



**Pengenalan Signature (Tanda Tangan)
Menggunakan *Extreme Learning Machine* (ELM)**

SKRIPSI

Oleh:

Dailami Firdaus

NIM 121910201117

**PROGRAM STUDI STRATA 1 TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2019**



**Pengenalan Signature (Tanda Tangan)
Menggunakan *Extreme Learning Machine (ELM)***

SKRIPSI

Diajukan guna memenuhi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Teknik Elektro (S1)
dan mencapai gelar Sarjana Teknik

Oleh:

Dailami Firdaus

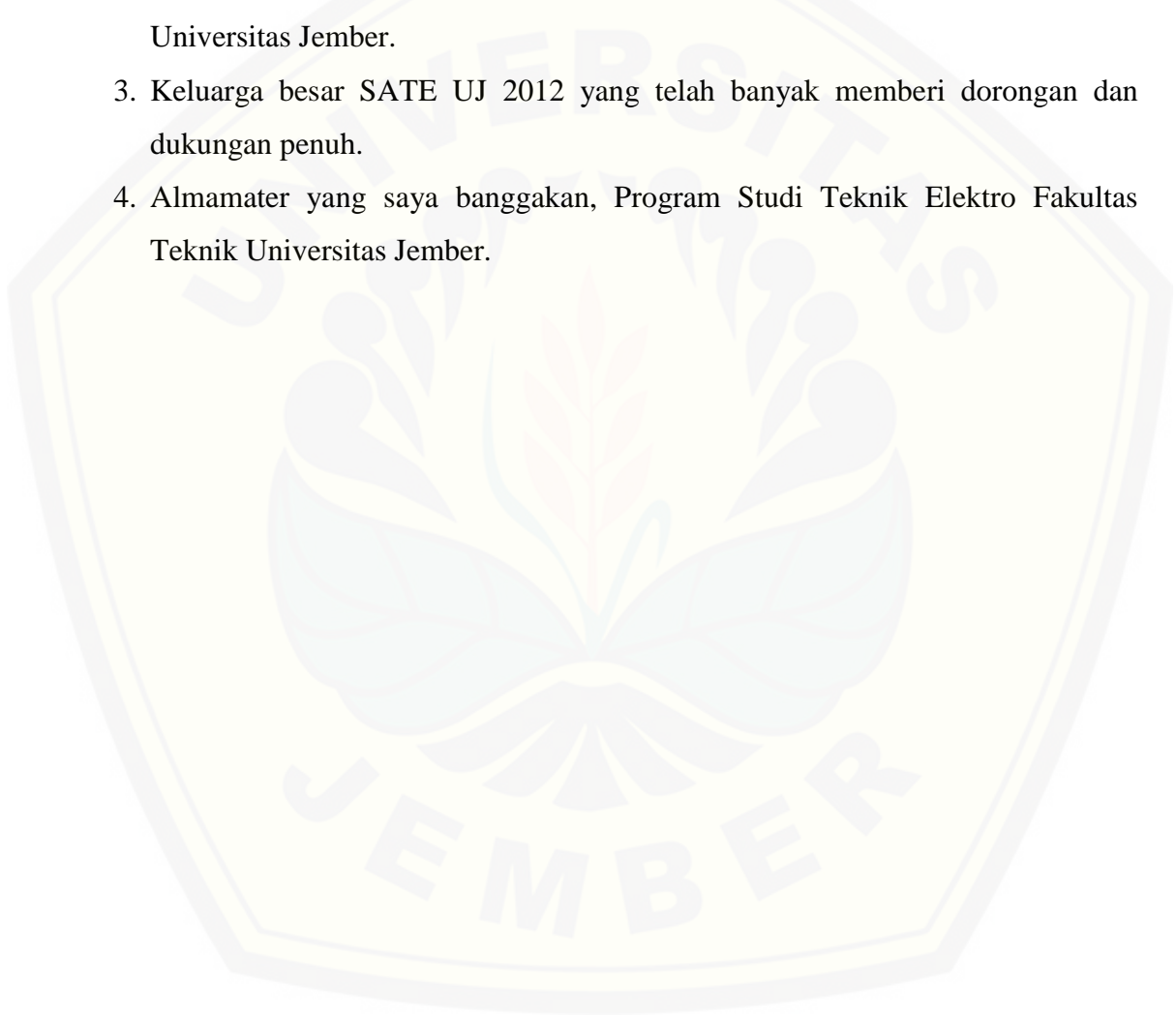
NIM 121910201117

**PROGRAM STUDI STRATA 1 TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2019**

Persembahan

Skripsi ini saya persembahkan untuk:

1. Kedua orang tua saya, Almarhum Ayah (Samsul Arifin), Almarhumah Ibu (Insaniah). Serta juga Kakak, Nenek, dan semua keluarga yang telah mendoakan.
2. Guru - guru SD Negeri Tisnonegaran 1 Probolinggo, SMP Negeri 1 Probolinggo, SMA Negeri 1 Dringu dan semua Dosen - dosen program studi Teknik Elektro Universitas Jember.
3. Keluarga besar SATE UJ 2012 yang telah banyak memberi dorongan dan dukungan penuh.
4. Almamater yang saya banggakan, Program Studi Teknik Elektro Fakultas Teknik Universitas Jember.



Motto

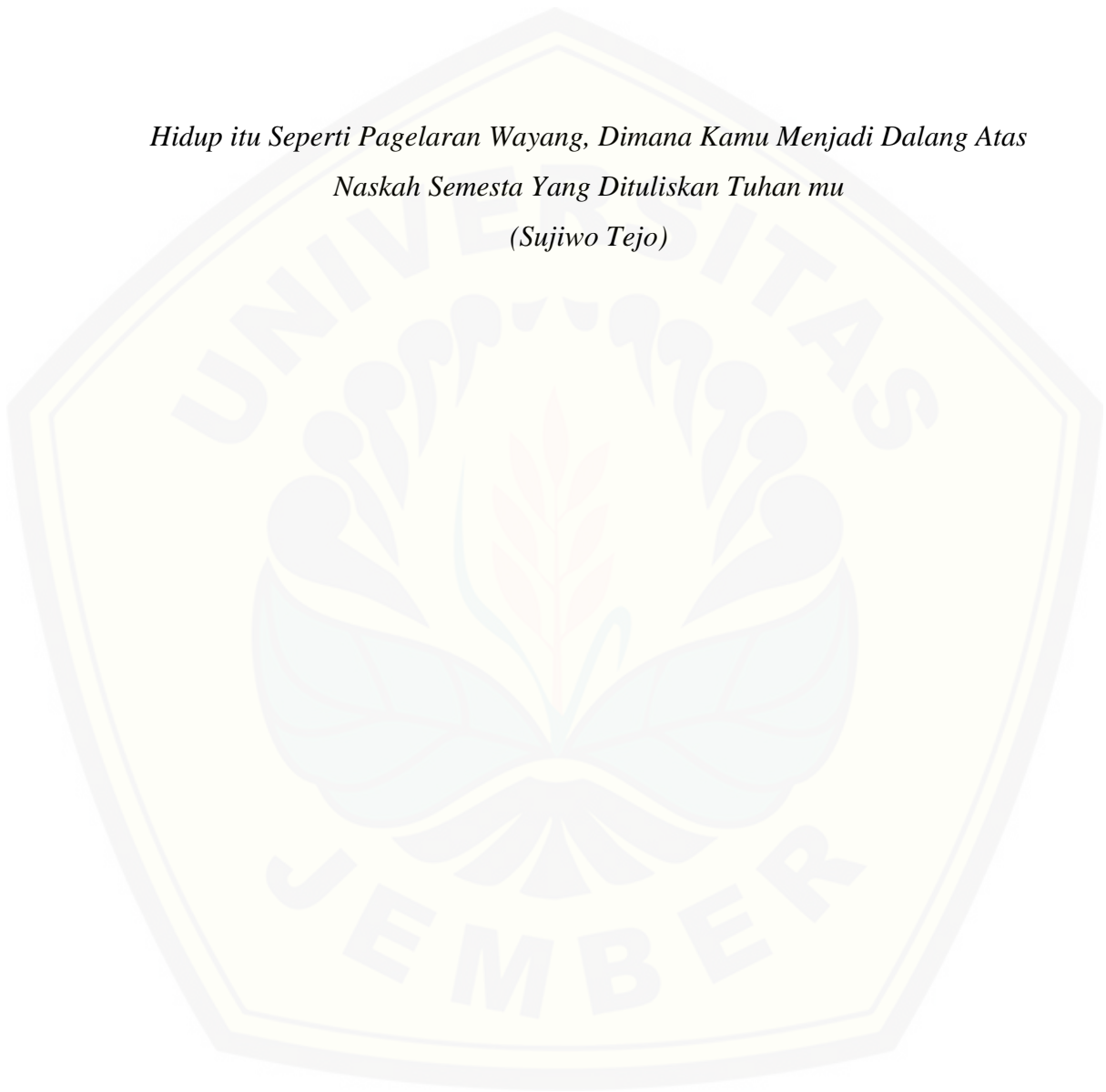
Berhenti Tidak Akan Membuatmu Tetap Hidup

(Penulis)

Hidup itu Seperti Pagelaran Wayang, Dimana Kamu Menjadi Dalang Atas

Naskah Semesta Yang Dituliskan Tuhan mu

(Sujiwo Tejo)



PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Dailami Firdaus

NIM : 121910201117

menyatakan dengan sesungguhnya bahwa karya tulis ilmiah yang berjudul “Pengenalan *Signature* (Tanda tangan) Menggunakan *Extreme Learning Machine* (ELM)” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, 27 Desember 2019

Yang menyatakan,

Dailami Firdaus
NIM. 121910201117

SKRIPSI

**Pengenalan Signature (Tanda Tangan)
Menggunakan *Extreme Learning Machine* (ELM)**

Oleh:

Dailami Firdaus
NIM 121910201117

Pembimbing:

Dosen Pembimbing Utama : Dodi Setiabudi, S.T., M.T.
NIP. 198405312008121004

Dosen Pembimbing Anggota : Catur Suko Sarwono, S.T.,
M.Si
NIP. 196801191997021001

PENGESAHAN

Skripsi berjudul “**Pengenalan Signature (Tanda Tangan) Menggunakan *Extreme Learning Machine* (ELM)**” telah diuji dan disahkan pada:

Hari, tanggal : Jum'at, 27 Desember 2019

Tempat : Ruang Ujian 1 Fakultas Teknik Universitas Jember

Dosen Pembimbing Utama,

Dosen Pembimbing Anggota,

Dodi Setiabudi, S.T., M.T.
NIP. 198405312008121004

Catur Suko Sarwono, S.T., M.Si
NIP. 196801191997021001

Penguji 1,

Penguji 2,

Widya Cahyadi, S.T., M.T.
NIP. 198511102014041001

Alfredo Bayu Satriya, S.T., M.T.
NIP. 198905192015041001

Mengesahkan
Dekan,

Dr. Ir. Entin Hidayah, M.U.M.
NIP. 196612151995032001

Pengenalan *Signature* (Tanda tangan) Menggunakan *Extreme Learning Machine* (ELM)

Dailami Firdaus

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

ABSTRAK

Tanda tangan (*Signature*) merupakan hasil dari proses menulis seseorang yang bersifat khusus sebagai substansi simbolik. Pada penelitian ini menggunakan 10 subjek yang akan diuji keaslian tanda tangannya. Objek yang akan digunakan sebagai data base discan agar dapat menghasilkan file yang nantinya akan di olah di aplikasi Atom. Kemudian file tersebut di ekstrak menjadi nilai Black and white menggunakan teknik *image processing*. Setelah itu hasil dari *image processing* tersebut ditraining oleh *Extreme Learning Machine* (ELM) menjadi nilai bias dan biner yang nantinya digunakan sebagai data base pengujian objek. Setelah tahap pengambilan database, gambar yang akan diuji kecocokannya berdasarkan database discan untuk menghasilkan file gambar. Kemudian file gambar tersebut diproses dan diuji oleh *Extreme Learning Machine*(ELM) apakah cocok dengan database yang telah didapatkan. Hasil pengujian pada penelitian ini, dapat dikatakan perubahan jumlah neuron atau perubahan struktur ELM (*Extreme Learning Machine*) dapat berpengaruh terhadap perubahan nilai akurasi database yang digunakan. Hal itu terbukti dengan terjadinya kenaikan dan penurunan nilai akurasi secara signifikan pada setiap perubahan rentang neuron (pada beberapa data tanda tangan subjek yang diuji). Akan tetapi ada beberapa data tanda tangan yang cenderung stabil nilai akurasinya, yaitu pada data subjek 03, subjek 05, dan subjek 08. Nilai akurasi semua subjek yang dihasilkan pada penelitian ini adalah 0,758850516.

Kata Kunci: *Signature, Extreme Learning Machine (ELM), Neuron.*

Signature Recognition Using Extreme Learning Machine (ELM)

Dailami Firdaus

Electrical Engineering, Engineering Faculty, Jember University

ABSTRACT

Signature (Signature) is the result of the process of writing someone who is special as a symbolic substance. In this study using 10 subjects to be tested for the authenticity of his signature. The object will be used as a scanned data base in order to produce files that will later be processed in the Atom application. Then the file is extracted into Black and white values using image processing techniques. After that the results of the image processing are trained by Extreme Learning Machine (ELM) to be biased and binary values which will be used as an object testing data base. After the database retrieval stage, the image to be tested for compatibility is based on the database scanned to produce an image file. Then the image file is processed and tested by Extreme Learning Machine (ELM) whether it matches the database that has been obtained. The test results in this study, it can be said that the change in the number of neurons or changes in the structure of the ELM (Extreme Learning Machine) can affect changes in the accuracy of the database used. This was evidenced by the increase and decrease in the value of accuracy significantly on each change in the range of neurons (on some of the signature data of the subjects tested). However, there are some signature data that tend to be stable, the accuracy of the data is subject data 03, subject 05, and subject 08. The accuracy value of all subjects produced in this study was 0.758850516.

Keywords: *Signature, Extreme Learning Machine (ELM), Neuron.*

RINGKASAN

Pengenalan *Signature* (Tanda tangan) Menggunakan *Extreme Learning Machine* (ELM); Dailami Firdaus, 121910201117; 2019; 81 halaman; Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Tanda tangan merupakan bentuk yang paling banyak digunakan untuk identifikasi seseorang. Fungsi tanda tangan sendiri adalah untuk pembuktian. Dalam kehidupan sehari-hari, tanda tangan digunakan sebagai identifikasi dari pemilik tanda tangan. Keberadaan tanda tangan dalam sebuah dokumen menyatakan bahwa pihak yang menandatangani, mengetahui, dan menyetujui seluruh isi dari suatu dokumen. Secara hukum tanda tangan yang dibubuhkan pada surat penting bisa mewakili diri orang yang bersangkutan.

Berdasarkan hasil pengujian pada penelitian ini, dapat dikatakan perubahan jumlah neuron atau perubahan struktur ELM (*Extreme Learning Machine*) dapat berpengaruh terhadap perubahan nilai akurasi database yang digunakan. Hal itu terbukti dengan terjadinya kenaikan dan penurunan nilai akurasi secara signifikan pada setiap perubahan rentang neuron (pada beberapa data tanda tangan subjek yang diuji). Akan tetapi ada beberapa data tanda tangan yang cenderung stabil nilai akurasinya, yaitu pada data subjek 03, subjek 05, dan subjek 08. Berdasarkan dari keseluruhan nilai akurasi data tanda tangan yang diuji, kemudian diambil rata-rata dari keseluruhan hasil pengujian tersebut. Rata-rata hasil nilai akurasi untuk keseluruhan subjek yaitu bernilai 0,758.

PRAKATA

Puji syukur kehadirat Allah SWT yang maha kuasa atas segalanya, karena dengan ridho, hidayah dan petunjuk-Nya, penulis dapat menyelesaikan skripsi ini. Selama penyusunan skripsi ini penulis mendapat bantuan berbagai pihak yang turut memberikan bantuan berupa motivasi, inspirasi, bimbingan, doa, fasilitas dan dukungan lainnya yang membantu memperlancar pengerjaan skripsi ini. Untuk itu penulis mengucapkan terimakasih kepada.

1. Ibu Dr. Ir. Entin Hidayah, M.U.M., Selaku Dekan Fakultas Teknik Universitas Jember;
2. Bapak Dr. Bambang Sri Kaloko, S.T., M.T., Selaku Ketua Jurusan Teknik Elektro Universitas Jember;
3. Bapak Dodi Setia Budi, S.T., M.T. dan bapak Catur Suko Sarwono, S.T., M.Si, selaku dosen pembimbing yang telah membimbing tugas akhir ini;
4. Bapak Widya Cahyadi, S.T., M.T. dan bapak Alfredo Bayu Satriya, S.T., M.T., selaku dosen penguji yang sudah memberikan saran untuk memperbaiki tugas akhir ini;
5. Almarhum kedua orang tua Bapak Samsul Arifin dan Ibu Insaniah, yang telah membesarkan, mendidik, mendoakan tiada henti, memberi motivasi semangat, menitikkan air mata dan memberi kasih sayang yang tak pernah habis serta pengorbanannya selama beliau ada;
6. Saudaraku Ghika Bangun Samudra dan Cyntia Virdiana;
7. Keluarga besar teknik elektro 2012 SATE UJ Serta semua pihak yang tidak dapat disebutkan satu per satu, yang telah mendukung dalam penyelesaian skripsi ini.

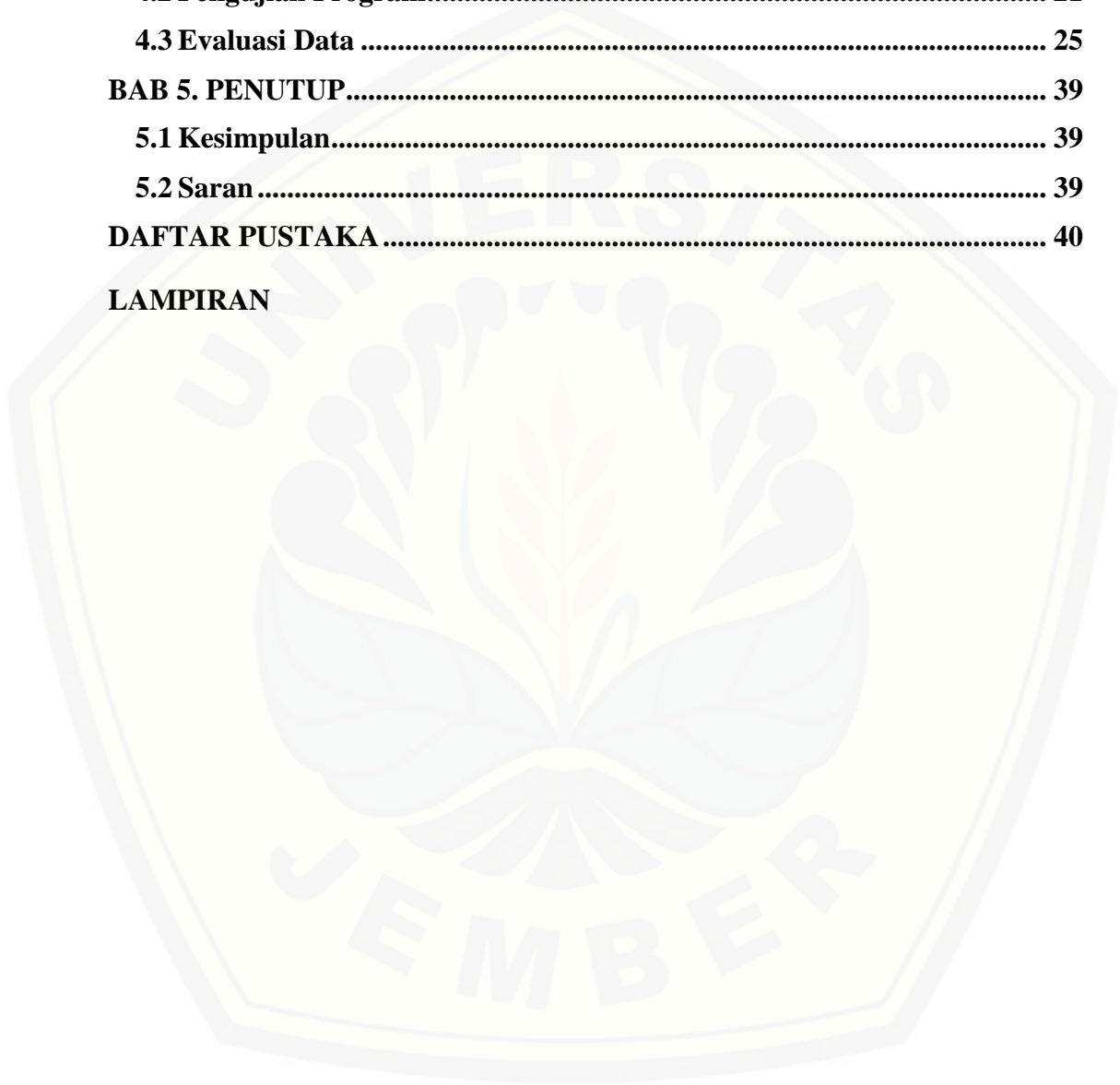
Jember, 27 Desember 2019

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN.....	iv
TUGAS AKHIR	v
HALAMAN PENGESAHAN.....	vi
ABSTRAK	vii
ABSTARK INGGRIIS	viii
RINGKASAN	ix
PRAKATA	x
DAFTAR ISI.....	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Peneitian.....	3
BAB 2. TINJAUAN PUSTAKA.....	4
2.1. <i>Neural Network</i>	4
2.2. <i>Extreme Learning Machine</i>	5
2.3. Citra Digital	9
2.3.1. Pengolahan Citra Warna	10
2.3.2. Citra <i>Gray-scale</i> (Keabuan)	12
2.3.3 Binerisasi Citra	13
2.3.4 <i>Thresholding</i>	14
BAB 3. METODELOGI PENELITIAN	15
3.1 Tahapan Penelitian.....	15

3.2 Perancangan Sistem	16
3.3 <i>Flow Chart</i>	18
BAB 4. HASIL DAN PEMBAHASAN	20
4.1 Pengambilan Data Subjek	20
4.2 Pengujian Program	21
4.3 Evaluasi Data	25
BAB 5. PENUTUP	39
5.1 Kesimpulan	39
5.2 Saran	39
DAFTAR PUSTAKA	40
LAMPIRAN	



DAFTAR GAMBAR

Gambar 2.1. Struktur <i>Neuron</i> Pada Otak Manusia	6
Gambar 2.2. Struktur <i>Extreme Learning Machine</i>	7
Gambar 2.3. Diagram Alir <i>Backpropagation</i> (ANN)	7
Gambar 2.4. Diagram Alir <i>Extreme Learning Machine</i> (ELM)	8
Gambar 2.5. Koordinat citra digital	9
Gambar 2.6. Matrik Citra Digital	10
Gambar 2.7. Blok diagram pengolah citra	11
Gambar 2.8. Nilai warna RGB dalam Hexadesimal	12
Gambar 2.9. Komposisi warna RGB	12
Gambar 2.10. Contoh konversi citra RGB ke citra abu-abu	13
Gambar 2.11. Citra Biner	13
Gambar 2.12. Contoh Derajat Kualitas <i>Tresholding</i>	14
Gambar 3.1. Blok Diagram Tahap Penelitian	15
Gambar 3.2. Blok Diagram Proses Pengambilan Data Base dan Pengujian Objek	16
Gambar 3.3. Flowchart Proses pengambilan Data Base	18
Gambar 3.4. Flowchart Proses Pengujian Objek	19
Gambar 4.1 Sampel Subjek 01	21
Gambar 4.2 Jumlah tanda tangan yang dibutuhkan untuk masing masing subjek	21
Gambar 4.3 Program Untuk Mengolah Data Gambar Menjadi Data Biner	22
Gambar 4.4 Data <i>Author</i> 1	22
Gambar 4.5 Data <i>Black White Author</i> 1	23
Gambar 4.6 Data <i>Resize Author</i> 1	23
Gambar 4.7 Program ELM Untuk Pengujian	24
Gambar 4.8 Data <i>Train</i> dan Data <i>Test</i>	24
Gambar 4.9 Proses <i>Running</i> Sistem	25
Gambar 4.10 Grafik akurasi subjek 01	26
Gambar 4.11 Grafik akurasi subjek 02	27
Gambar 4.12 Grafik akurasi subjek 03	28
Gambar 4.13 Grafik akurasi subjek 04	29

Gambar 4.14 Grafik akurasi subjek 05	31
Gambar 4.15 Grafik akurasi subjek 06	32
Gambar 4.16 Grafik akurasi subjek 07	33
Gambar 4.17 Grafik akurasi subjek 08	35
Gambar 4.18 Grafik akurasi subjek 09	36
Gambar 4.19 Grafik akurasi subjek 10	37



DAFTAR TABEL

Tabel 4.1. Data Akurasi Subjek 01	25
Tabel 4.2. Data Akurasi Subjek 02	26
Tabel 4.3. Data Akurasi Subjek 03	28
Tabel 4.4. Data Akurasi Subjek 04	29
Tabel 4.5. Data Akurasi Subjek 05	30
Tabel 4.6. Data Akurasi Subjek 06	31
Tabel 4.7. Data Akurasi Subjek 07	33
Tabel 4.8. Data Akurasi Subjek 08	34
Tabel 4.9. Data Akurasi Subjek 09	35
Tabel 4.10. Data Akurasi Subjek 10	36
Tabel 4.11. Data Rata-Rata Akurasi	38

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Tanda tangan (*Signature*) merupakan hasil dari proses menulis seseorang yang bersifat khusus sebagai substansi simbolik. Tanda tangan merupakan bentuk yang paling banyak digunakan untuk identifikasi seseorang. Fungsi tanda tangan sendiri adalah untuk membuktikan. Dalam kehidupan sehari-hari, tanda tangan digunakan sebagai identifikasi dari pemilik tanda tangan. Keberadaan tanda tangan dalam sebuah dokumen menyatakan bahwa pihak yang menandatangani, mengetahui, dan menyetujui seluruh isi dari suatu dokumen.

Secara hukum tanda tangan yang dibubuhkan pada surat penting bisa mewakili diri orang yang bersangkutan. Tanda tangan dianggap lebih bernilai hukum. Betapa penting artinya, sampai sampai hakim pun bisa menjebloskan seseorang ke dalam penjara, selembar kertas *cek* bisa ditukar dengan uang kontan, dan sebagainya. Karena punya arti penting itu (secara hukum bisa mewakili diri seseorang), sehingga muncul tindakan pemalsuan tanda tangan cek atau surat penting lainnya.

Tak seorang pun mengajarkan bagaimana merancang atau memilih tanda tangan yang cocok untuk masing-masing personal. Setiap orang menciptakan sendiri tanda tangannya, setelah berkali-kali mencoba corat-coret di kertas kosong. Keragu-raguan dalam membuat tanda tangan muncul ketika remaja. Seturut jalannya waktu tanda tangan akan mengalami banyak perubahan. Selain faktor usia, ia bisa berubah karena perubahan status (pernikahan atau sosial). Bahkan, menurut Anna Koren, tanda tangan bisa berubah kapan saja, selama yang bersangkutan masih hidup. Sementara menurut Margaret Gullan-Whur, perubahan mendadak pada tanda tangan mungkin saja disengaja karena ketidakpuasan penampilan tulisan, yang mengungkapkan ketidakpuasan terhadap diri sendiri. Pada remaja hal ini bisa terjadi. Setelah dewasa, sulit untuk mengubah format tanda tangan tanpa mengubah bagian lain.

Menurut John Nimpoeno, ada dua teknik yang bisa diterapkan untuk menentukan palsu-tidaknya sebuah tanda tangan, yaitu teknik mekanis (melibatkan

peralatan tertentu) dan psikofisiologis (meminjam kaidah-kaidah yang berlaku dalam grafologi tapi secara terbatas). Pada teknik analisis psikofisiologis masih dibutuhkan tulisan tangan dari orang yang dianggap sebagai pemilik tanda tangan, dan orang yang dicurigai memalsukan tanda tangan. Tulisan tangan itu dipakai untuk perbandingan. Hasil pengamatan cara ini bisa lemah kalau tidak dilakukan oleh psikolog atau grafolog. Sedangkan pemeriksaan tanda tangan secara mekanis biasanya menggunakan peralatan khusus. Pengamatannya lebih menekankan pada hal-hal berkaitan dengan pemalsuan mekanis.

Tanda tangan memiliki informasi yang menarik, tetapi tidak dapat dipelaskan dari tulisan tangan. Seseorang tidak pernah ada yang membuat tandanya sendiri yang identik, berapapun tanda tangan yang akan dibuat. Tanda tangan dibuat dengan kesadaran penuh oleh pemiliknya, dikatakan secara terus menerus sehingga membuat formula yang baku. Hal tersebut diikuti dengan doktrin-doktrin yang menjelaskan makna dari stimulan yang dimaksud. Pada saat yang sama seseorang mulai memikirkan bentuk huruf yang akan ia buat. Pada proses tahap awal tentu banyak cacat yang dialami oleh orang tersebut. Setelah melalui beberapa proses, seseorang akan dapat mengenali lebih banyak huruf dengan ragam varian makna yang ada. Seiring berjalannya waktu seseorang akan menjalankannya dengan reflek atau naluri tanpa perlu banyak berpikir.

Berdasarkan latar belakang tersebut, penulis mempunyai gagasan bahwa proses pembacaan tanda tangan yang dapat meringankan kinerja manusia menggunakan metode *Image Processing* dan *Neural network* yang berjudul "Pengenalan tanda tangan (signature) menggunakan *Extreme Learning Machine* (ELM)". Sebuah sistem yang dapat membaca pola tanda tangan seseorang menggunakan pengolahan citra digital dan *Extreme Learning Machine* (ELM). Penelitian ini lebih fokus pada proses pengolahan citra digital dan *Extreme Learning Machine* menggunakan *software* Atom yang berbahasa program *Python*. Langkah-langkah yang digunakan untuk menganalisa permasalahan tersebut dan selain itu perlu juga diamati mengenai hambatan-hambatan yang terjadi sehingga dapat meningkatkan kinerja alat tersebut.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut maka dapat dirumuskan masalah sebagai berikut:

1. Bagaimana menganalisa pembacaan tanda tangan pada sistem berbasis *Extreme Learning Machine* (ELM)?
2. Bagaimana kecepatan traning sistem terhadap ketepatan/akurasi pembacaan tanda tangan pada sistem berbasis *Extreme Learning Machine* (ELM)?

1.3 Batasan Masalah

Untuk memperjelas dan menghindari meluasnya masalah, maka batasan masalahnya yaitu:

1. Metode yang digunakan yaitu *Extreme Learning Machine* (ELM).
2. Alas / dasar yang digunakan yaitu kertas berwarna putih.
3. Bolpoin yang digunakan yaitu berwarna hitam.
4. Aplikasi yang digunakan Atom dan menggunakan bahasa program *Python*.

1.4 Tujuan Penelitian

Adapun tujuan pada penelitian ini adalah sebagai berikut:

1. Mengetahui proses pembacaan tanda tangan pada sistem berbasis *Extreme Learning Machine* (ELM).
2. Menganalisa pembacaan tanda tangan pada sistem berbasis *Extreme Learning Machine* (ELM).

1.5 Manfaat Penelitian

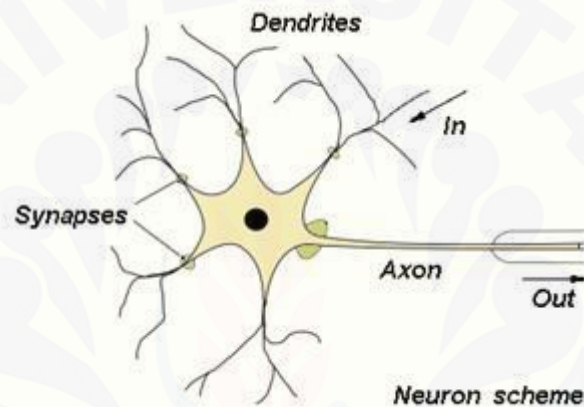
Adapun tujuan yang akan diperoleh dari penelitian ini adalah sebagai berikut :

1. Mempercepat dalam proses pengujian keaslian tanda tangan.
2. Membantu seseorang menganalisa terhadap keaslian tanda tangan.

BAB 2. TINJAUAN PUSTAKA

2.1 *Neural Network*

Neural Network merupakan kategori ilmu *Soft Computing*. *Neural Network* sebenarnya mengadopsi dari kemampuan otak manusia yang mampu memberikan stimulasi/rangsangan, melakukan proses, dan memberikan output. Output diperoleh dari variasi stimulasi dan proses yang terjadi di dalam otak manusia. Kemampuan manusia dalam memproses informasi merupakan hasil kompleksitas proses di dalam otak.



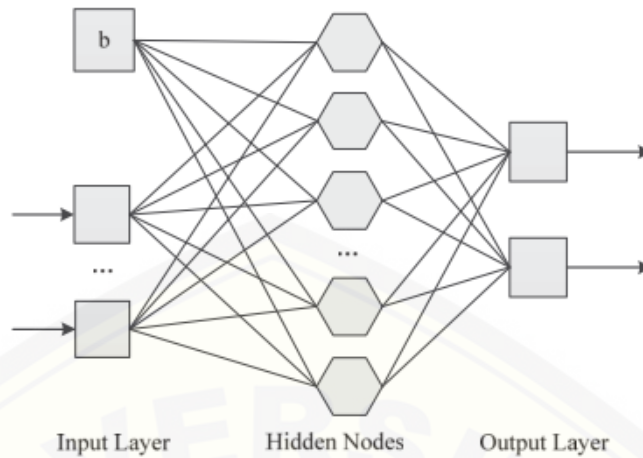
Gambar 2.1 Struktur Neuron pada otak manusia

Sebuah neuron menerima impuls dari neuron lain melalui dendrit dan mengirimkan sinyal yang dihasilkan oleh badan sel melalui akson. Akson dari sel syaraf ini bercabang-cabang dan berhubungan dengan dendrit dari sel syaraf lain dengan cara mengirimkan impuls melalui sinapsis. Sinapsis adalah unit fungsional antara 2 buah sel syaraf, misal A dan B, dimana yang satu adalah serabut akson dari neuron A dan satunya lagi adalah dendrit dari neuron B. Kekuatan sinapsis bisa menurun/meningkat tergantung seberapa besar tingkat propagasi (penyiaran) sinyal yang diterimanya. Impuls-impuls sinyal (informasi) akan diterima oleh neuron lain jika memenuhi batasan tertentu, yang sering disebut dengan nilai ambang (*threshold*).

2.2 *Extreme Learning Machine (ELM)*

Extreme Learning Machine merupakan pengembangan dari sistem kecerdasan buatan *artificial neural network* (ANN) yang diperkenalkan oleh Huang pada tahun 2004. (Huang, Zhu and Siew, 2004). Pada dasarnya *Extreme Learning Machines* adalah *Single Hidden Layer Feedforward Neural Networks* (SLFNs) yang memiliki kelebihan dibandingkan dengan jaringan syaraf tiruan biasa dengan model pembelajaran *backpropagation* (BP-ANN) (Cambria *et al.*, 2013), terutama dalam hal *learning speed*. Disamping itu, pada metode ELM ini, memiliki *error rate* yang lebih minim apabila dibandingkan dengan metode BP-ANN apabila dihitung berdasarkan perhitungan error menggunakan parameter statistik yaitu nilai MSE (*Mean Square Error*) dan MAPE (*Mean Absolute Percentage Error*) (Huang, Zhu and Siew, 2004).

Pada *training process* ANN metode *conventional gradient based learning algorithm* seperti pada jaringan syaraf tiruan menggunakan *learning process backpropagation* (BP -ANN) dan pada *learning process* Lavenberg – Marquadt (LM), parameter JST (jaringan syaraf tiruan) *feedforward* –nya haruslah ditentukan secara manual. Parameter yang dimaksud adalah parameter *input weight* dan *hidden bias*. Dikarenakan pada jaringan JST parameter – parameter saling berhubungan, maka semakin banyak *hidden bias* dan *layer* yang dibangun, maka *learning speed* yang dibutuhkan akan membutuhkan waktu yang lama dan sering terjebak pada *local minima*. (Huang, Zhu and Siew, 2004). Sedangkan pada *Extreme Learning Machine*, parameter – parameter tersebut dipilih secara *random* pada awal pelatihan, sehingga ELM cenderung memiliki kecepatan pembelajaran / *learning speed* yang cepat dan memiliki performa yang baik (*good generalization performance*). (Huang, Zhu and Siew, 2004).



Gambar 2.2. Struktur *Extreme Learning Machine*

(Sumber : Tan, Sa, and Yu 2016)

Metode ELM mempunyai model matematis yang berbeda dari jaringan syaraf tiruan *feedforward*. Model matematis dari ELM lebih sederhana dan efektif. Berikut model matematis dari ELM :

Pertimbangkan satu set n sampel (x_i, y_i) , dengan $1 \leq i \leq n$ ()

$x_i \in \mathbb{R}^p$ dan $y_i \in \mathbb{R}$, SFLN dengan m sebagai *hidden neurons* didalam *hidden layer* yang dapat diekpresikan dengan persamaan berikut :

$$\sum_{j=1}^m \beta_j f(\mathbf{w}_j \mathbf{x}_i + \mathbf{b}_j), 1 \leq i \leq n \dots\dots\dots (2.19)$$

Dengan β_j adalah keluaran beban, f adalah fungsi aktivasi, \mathbf{w}_j adalah *input* beban dan \mathbf{b}_j adalah bias. Menunjukkan \hat{y}_i dengan *output* diestimasi oleh SFLN, dalam kasus hipotetis di mana SLFN dengan sempurna mendekati keluaran aktual y_i , hubungannya adalah:

$$\sum_{j=1}^m \beta_j f(\mathbf{w}_j \mathbf{x}_i + \mathbf{b}_j) = y_i, 1 \leq i \leq n \dots\dots\dots (2.20)$$

yang ditulis dalam bentuk matriks sebagai $\mathbf{H}\boldsymbol{\beta} = \mathbf{y}$, dengan :

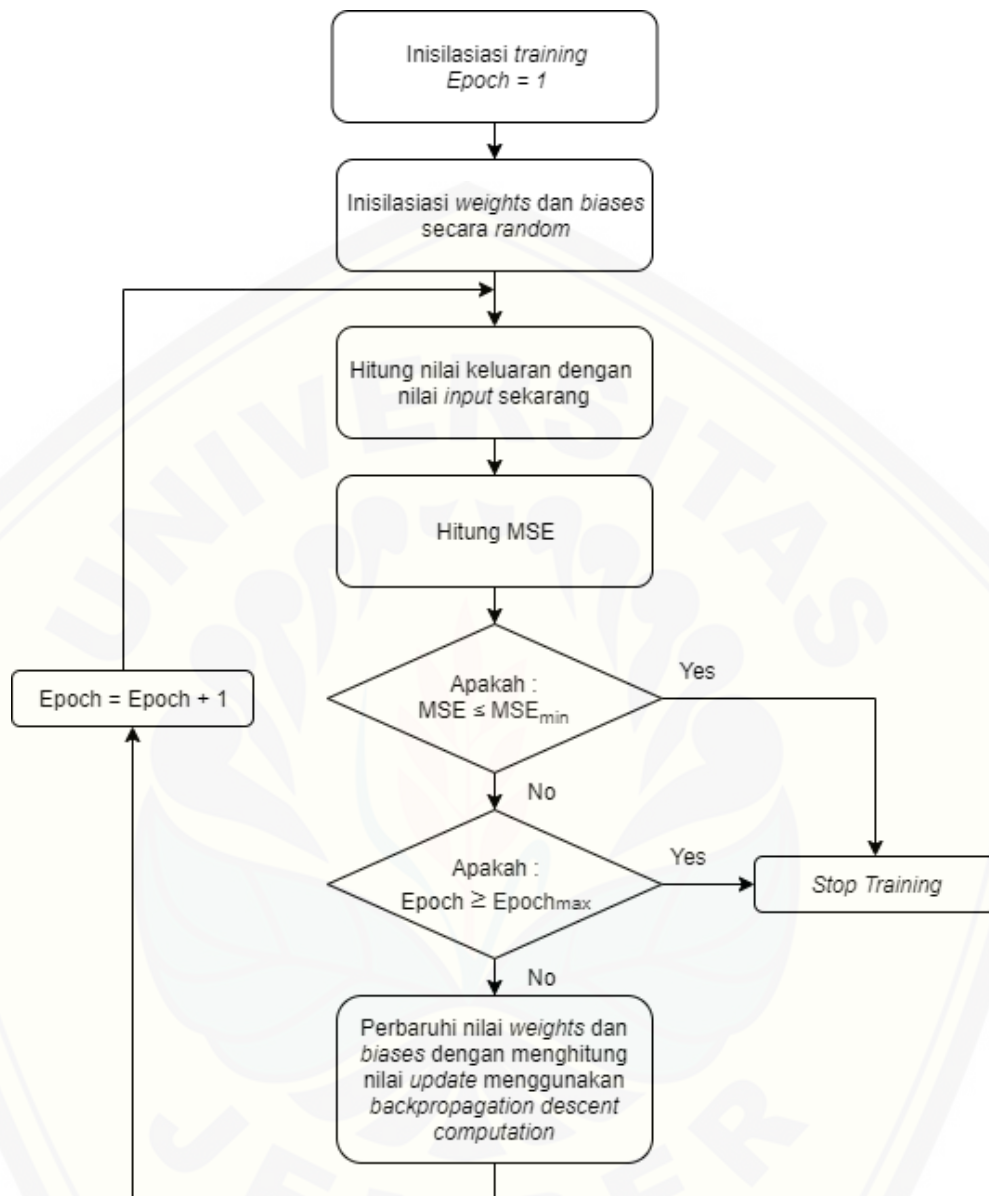
$$\mathbf{H} = \begin{pmatrix} f(\mathbf{w}_1 \mathbf{x}_1 + \mathbf{b}_1) & L & f(\mathbf{w}_m \mathbf{x}_1 + \mathbf{b}_m) \\ M & O & M \\ f(\mathbf{w}_1 \mathbf{x}_n + \mathbf{b}_1) & L & f(\mathbf{w}_m \mathbf{x}_n + \mathbf{b}_m) \end{pmatrix} \dots\dots\dots (2.21)$$

$$\beta = (\beta_1, \beta_2, K, \beta_j)^T \text{ dan } \mathbf{y} = (y_1, y_2, K, y_i)^T \dots\dots\dots (2.22)$$

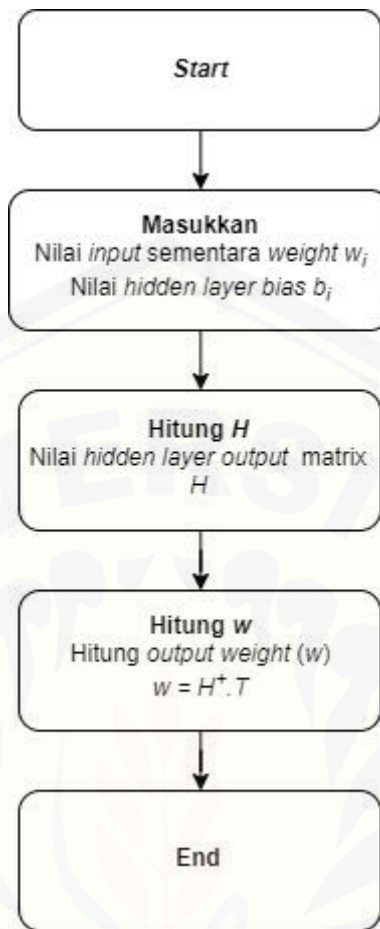
Dengan demikian pendekatan ELM adalah untuk menginisialisasi secara acak \mathbf{w}_j dan \mathbf{b}_j , dan menghitung bobot keluaran $\beta = \mathbf{H}^\dagger \mathbf{y}$ oleh Moore-j Penrose pseudo-invers. Dengan demikian, metode pembelajaran sederhana untuk SLFN yang disebut *Extreme Learning Machine* (ELM) dapat diringkas sebagai berikut:

- Langkah 1) Tentukan fungsi aktif simpul tersembunyi f dan simpul tersembunyi m .
- Langkah 2) Secara acak menetapkan bobot input dan bias \mathbf{b} .
- Langkah 3) Hitung matriks keluaran lapisan tersembunyi \mathbf{H} .
- Langkah 4) Hitung bobot *output* β .

Parameter yang dipelajari termasuk w_i , b , \mathbf{H} , β dan f , m adalah digunakan sebagai model untuk mengklasifikasikan dataset uji (Tan, Sa and Yu, 2016). Apabila dilihat dari segi algoritma yang digunakan antara *artificial neural network* (ANN) dengan *extreme learning machine* (ELM), maka akan terlihat bahwa ELM memiliki algoritma yang singkat dibandingkan dengan ANN. Perbedaan terletak pada algoritma yang digunakan untuk mencari nilai beban atau *weight* dan bias terbaik pada masing – masing neuron pada layer. Pada algoritma ANN, untuk dapat mencapai hasil yang terbaik, maka digunakanlah algoritma *backpropagation*, dimana *error* yang dihasilkan oleh keluaran *neuron output* akan digunakan untuk memperbarui nilai beban dan bias dengan menggunakan *gradient computation backpropagation* sampai nilai *error* yang dihasilkan sesuai dengan yang diharapkan oleh pengguna. Hal ini memungkinkan iterasi atau perulangan algoritma lebih dari 100 kali untuk dapat mencapai hasil yang diinginkan. Semakin banyak iterasi atau *epoch* yang dibutuhkan dalam sekali proses *training*, maka semakin lama waktu yang dibutuhkan. Sedangkan pada ELM hal tersebut tidak dilakukan. Untuk dapat mempercepat proses penentuan beban atau *weight* dan bias pada jaringan, maka ELM hanya perlu melakukan perhitungan dengan menggunakan pseudo invers. Dengan melakukan hal tersebut, maka nilai *neuro* terbaik akan mudah didapatkan dalam waktu yang sangat singkat. Untuk lebih jelasnya hal tersebut dapat dilihat pada diagram alir pada Gambar 2.13 untuk ANN dan Gambar 2.14 untuk ELM.

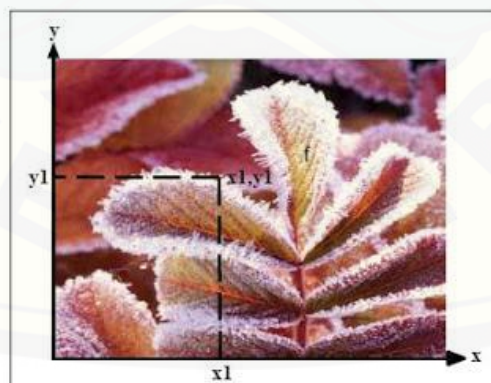


Gambar 2.3. Diagram Alir *Backpropagation* (ANN)



Gambar 2.4. Diagram Alir *Extreme Learning Machine* (ELM)

2.3 Citra Digital



Gambar 2.5 Koordinat citra digital

(sumber: <http://www.temukanpengertian.com>)

Citra digital adalah gambar dua dimensi yang bisa ditampilkan pada layar komputer sebagai himpunan/ diskrit nilai digital yang disebut pixel/ picture

elements. Dalam tinjauan matematis, citra merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi.

Citra digital adalah citra $f(x,y)$ dimana dilakukan diskritisasi koordinat sampling/ spasial dan diskritisasi tingkat kwantisasi (keabuan/ kecermerlangannya). Citra digital merupakan fungsi intensitas cahaya $f(x,y)$, dimana harga x dan harga y adalah koordinat spasial. Harga fungsi tersebut di setiap titik (x,y) merupakan tingkat kecermerlangan citra pada titik tersebut.

Citra digital merupakan suatu matriks dimana indeks baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matriksnya (yang disebut sebagai elemen gambar/ pixel/ piksel/ pels/ picture element) menyatakan tingkat keabuan pada titik tersebut. Matrik yang dinyatakan Citra digital yaitu dengan matriks berukuran N (baris/tinggi) x M (kolom/lebar).

N = jumlah baris $0 = y = N - 1$.

M = jumlah kolom $0 = x = M - 1$.

L = maksimal warna intensitas $0 = f(x,y) = L - 1$.

(gray level/ derajat keabuan)

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

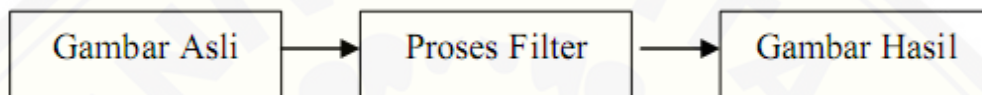
Gambar 2.6 Matrik Citra Digital

(sumber: <http://www.temukanpengertian.com>)

2.3.1 Pengolahan Citra Warna

Image processing atau sering disebut dengan pengolahan citra digital merupakan suatu metode yang digunakan untuk mengolah atau memproses dari gambar asli sehingga menghasilkan gambar lain yang sesuai dengan kebutuhan. Pengambilan gambar bisa dilakukan oleh kamera video atau alat-alat yang lain yang dapat digunakan untuk mentransfer gambar. Dalam pengolahan citra, dilakukan operasi terhadap citra asli menjadi citra baru berdasarkan citra asli. Operasi yang dilakukan pada citra dikategorikan sebagai berikut :

1. *Point*, yaitu operasi yang menghasilkan output dimana setiap *pixel* hanya dipengaruhi oleh *pixel* pada posisi yang sama dari citra asli.
2. *Local*, yaitu operasi yang menghasilkan output dimana *pixel*nya dipengaruhi oleh *pixel-pixel* tetangga pada citra asli.
3. *Global*, yaitu operasi yang menghasilkan output dimana *pixel*nya dipengaruhi oleh semua *pixel* yang ada dalam citra asli. Misalnya ada suatu gambar yang terlalu gelap maka dengan *image processing* gambar tersebut bisa diproses sehingga mendapatkan gambar yang jelas. Secara garis besar bisa digambarkan seperti blok diagram pada gambar dibawah ini:

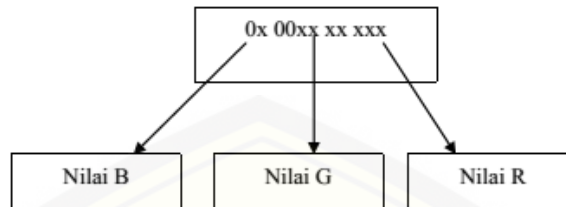


Gambar 2.7 Blok diagram pengolahan citra

Citra dapat berbentuk foto hitam putih atau berwarna, sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu pita magnetik. Menurut presisi yang digunakan untuk menyatakan titik-titik koordinat pada ranah waktu atau bidang dan untuk menyatakan nilai keabuan atau warna suatu citra, maka secara teoritis citra dapat dikelompokkan menjadi empat kelas citra, yaitu cara kontinu-kontinu, kontinu-diskret, diskret-kontinu, dan diskret-diskret; dengan label pertama menyatakan presisi dari titik-titik koordinat pada bidang citra sedangkan label kedua menyatakan presisi nilai keabuan atau warna. Kontinu dinyatakan dengan presisi takhingga, sedangkan diskret dinyatakan dengan presisi angka berhingga. Perubahan citra yang bersifat kontinu menjadi citra yang bersifat diskret memerlukan pembuatan kisi-kisi arah vertikal dan horisontal, sehingga diperoleh citra dalam bentuk larik dua dimensi. Proses tersebut dikenal sebagai proses digitalisasi atau pencuplikan (*sampling*). Setiap elemen larik tersebut dikenal sebagai elemen gambar atau piksel.

Dasar dari pengolahan citra adalah pengolahan warna RGB pada posisi tertentu. Dalam pengolahan citra warna dipresentasikan dengan nilai hexadesimal dari 0x00000000 sampai 0x00ffffff. Warna hitam adalah 0x00000000 dan warna putih

adalah 0x00ffffff. Definisi nilai warna di atas seperti gambar 2.2, variabel 0x00 menyatakan angka dibelakangnya adalah hexadesimal.



Gambar 2.8 Nilai warna RGB dalam Hexadesimal

(Sumber : Istri Sulistyowati; 2010).

Terlihat bahwa setiap warna mempunyai *range* nilai 00 (angka desimalnya adalah 0) dan ff (angka desimalnya adalah 255), atau mempunyai nilai derajat keabuan $256 = 2^8$. Dengan demikian range warna yang digunakan adalah $(2^8)(2^8)(2^8) = 2^{24}$ (atau yang dikenal dengan istilah *True Colour* pada Windows). Nilai warna yang digunakan di atas merupakan gabungan warna cahaya merah, hijau dan biru seperti yang terlihat pada gambar 2.3. Sehingga untuk menentukan nilai dari suatu warna yang bukan warna dasar digunakan gabungan skala kecerahan dari setiap warnanya. (Sulistyowati, 2010:3).



Gambar 2.9 Komposisi warna RGB

(Sumber : Istri Sulistyowati; 2010)

2.3.2 Citra *Gray-scale*(Keabuan)

Proses awal yang banyak dilakukan dalam *image processing* adalah mengubah citra berwarna menjadi citra abu-abu, hal ini digunakan untuk menyederhanakan model citra. Seperti dijelaskan didepan, citra berwarna terdiri 3 layer matrik yaitu R-layer, G-layer, B-layer (Distira, 2012:21). Sehingga untuk melakukan proses selanjutnya tetap diperhatikan 3 layer diatas. Bila setiap proses perhitungan dilakukan menggunakan tiga layer, berarti dilakukan tiga perhitungan yang sama. Sehingga konsep itu diubah dengan mengubah 3 layer di atas menjadi

1 layer matrik gray-scale dan hasilnya adalah citra gray-scale. Dalam citra ini tidak ada lagi warna, yang ada adalah derajat keabuan. (Distira, 2012:22)



Gambar 2.10 Contoh konversi citra RGB ke citra abu-abu (Sumber :Reda Anggra Distira ; 2012)

Dalam konversi citra RGB ke model abu-abu, diperlukan mengubah komposisi nilai matrik RGB. Hal tersebut didapatkan dengan mencari nilai rata-rata dari r, g dan b menjadi nilai abu-abu dengan nilai s. Berikut rumus untuk konversi citra RGB ke model abu-abu.

$$S = \frac{R+G+B}{3} \dots\dots\dots (2.1)$$

2.3.3 Binerisasi Citra

Salah satu teknik awal dalam pra proses pengolahan citra digital, salah satunya ialah konversi citra gray ke citra biner (binerisasi). Citra biner (hitam-putih) diperlukan dalam pra proses pengolahan citra untuk keperluan pengenalan objek seperti simbol, angka atau huruf. Proses konversi citra *gray-scale* ke citra biner, Dapat dituliskan dalam rumus :

Jika $x < 128$ maka $x=0$, jika tidak maka $x=255$.
(2.2)

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

Gambar 2.11 Citra Biner

2.3.4 Thresholding

Thresholding merupakan konversi citra hitam – putih ke citra biner dilakukan dengan cara mengelompokkan nilai derajat keabuan setiap pixel kedalam 2 kelas, hitam dan putih. Pada citra hitam putih terdapat 256 level, artinya mempunyai skala “0” sampai “255” atau [0,255], dalam hal ini nilai intensitas 0 menyatakan hitam, dan nilai intensitas 255 menyatakan putih, dan nilai antara 0 sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih. Secara umum proses pengambangan citra *gray-scale* untuk menghasilkan citra biner adalah sebagai berikut.

$$G(x,y) = \begin{cases} 1 & \text{iff } (x,y) \geq T \\ 0 & \text{iff } (x,y) < T \end{cases} \dots\dots\dots (2.3)$$

Dimana :

$G(x,y)$ = citra biner

$f(x,y)$ = citra *grayscale*

T = nilai ambang

Nilai T memegang peranan yang sangat penting dalam proses pengambangan. Kualitas hasil citra biner sangat tergantung pada nilai T yang digunakan.

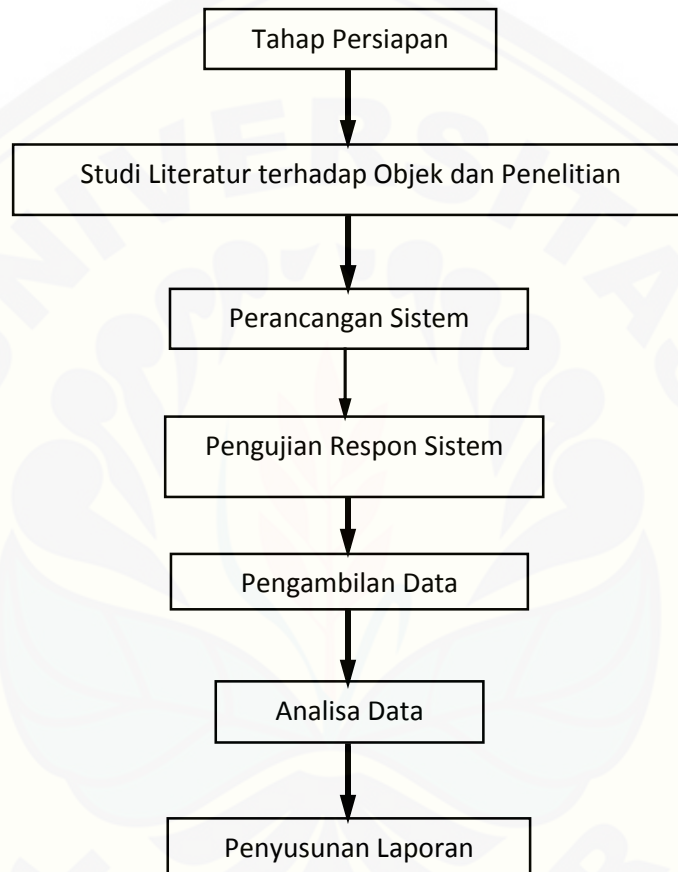


Gambar 2.12 Contoh Derajat Kualitas *Tresholding* (Sumber :M. Irwan Bustamiet all; 2013)

BAB 3. METODELOGI PENELITIAN

3.1 Tahapan Penelitian

Dalam memecahkan masalah pada penelitian yang diamati dibutuhkan langkah-langkah untuk menguraikan pendekatan dan model dari masalah tersebut. Langkah-langkah yang dilakukan adalah sebagai berikut:



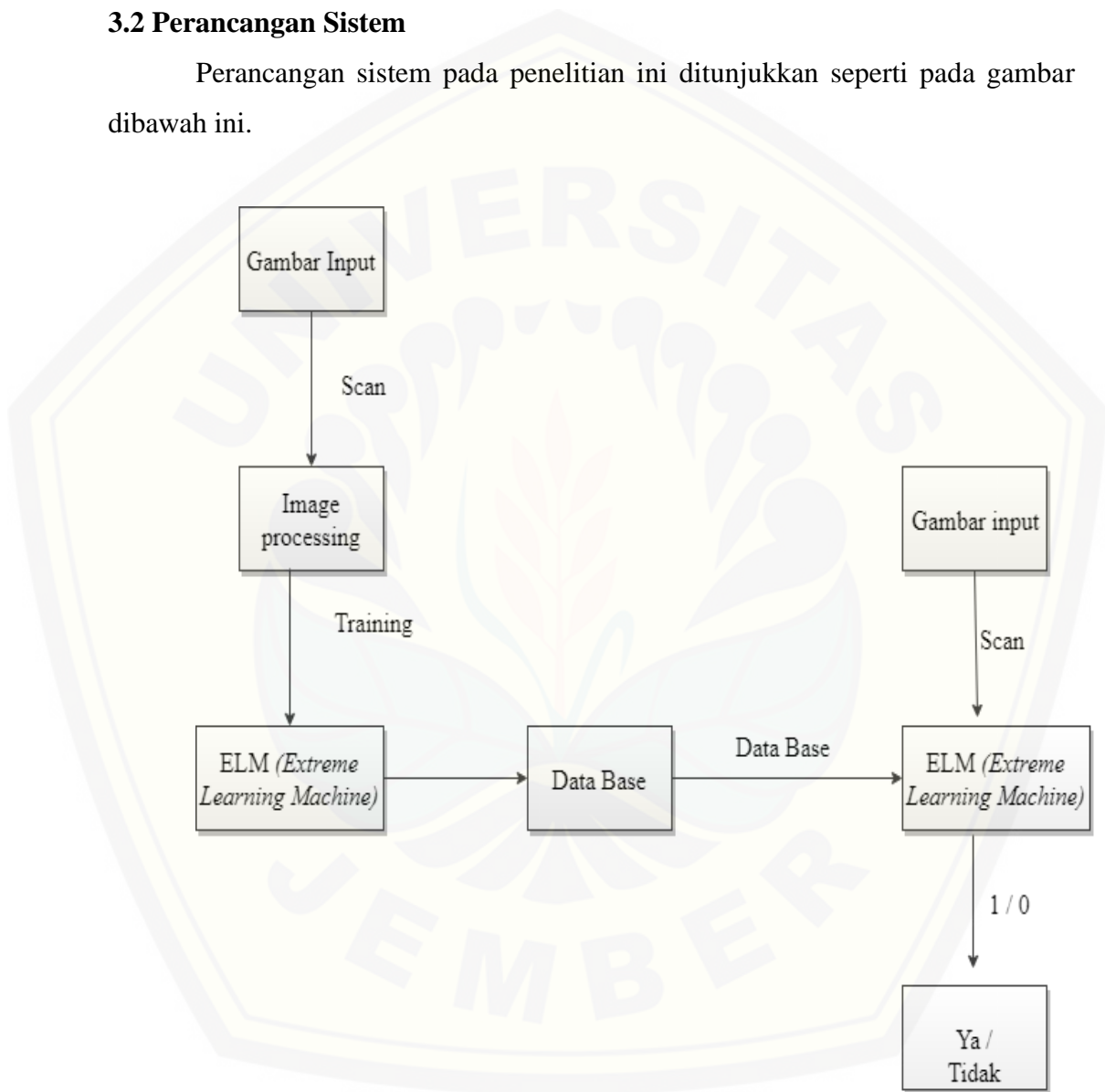
Gambar 3.1 Blok Diagram Tahap Penelitian

Pada tahap penelitian ini meliputi beberapa tahap mulai dari persiapan alat dan bahan-bahan yang nantinya akan disusun sesuai rancangan. Selanjutnya studi literatur terhadap objek dan penelitian yaitu pada penelitian ini tentang pengujian pembacaan tanda tangan berbasis *Extreme Learning Machine*(ELM) dengan mencari literatur dari buku maupun internet, serta dari penelitian-penelitian sebelumnya. Setelah mempelajari cara kerja dan fungsi semua alat dan sistem yang digunakan, langkah selanjutnya membangun system perancangan penelitian. Proses selanjutnya adalah pengujian sistem sampai benar-benar sesuai. Kemudian

dilakukan pengambilan data yang telah selesai dirancang. Pengolahan data dilakukan setelah melakukan pengambilan data yang kemudian disusun menjadi sebuah laporan.

3.2 Perancangan Sistem

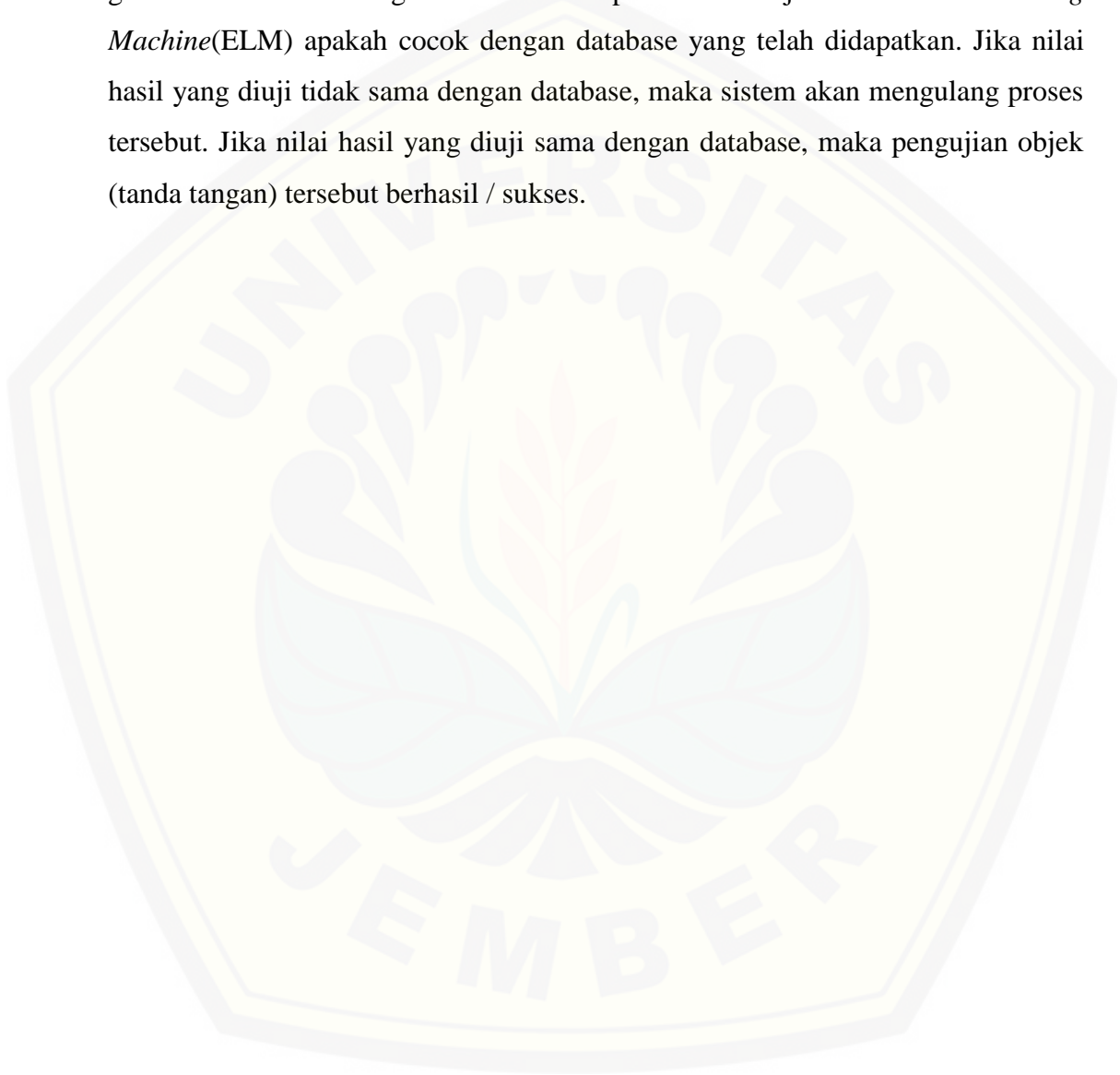
Perancangan sistem pada penelitian ini ditunjukkan seperti pada gambar dibawah ini.



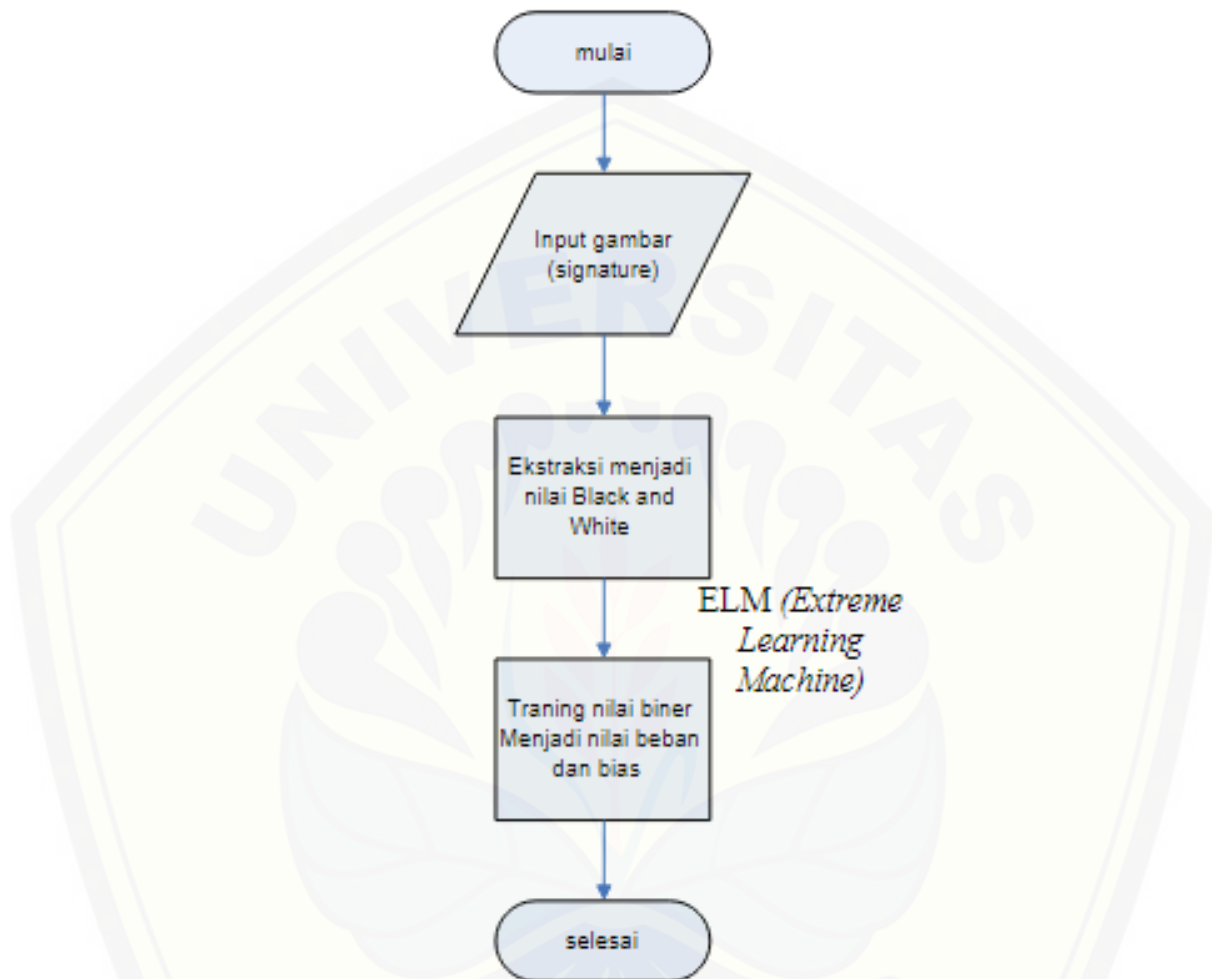
Gambar 3.2 Blok Diagram Proses Pengambilan Data Base dan Pengujian Objek

Objek yang akan digunakan sebagai data base discan agar dapat menghasilkan file yang nantinya akan di olah di aplikasi Atom. Kemudian file tersebut di ekstrak menjadi nilai Black and white menggunakan teknik *image*

processing. Setelah itu hasil dari *image processing* tersebut ditraining oleh *Extreme Learning Machine*(ELM) menjadi nilai bias dan biner yang nantinya digunakan sebagai data base pengujian objek. Setelah tahap pengambilan database, gambar yang akan diuji kecocokannya berdasarkan database discan untuk menghasilkan file gambar. Kemudian file gambar tersebut diproses dan diuji oleh *Extreme Learning Machine*(ELM) apakah cocok dengan database yang telah didapatkan. Jika nilai hasil yang diuji tidak sama dengan database, maka sistem akan mengulang proses tersebut. Jika nilai hasil yang diuji sama dengan database, maka pengujian objek (tanda tangan) tersebut berhasil / sukses.

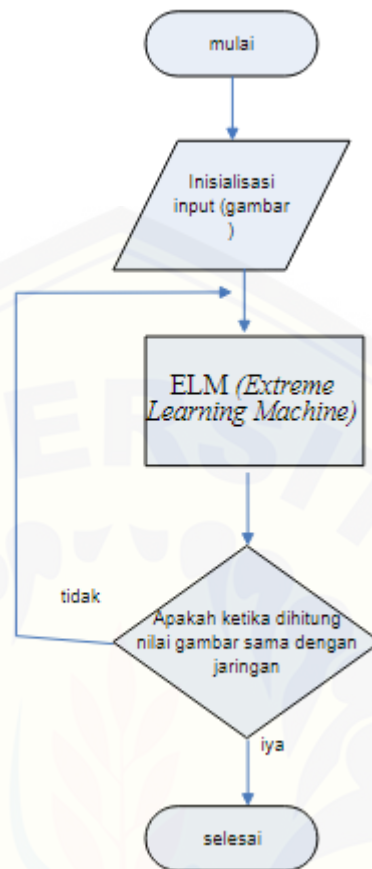


3.3 Flow Chart



Gambar 3.3 Flowchart Proses pengambilan Data Base

Berdasarkan gambar 3.3 yaitu flowchart proses pengambilan data base. Objek yang akan digunakan sebagai data base discan agar dapat menghasilkan file yang nantinya akan di olah di aplikasi Atom. Kemudian file tersebut di ekstrak menjadi nilai Black and white menggunakan teknik *image processing*. Setelah itu hasil dari *image processing* tersebut ditraining oleh *Extreme Learning Machine*(ELM) menjadi nilai bias dan biner yang nantinya digunakan sebagai data base pengujian objek.



Gambar 3.4 Flowchart Proses Pengujian Objek

Berdasarkan gambar 3.4 yaitu flowchart proses pengujian objek (tanda tangan). Setelah tahap pengambilan database, gambar yang akan diuji kecocokannya berdasarkan database discan untuk menghasilkan file gambar. Kemudian file gambar tersebut diproses dan diuji oleh *Extreme Learning Machine*(ELM) apakah cocok dengan database yang telah didapatkan. Jika nilai hasil yang diuji tidak sama dengan database, maka sistem akan mengulang proses tersebut. Jika nilai hasil yang diuji sama dengan database, maka pengujian objek (tanda tangan) tersebut berhasil / sukses.

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan yang telah dilakukan maka dapat disimpulkan bahwa :

1. Program sistem ELM (*Extreme Learning Machine*) yang digunakan untuk pengujian tingkat klarifikasi tanda tangan asli atau tidak, telah berhasil. Nilai akurasi yang dihasilkan pada penelitian ini adalah 0,758850516.
2. Berdasarkan hasil pengujian pada penelitian ini, dapat dikatakan perubahan jumlah neuron atau perubahan struktur ELM (*Extreme Learning Machine*) dapat berpengaruh terhadap perubahan nilai akurasi database yang digunakan. Hal itu terbukti dengan terjadinya kenaikan dan penurunan nilai akurasi secara signifikan pada setiap perubahan rentang neuron (pada beberapa data tanda tangan subjek yang diuji). Akan tetapi ada beberapa data tanda tangan yang cenderung stabil nilai akurasinya, yaitu pada data subjek 03, subjek 05, dan subjek 08.

5.2 Saran

Beberapa saran yang dapat dipertimbangkan untuk pengembangan sistem pembacaan *signature* (tanda tangan) selanjutnya yaitu:

1. Untuk melakukan tuning neuron agar akurasi yang dihasilkan dapat lebih maksimal. Salah satu contoh bisa menggunakan algoritma optimasi seperti *Particle Swarm Optimisation* (PSO) atau yang lain.
2. Memperbanyak jumlah database subjek agar akurasi rata-rata dapat semakin terlihat dan akurat. Jumlah database yang dimaksud adalah jumlah data *genuine* dan *forged*.

DAFTAR PUSTAKA

- CHAVEZ-HURTADO, J.L. & CORTESFREGOSO, J.H.,2013. Forecasting Mexican inflation using neural networks. 23rd International Conference on Electronics, Communications and Computing, CONIELECOMP 2013, pp.32–35.
- HANDIKA, I.P.S., GIRIANTARI, I.A. & DHARMA, A., 2016. Perbandingan Metode Extreme Learning Machine dan Particle Swarm Optimization Extreme Learning Machine untuk Peramalan Jumlah Penjualan Barang. *Teknologi Elektro*, 15(1), pp.84–90.
- HIDAYAT, R. & SUPRAPTO, 2012. Meminimalisasi Nilai Error Peramalan dengan Algoritma Extreme Learning Machine (ELM). *Optimasi Sistem Industri*, 11(1), pp.187–192.
- HILL, T., CONNOR, M.O. & REMUS, W. 1996. Neural Network Models for Time Series Forecasts. *Management Science*, 42(7), pp.1082–1092.
- HUANG, G. ET AL., 2015. Trends in extreme learning machines: A review. *Neural Networks*, 61, pp.32–48. Available at: <http://dx.doi.org/10.1016/j.neunet.2014.10.001>
- HUANG, G. BIN, ZHU, Q.Y. & SIEW, C.K., 2006. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1–3), pp.489–501.
- SUGIANTO, N.A., CHOLISSODIN, I. & WIDODO, A.W., 2018. Klasifikasi Keminatan Menggunakan Algoritme Extreme Learning Machine dan Particle Swarm Optimization untuk Seleksi Fitur (Studi Kasus : Program Studi Teknik Informatika FILKOM UB). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2(5), pp.1856–1865.

LAMPIRAN

Data Gambar *Signature*





Penul

Penul

Penul

Penul

Penul

Penul

Penul

Penul

Penul

Penul

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]







Say

Say

Say

Say

Say

Say

Say

Say

Say

Say





Program Pengujian *ELM* (*extreme learning machine*)

```
import cv2
import numpy as np
import pandas as pd
import os

def prepare(input):
    #gray_img = cv2.cvtColor(input, cv2.COLOR_BGR2GRAY)
    ret, bw_img = cv2.threshold(input,200,255,cv2.THRESH_BINARY)

    flatten_img = cv2.resize(bw_img, (40, 10), interpolation=cv2.INTER_AREA).flatten()

    # resize to 400x100
    resized = cv2.resize(bw_img, (400, 100), interpolation=cv2.INTER_AREA)
    columns = np.sum(resized, axis=0) # sum of all columns
    lines = np.sum(resized, axis=1) # sum of all lines

    resized = resized/255
    r, c = resized.shape
    shape = r*c

    resized = np.ravel(resized)

    return(resized)

current_dir = os.path.dirname('E:/skripsweet/signature dhion/ELM/1FOLDER/')

author = '10'
```



```
training_folder = os.path.join(current_dir, 'training/', author)
test_folder = os.path.join(current_dir, 'test/', author)

training_data = []
training_labels = []
training_data = np.empty((len(os.listdir(training_folder)),int(40000)))
training_labels = np.empty((len(os.listdir(training_folder)),int(1)))

x = 0
for filename in os.listdir(training_folder):
    img = cv2.imread(os.path.join(training_folder, filename), 0)
    training_data[x] = prepare(img)
    if "genuine" in filename : training_labels[x] = 1
    else: training_labels[x] = -1
    x+=1

training_data.shape
training_labels.shape

from elm import ELM
from sklearn.preprocessing import normalize
from sklearn.datasets import fetch_openml
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(training_data, training_labels,
test_size=0.6)

X_train.shape
X_test.shape

# elm = ELM(hid_num=500).fit(X_train, y_train)
```

```
# print("ELM Accuracy %0.3f " % elm.score(X_test, y_test))

min_neuron = 100
max_neuron = 3000
interval = 100
accuracy_data = np.empty([int((max_neuron - min_neuron)/interval),2])

for x in range(int((max_neuron-min_neuron)/interval)):
    print(min_neuron)
    min_neuron = min_neuron + interval
    elm = ELM(hid_num=500).fit(X_train, y_train)
    accuracy_data[x] = min_neuron, elm.score(X_test, y_test)

pd.DataFrame(accuracy_data)
```

Hasil Pengujian

Nomor	Jumlah Neuron	Akurasi
0	200	0,733333
1	300	0,733333
2	400	0,733333
3	500	0,733333
4	600	0,733333
5	700	0,8
6	800	0,533333
7	900	0,733333
8	1000	0,733333
9	1100	0,733333
10	1200	0,733333
11	1300	0,733333
12	1400	0,733333
13	1500	0,8
14	1600	0,733333
15	1700	0,733333
16	1800	0,733333
17	1900	0,733333
18	2000	0,733333
19	2100	0,8
20	2200	0,733333
21	2300	0,733333
22	2400	0,733333
23	2500	0,733333
24	2600	0,733333
25	2700	0,733333
26	2800	0,733333
27	2900	0,733333
28	3000	0,733333
Rata-rata akurasi		0,733333034

Tabel Akurasi Subjek 01

Nomor	Jumlah Neuron	Akurasi
0	200	0,666667
1	300	0,8
2	400	0,733333
3	500	0,733333
4	600	0,733333
5	700	0,6
6	800	0,8
7	900	0,6
8	1000	0,733333
9	1100	0,8
10	1200	0,866667
11	1300	0,8
12	1400	0,733333
13	1500	0,666667
14	1600	0,533333
15	1700	0,733333
16	1800	0,733333
17	1900	0,666667
18	2000	0,866667
19	2100	0,8
20	2200	0,8
21	2300	0,6
22	2400	0,466667
23	2500	0,8
24	2600	0,8
25	2700	0,733333
26	2800	0,8
27	2900	0,8
28	3000	0,733333
Rata-rata akurasi		0,728735586

Tabel Akurasi Subjek 02

Nomor	Jumlah Neuron	Akurasi
0	200	0,733333
1	300	0,733333
2	400	0,733333
3	500	0,733333
4	600	0,733333
5	700	0,733333
6	800	0,733333
7	900	0,733333
8	1000	0,733333
9	1100	0,733333
10	1200	0,733333
11	1300	0,733333
12	1400	0,733333
13	1500	0,733333
14	1600	0,733333
15	1700	0,733333
16	1800	0,733333
17	1900	0,733333
18	2000	0,733333
19	2100	0,733333
20	2200	0,733333
21	2300	0,733333
22	2400	0,733333
23	2500	0,733333
24	2600	0,733333
25	2700	0,733333
26	2800	0,733333
28	3000	0,733333
Rata-rata akurasi		0,733333

Tabel Akurasi Subjek 03

Nomor	Jumlah Neuron	Akurasi
0	200	0,933333
1	300	0,8
2	400	0,666667
3	500	0,733333
4	600	0,8
5	700	0,733333
6	800	0,866667
7	900	0,866667
8	1000	0,733333
9	1100	0,8
10	1200	0,866667
11	1300	0,866667
12	1400	0,733333
13	1500	0,933333
14	1600	0,866667
15	1700	0,733333
16	1800	0,733333
17	1900	0,933333
18	2000	0,666667
19	2100	0,866667
20	2200	0,733333
21	2300	0,733333
22	2400	0,6
23	2500	0,866667
24	2600	0,8
25	2700	0,8
26	2800	0,733333
27	2900	0,8
28	3000	0,533333
Rata-rata akurasi		0,783908

Tabel Akurasi Subjek 04

Nomor	Jumlah Neuron	Akurasi
0	200	0,733333
1	300	0,733333
2	400	0,733333
3	500	0,733333
4	600	0,733333
5	700	0,733333
6	800	0,733333
7	900	0,733333
8	1000	0,733333
9	1100	0,733333
10	1200	0,733333
11	1300	0,733333
12	1400	0,733333
13	1500	0,733333
14	1600	0,733333
15	1700	0,733333
16	1800	0,733333
17	1900	0,733333
18	2000	0,733333
19	2100	0,733333
20	2200	0,733333
21	2300	0,733333
22	2400	0,733333
23	2500	0,733333
24	2600	0,733333
25	2700	0,733333
26	2800	0,733333
27	2900	0,733333
28	3000	0,733333
Rata-rata akurasi		0,733333

Tabel Akurasi Subjek 05

Nomor	Jumlah Neuron	Akurasi
0	200	0,8
1	300	0,8
2	400	0,8
3	500	0,8
4	600	0,8
5	700	0,8
6	800	0,666667
7	900	0,8
8	1000	0,8
9	1100	0,8
10	1200	0,8
11	1300	0,8
12	1400	0,8
13	1500	0,8
14	1600	0,8
15	1700	0,733333
16	1800	0,8
17	1900	0,8
18	2000	0,8
19	2100	0,8
20	2200	0,8
21	2300	0,8
22	2400	0,8
23	2500	0,8
24	2600	0,8
25	2700	0,8
26	2800	0,866667
27	2900	0,8
28	3000	0,8
Rata-rata akurasi		0,79540231

Tabel Akurasi Subjek 06

Nomor	Jumlah Neuron	Akurasi
0	200	0,8
1	300	0,8
2	400	0,866667
3	500	0,8
4	600	0,933333
5	700	0,8
6	800	0,8
7	900	0,733333
8	1000	0,733333
9	1100	0,8
10	1200	0,733333
11	1300	0,733333
12	1400	0,8
13	1500	0,733333
14	1600	0,6
15	1700	0,8
16	1800	0,866667
17	1900	0,8
18	2000	0,866667
19	2100	0,866667
20	2200	0,6
21	2300	0,866667
22	2400	0,866667
23	2500	0,866667
24	2600	0,866667
25	2700	0,666667
26	2800	0,8
27	2900	0,933333
28	3000	0,8
Rata-rata akurasi		0,797701172

Tabel Akurasi Subjek 07

Nomor	Jumlah Neuron	Akurasi
0	200	0,666667
1	300	0,666667
2	400	0,666667
3	500	0,666667
4	600	0,666667
5	700	0,666667
6	800	0,666667
7	900	0,666667
8	1000	0,666667
9	1100	0,666667
10	1200	0,666667
11	1300	0,666667
12	1400	0,666667
13	1500	0,666667
14	1600	0,666667
15	1700	0,666667
16	1800	0,666667
17	1900	0,666667
18	2000	0,666667
19	2100	0,666667
20	2200	0,666667
21	2300	0,666667
22	2400	0,666667
23	2500	0,666667
24	2600	0,666667
25	2700	0,666667
26	2800	0,666667
27	2900	0,666667
28	3000	0,666667
Rata-rata akurasi		0,666667

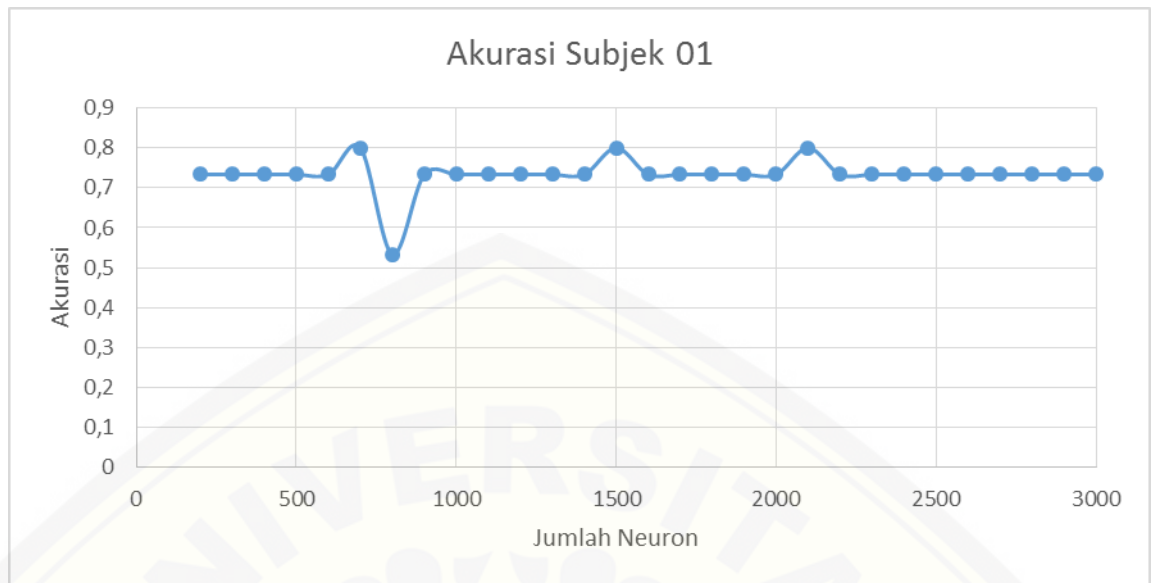
Tabel Akurasi Subjek 08

Nomor	Jumlah Neuron	Akurasi
0	200	0,8
1	300	0,8
2	400	0,866667
3	500	0,733333
4	600	0,8
5	700	0,8
6	800	0,8
7	900	0,8
8	1000	0,8
9	1100	0,8
10	1200	0,8
11	1300	0,666667
12	1400	0,8
13	1500	0,733333
14	1600	0,666667
15	1700	0,8
16	1800	0,866667
17	1900	0,8
18	2000	0,8
19	2100	0,733333
20	2200	0,8
21	2300	0,8
22	2400	0,8
23	2500	0,866667
24	2600	0,8
25	2700	0,8
26	2800	0,733333
27	2900	0,8
28	3000	0,8
Rata-rata akurasi		0,788505759

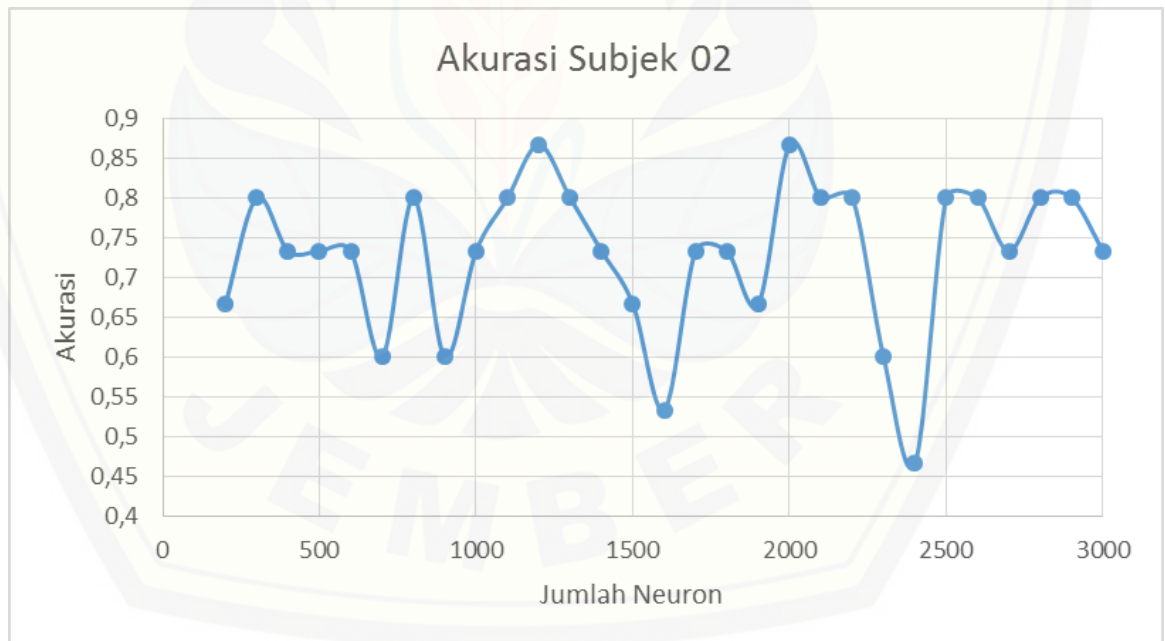
Tabel Akurasi Subjek 09

Nomor	Jumlah Neuron	Akurasi
0	200	0,933333
1	300	0,733333
2	400	0,866667
3	500	0,866667
4	600	0,866667
5	700	0,8
6	800	0,866667
7	900	0,866667
8	1000	0,8
9	1100	0,866667
10	1200	0,866667
11	1300	0,866667
12	1400	0,8
13	1500	0,8
14	1600	0,8
15	1700	0,733333
16	1800	0,8
17	1900	0,8
18	2000	0,8
19	2100	0,733333
20	2200	0,866667
21	2300	0,866667
22	2400	0,866667
23	2500	0,933333
24	2600	0,933333
25	2700	0,733333
26	2800	0,6
27	2900	0,866667
28	3000	0,866667
Rata-rata akurasi		0,827586276

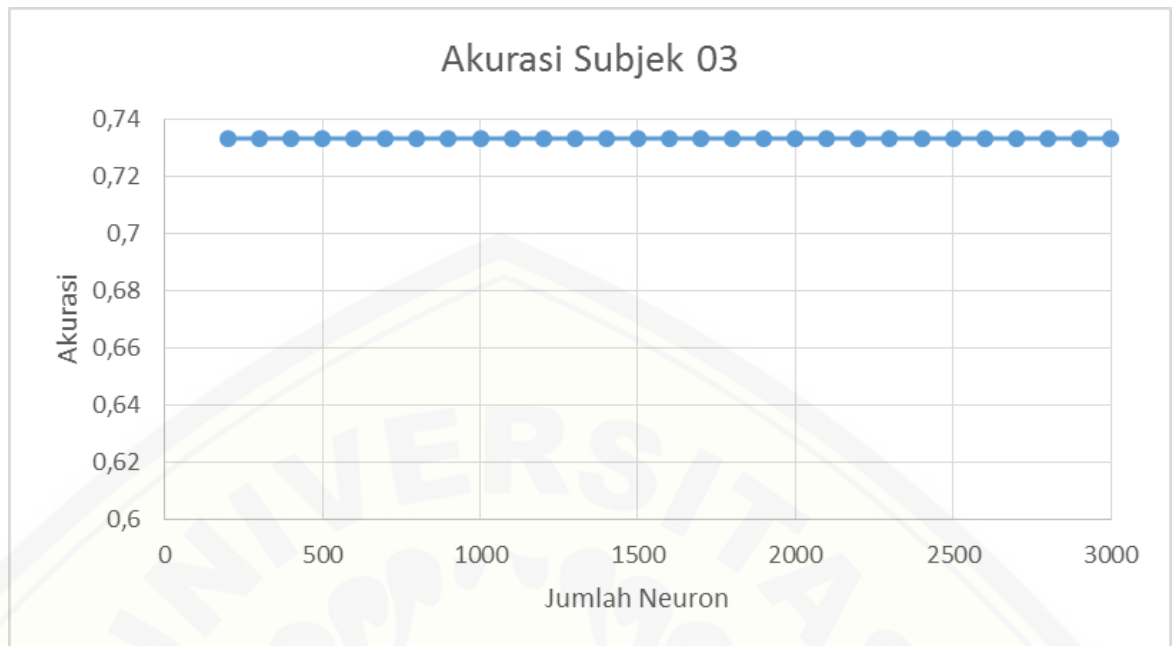
Tabel Akurasi Subjek 10



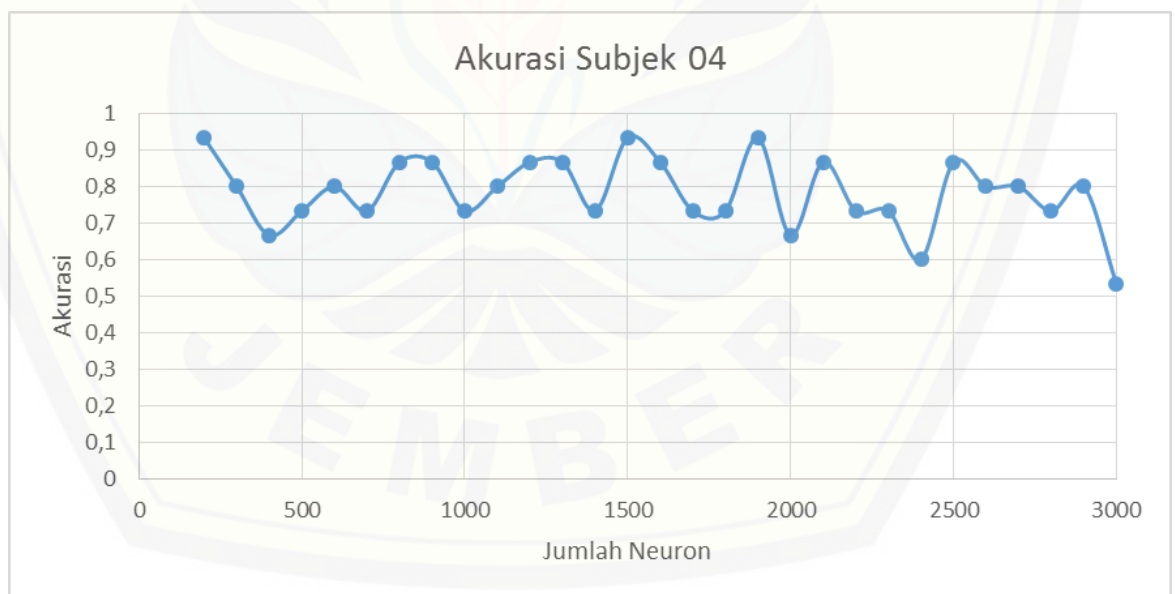
Grafik Akurasi Subjek 01



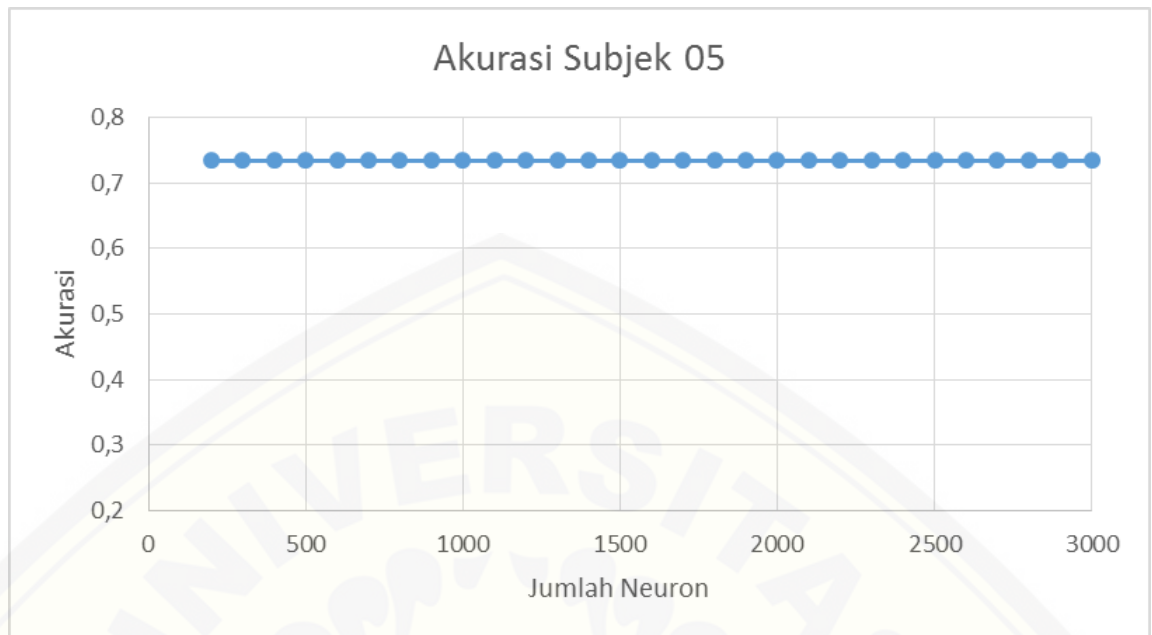
Grafik Akurasi Subjek 02



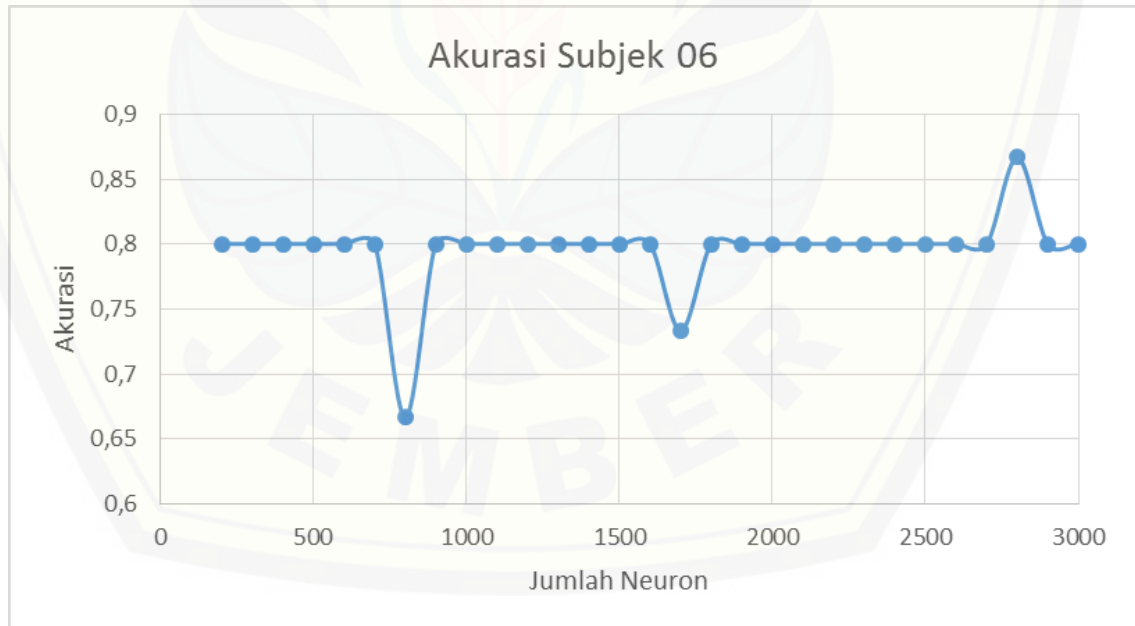
Grafik Akurasi Subjek 03



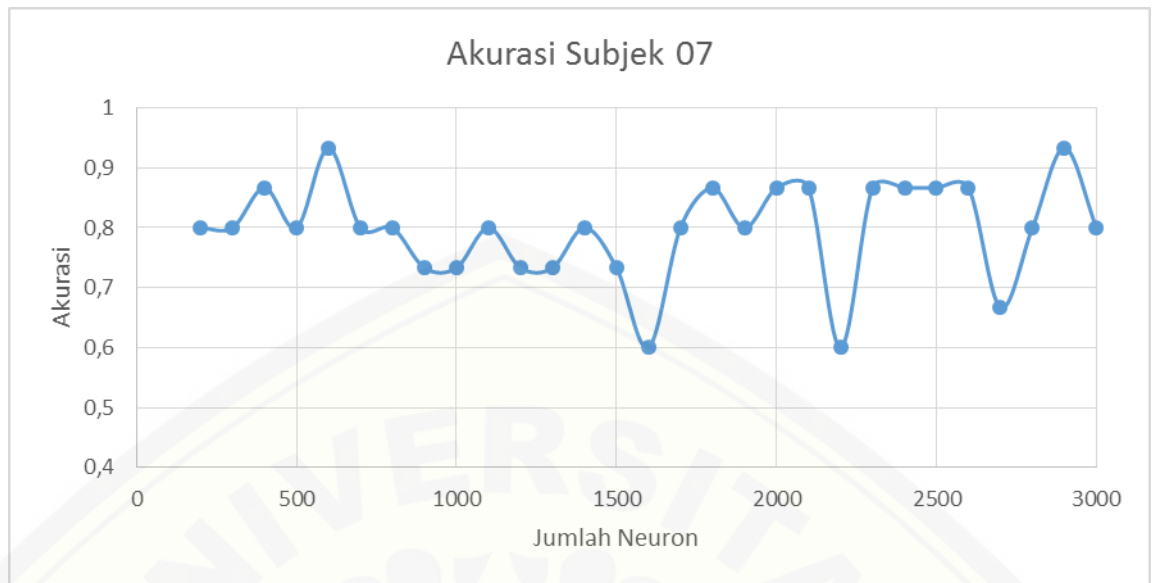
Grafik Akurasi Subjek 04



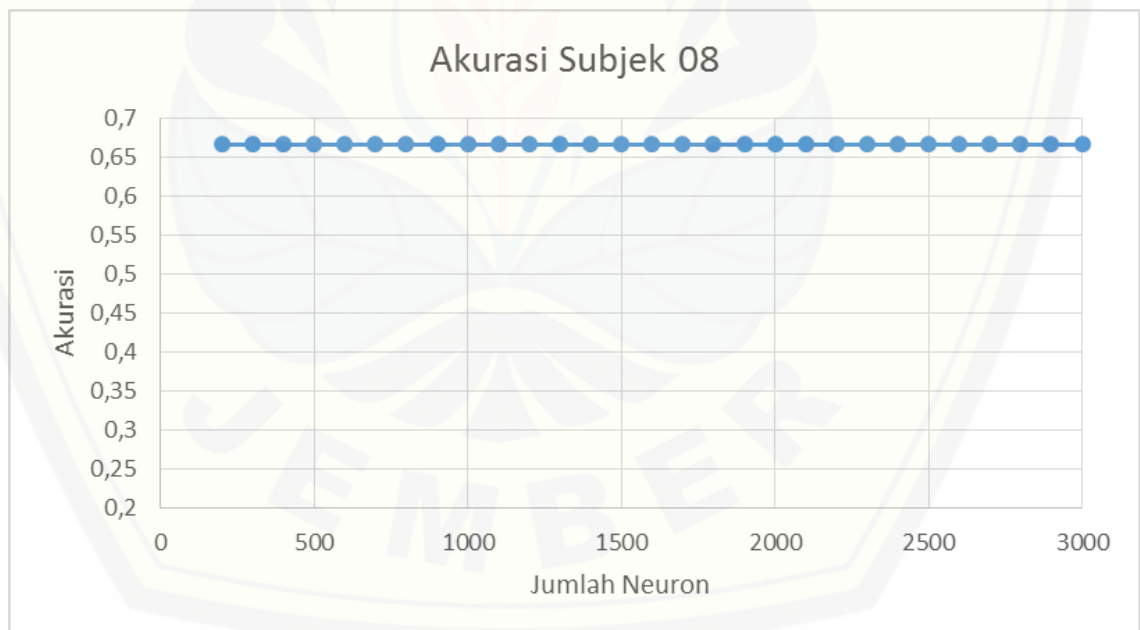
Grafik Akurasi Subjek 05



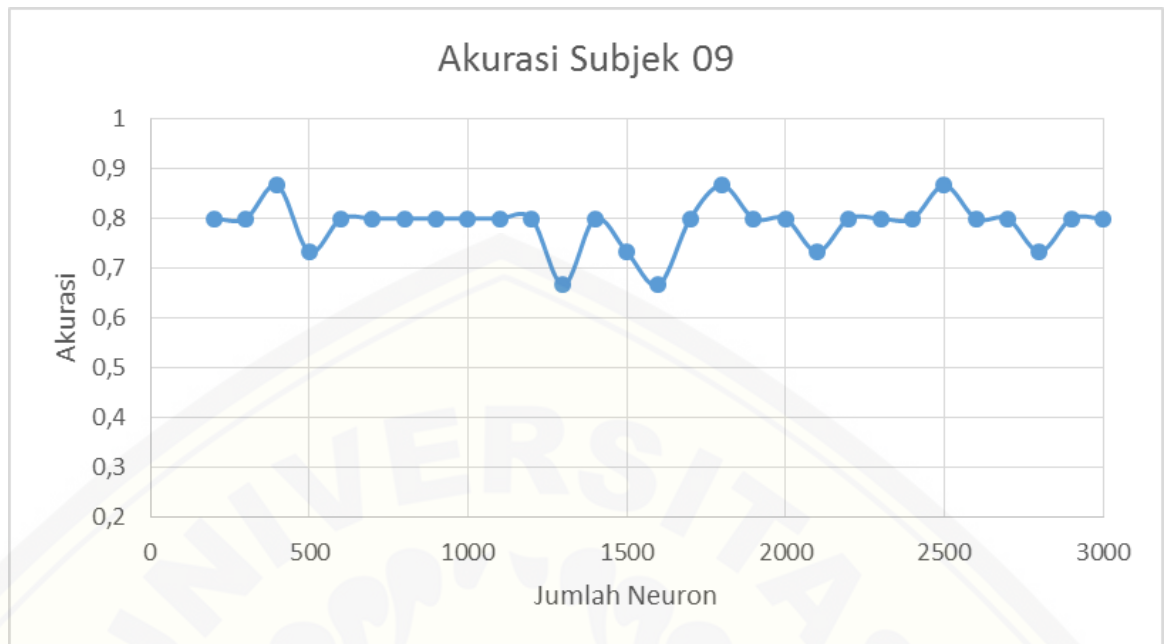
Grafik Akurasi Subjek 06



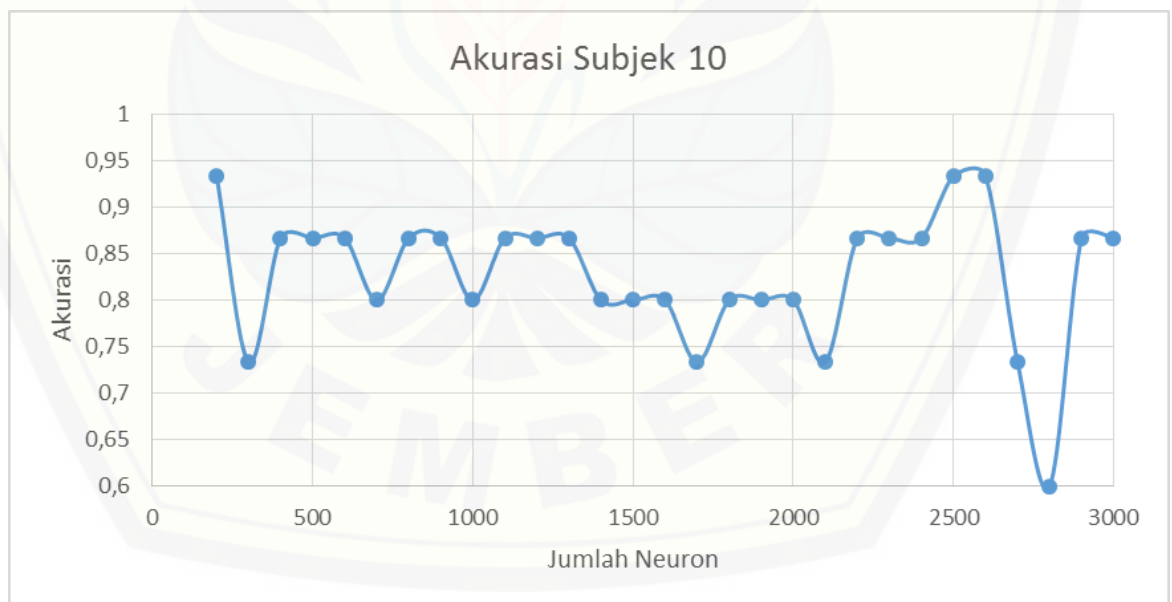
Grafik Akurasi Subjek 07



Grafik Akurasi Subjek 08



Grafik Akurasi Subjek 09



Grafik Akurasi Subjek 10