, S.A.A., dan Ali, S.U. (2019), “Transfer Learning of a Neural Network Using Deep Learning to Perform Face Recognition”, *International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, hal. 1-5, Swat, Pakistan.
- Kingma, D.P., dan Ba, J.L. (2015), “ADAM: A Method For Stochastic Optimization”. *International Conference On Learning Representations*, hal 1-15, San Diego, USA.
- Lesnussa, Y.A., Sinay, L.J., dan Idah, M.R. (2017), “Aplikasi Jaringan Saraf Tiruan Backpropagation untuk Penyebaran Penyakit Demam Berdarah Dengue (DBD) di Kota Ambon”, *Jurnal Matematika Integratif*, Vol. 13, No. 2 hal. 63–72, Sumedang, Indonesia
- Mahmud, F., Khatun, M.T., Zuhori, S.T., Afroge, S., Aktar, M., dan Pal, B. (2015), “Face recognition using Principle Component Analysis and Linear Discriminant Analysis”, *Proceedings of Second Electrical Engineering and Information Communication Technology (ICEEICT)*, hal. 1-4, Dhaka, Bangladesh.
- Manju D., dan Radha, V. (2019), “A Novel Approach for Pose Invariant Face Recognition in Surveillance Videos”, *Proceedings of International Conference on Computational Intelligence and Data Science (ICCIDS)*, Vol. 167, hal. 890-899, Gurgaon, India.
- Martinez, A.M., dan Benavente, R. (1998), The AR Face Database, CVC Technical Report #24.
- Martinez, A.M., dan Kak, A. C. (2001), "PCA versus LDA", *Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, hal. 228-233.

- Marti, N.W. (2010), "Pemanfaatan GUI Dalam Pengembangan Perangkat Lunak Pengenalan Citra Wajah Manusia menggunakan Metode Eigenface", *Conference on SNATI (Seminar Nasional Aplikasi Teknologi Informasi)*, hal. 11-16, Yogyakarta, Indonesia.
- Mohammed, N.N., Khaleel, M.I., Latif, M., dan Khalid, Z. (2018), "Face Recognition Based on PCA with Weighted and Normalized Mahalanobis distance", *International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, hal. 267-267, Bangkok, Thailand.
- Mulyono, I.U.W., Setiadi, D.R.I.M., Susanto, A., Rachmawanto, E.H., Fahmi, A., dan Muljono. (2019), "Performance Analysis of Face Recognition using Eigenface Approach", *Conference on International Seminar on Application for Technology of Information and Communication (iSemantic)*, Vol. 1, hal. 12-16, Semarang, Indonesia.
- Mau, S.D.B. (2016), "Pengaruh Histogram Equalization Untuk Perbaikan Kualitas Citra Digital", *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, Vol. 7, No. 1, hal 177-182, Kudus, Indonesia.
- Qu, X., Wei, T., Peng, C., dan Dua, P. (2018), "A Fast Face Recognition System Based on Deep Learning", *Conference on 11th International Symposium on Computational Intelligence and Design (ISCID)*, Vol. 1, hal. 289-292, Hangzhou, China.
- Ravi, S., dan Nayeem, S. (2013), "A Study on Face Recognition Technique based on Eigenface". *International Journal of Applied Information Systems (IJ AIS)*, Vol. 5, No. 4, hal. 57-62, New York, USA.
- Reshma, M.R., dan Kannan, B. (2018), "Approaches on Partial Face Recognition: A Literature Review", *International Conference on Trends in Electronics and Informatics (ICOEI)*, hal. 538-544, Tirunelveli, India.
- Santoso, A., dan Ariyanto, G. (2018), "Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah", *Jurnal Emitor*, Vol.18, No. 01, hal 15-21, Surakarta, Indonesia.
- Senthilkumar, R., dan Gnanamurthy, R.K. (2017), "Performance improvement in classification rate of appearance based statistical face recognition methods

- using SVM classifier”, *Proceedings of Fourth International Conference on Advanced Computing and Communication Systems (ICACCS)*, hal. 1-7, Coimbatore, India.
- Sivanandam S.N. (2009), *Introduction to Artificial Neural Network*, Vikas, New Delhi.
- Sripathi, D. (2003), *Efficient Implementations of Discrete Wavelet Transform using FPGAs*, Master, Florida State University, Florida.
- Suja, S., George, N., dan George, A. (2018). “Classification of Grades of Astrocytoma Images from MRI Using Deep Neural Network”, *Proceedings of Second International Conference on Trends in Electronics and Informatics*. Vol. 61, UK.
- Sulistiyo, W., Suyanto, B., Hestiningih, I., Mardiyono, dan Sukamto (2014), “Rancang Bangun Prototipe Aplikasi Pengenalan Wajah untuk Sistem Absensi Alternatif dengan Metode Haar Like Feature dan Eigenface”, *JTET (Jurnal Teknik Elektro Terapan)*, Vol. 3, No. 2, hal 93-98, Semarang, Indonesia
- Suma'inna dan Gumilar, G. (2013), “Implementasi Transformasi Wavelet Daubechies Pada Kompresi Citra Digital”, *CAUCHY - Jurnal Matematika Murni Dan Aplikasi*, Vol.2, No. 4, hal. 211-215, Malang, Indonesia
- Sutoyo, T., Mulyanto, E., Suhartono, V., dan Nurhayati, O.D. (2009), *Teori Pengolahan Citra Digital*, Andi, Yogyakarta.
- Syahdeini, A. (2015), *Klasifikasi Emosi Dari Data Sinyal EEG Menggunakan Independent Component Analysis (ICA), Wavelet Denoising Dan Multiple Discriminant Analysis (MDA)*, Thesis, Institut Teknologi Sepuluh Nopember, Surabaya.
- Tabassuma, F., Islama, M.I., Khan, R.T., dan Ami, M.R. (2020), “Human face recognition with combination of DWT and machine learning”, *Journal of King Saud University - Computer and Information Sciences*, hal. 1-11, Riyadh, Arab Saudi.

- Tabian, I., Fu, H., dan Khodaei, Z.S. (2019), “A Convolutional Neural Network for Impact Detection and Characterization of Complex Composite Structures”, *Sensors*, Vol. 19, No. 22, hal. 1-25, Basel, Switzerland.
- Tanaka, M. dan Okutomi, M. (2014), “A novel inference of a restricted boltzmann machine”, *Proceedings of 22nd International Conference on Pattern Recognition (ICPR)*, hal. 1526-1531, Stockholm, Sweden.
- Terzija, N. (2006), *Robust Digital Image Watermarking Algorithms for Copyright Protection*, Thesis Ph.D., Universität Duisburg, Essen.
- Toleubay Y., dan James A.P. (2020), “Getting Started with TensorFlow Deep Learning”, *Deep Learning Classifiers with Memristive Networks*, hal 57-67, Springer, Cham
- Umar, F., dan Darwis, H. (2019), “Watermarking Citra Digital Berwarna Menggunakan Stationary Wavelet Transform (SWT)”, *ILKOM Jurnal Ilmiah*, Vol. 11, No. 1, hal. 1-10, Makassar, Indonesia
- Vishwakarma, V.P., Dalal, S., dan Sisaudia, V. (2018), “Efficient Feature Extraction using DWT-DCT for Robust Face Recognition under varying Illuminations”, *IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, hal. 982-987, New Delhi, India.
- Wang, M., Jiang, H., dan Li, Y. (2010), “Face Recognition based on DWT/DCT and SVM”, *International Conference on Computer Application and System Modeling (ICCASM)*, Vol 3, hal. 507-510, Taiyuan, China.
- Watson, A.B. (1994). “Image Compression Using the Discrete Cosine Transform”. *Mathematica Journal*. Vol. 4, No. 1, hal. 81-88, Canada, USA.
- Zhou, L., Li, W., Du, Y., Lei, B., dan Liang, S. (2019), “Adaptive illumination-invariant face recognition via local nonlinear multi-layer contrast feature”, *Journal of Visual Communication and Image Representation*. Vol. 64, hal 1-13.

[Halaman ini sengaja dikosongkan]

LAMPIRAN

Akurasi 5 Epoch

Epoch	Akurasi
1	0.77
2	5.77
3	10
4	25
5	31.92

Akurasi 10 Epoch

Epoch	Akurasi	Epoch	Akurasi
1	1.15	6	55.38
2	6.15	7	58.08
3	11.92	8	67.31
4	22.69	9	79.23
5	48.85	10	74.62

Akurasi 15 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	1.92	6	51.54	11	73.85
2	4.62	7	53.08	12	72.69
3	19.62	8	67.31	13	76.92
4	26.54	9	61.92	14	80.38
5	40	10	63.85	15	75

Akurasi 20 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	3.08	6	49.62	11	74.62	16	85
2	3.46	7	63.85	12	79.23	17	83.08
3	11.92	8	57.69	13	78.85	18	81.54
4	18.46	9	61.15	14	79.23	19	87.31
5	37.69	10	67.31	15	82.69	20	82.69

Akurasi 25 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	0	11	65	21	74.62
2	5	12	63.85	22	86.54
3	6.54	13	73.08	23	80
4	10.38	14	78.08	24	83.85
5	25	15	68.08	25	86.15
6	28.85	16	74.62		
7	46.15	17	74.23		
8	45	18	77.31		
9	56.92	19	76.15		
10	53.08	20	78.85		

Akurasi 30 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	2.69	11	69.23	21	88.46
2	5	12	78.08	22	86.92
3	15	13	73.85	23	89.23

4	14.62	14	70.38	24	87.69
5	30.38	15	79.62	25	88.46
6	35.77	16	76.15	26	85.38
7	52.69	17	84.23	27	88.85
8	55	18	88.08	28	88.46
9	70.38	19	90	29	88.46
10	67.31	20	87.31	30	86.54

Akurasi 35 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	3.46	11	71.92	21	83.46	31	87.69
2	8.08	12	76.92	22	84.23	32	91.15
3	13.46	13	76.54	23	87.31	33	88.85
4	35.77	14	81.54	24	81.92	34	91.54
5	38.85	15	88.85	25	81.15	35	90
6	58.08	16	82.31	26	81.15		
7	67.31	17	85.77	27	86.92		
8	61.92	18	81.54	28	85.77		
9	71.92	19	77.69	29	81.54		
10	73.85	20	85.38	30	83.46		

Akurasi 40 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	0	11	53.85	21	71.15	31	85
2	3.08	12	64.23	22	83.08	32	89.23
3	4.62	13	65	23	87.31	33	74.62

4	9.231	14	73.85	24	83.85	34	83.85
5	16.15	15	70.77	25	87.31	35	87.31
6	16.54	16	75	26	79.23	36	89.23
7	34.23	17	70.77	27	85.38	37	88.85
8	41.54	18	78.08	28	79.23	38	88.85
9	48.08	19	72.31	29	84.23	39	89.62
10	53.08	20	79.62	30	84.23	40	87.69

Akurasi 45 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	1.15	16	65.77	31	83.08
2	3.08	17	71.54	32	83.08
3	4.23	18	71.15	33	84.62
4	6.92	19	71.92	34	88.85
5	16.54	20	62.69	35	82.69
6	19.23	21	70.38	36	86.54
7	25	22	77.31	37	88.85
8	39.23	23	80	38	83.46
9	37.31	24	79.62	39	85.38
10	40.38	25	83.46	40	80.38
11	50	26	77.69	41	85.77
12	53.85	27	80	42	88.85
13	61.92	28	83.46	43	91.54
14	52.69	29	83.08	44	89.62
15	61.54	30	85	45	89.62

Akurasi 50 Epoch

Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi	Epoch	Akurasi
1	3.46	16	59.23	31	61.54	46	68.08
2	6.92	17	57.69	32	62.69	47	60.77
3	17.69	18	53.08	33	59.62	48	63.46
4	20.77	19	54.62	34	64.62	49	63.08
5	28.85	20	52.31	35	58.46	50	65
6	31.54	21	53.46	36	61.54		
7	34.23	22	60.77	37	61.92		
8	43.85	23	51.54	38	66.15		
9	35	24	58.85	39	66.92		
10	38.08	25	53.85	40	55.38		
11	46.92	26	60.77	41	59.23		
12	43.08	27	64.62	42	62.69		
13	29.23	28	60.77	43	62.69		
14	52.31	29	60	44	61.92		
15	50.77	30	60	45	61.15		

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Afrizal Laksita Akbar, akrab dipanggil rizal, lahir di Kediri, 29 April 1992. Putra kedua dari pasangan Bapak Qomarudin dan Ibu Nefi Nurafiati. Penulis menempuh pendidikan formal di SDN 1 Ngondang Pace, SMPN 1 Pace, SMAN 1 Nganjuk serta kuliah Sarjana di Universitas Brawijaya Malang. Kemudian tahun 2015 diterima sebagai mahasiswa Pasca Sarjana di Teknik Informatika ITS melalui jalur umum. Penulis dapat dihubungi melalui email: afrizal.la@gmail.com.