

# **RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER PADA CITRA PLAT NOMOR KENDARAAN**

## **SKRIPSI**



**Disusun Oleh:  
SYAIFUL ULUM  
10.18.022**

**PROGRAM STUDI TEKNIK INFORMATIKA S-1  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2014**

RANCANG BANGUN ALIRAN PEMROSESAN KARAKTER  
PADA CIKRA MAT NOMBOR KENDARAAN

SIMPULAN



Dibina Oleh:

SAIFUL ULLAH

1018022

PROGRAM STUDI TEKNIK INFORMATIKA S-1  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2014

**LEMBAR PERSETUJUAN DAN PENGESAHAN**

**RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER  
PADA CITRA PLAT NOMOR KENDARAAN**

**SKRIPSI**

*Disusun dan diajukan untuk melengkapi dan memenuhi persyaratan guna  
mencapai Gelar Sarjana Komputer Strata Satu (S-1)*

**Disusun Oleh :  
Syaiful Ulum  
10.18.022**

Diperiksa dan Disetujui,

Dosen Pembimbing I



**Ir. F. Yudi Limpraptono, MT**  
NIP.Y. 1039500274

Dosen Pembimbing II



**Ali Mahmudi, B.Eng.P.hd**  
NIP.P. 1031000429

Program Studi Teknik Informatika S-1  
Ketua



**Joseph Dedy Irawan, ST., MT.**  
NIP. 197404162005011002

**PROGRAM STUDI TEKNIK INFORMATIKA S-1  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG**

**2014**



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
PROGRAM STUDI TEKNIK INFORMATIKA S-1

---

## LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Syaiful Ulum  
NIM : 10.18.022  
Program Studi : Teknik Informatika S-1

Menyatakan bahwa skripsi yang berjudul:

RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER PADA CITRA  
PLAT NOMOR KENDARAAN

Merupakan hasil karya sendiri, bukan plagiasi dari karya orang lain. Dalam skripsi ini tidak memuat karya orang lain, kecuali dicantumkan sumbernya sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini penulis buat dengan sebenar-benarnya, dan apabila di kemudian hari penulis terbukti melakukan pelanggaran, maka penulis akan bersedia menerima sanksi yang berlaku sesuai ketentuan akademik.

Malang, Maret 2014

Penulis,

  
**METERAI  
TEMPEL**  
PAJAK PEMANGGUK BELANJA  
TGL. 20  
5A4DBACF141303821  
ENAM RIBU RUPIAH  
**6000** **DJP**  
**Syaiful Ulum**

# RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER PADA CITRA PLAT NOMOR KENDARAAN

Syaiful Ulum

Program Studi Teknik Informatika S-1  
Fakultas Teknologi Industri  
Institut Teknologi Nasional Malang  
Jl. Raya Karanglo Km 2 Malang  
Email : syaifululum24@gmail.com

**Dosen Pembimbing: 1. Ir. F. Yudi Limpraptono, MT**  
**2. Ali Mahmudi, B.Eng, PhD**

## Abstrak

*Penerapan optical character recognition untuk pengenalan plat nomor kendaraan adalah suatu pengenalan huruf dan angka yang terdapat pada plat nomor kendaraan. OCR (Optical Character Recognition) sendiri merupakan suatu teknologi yang memungkinkan mesin atau komputer secara otomatis dapat mengenali karakter-karakter yang terdapat pada suatu citra melalui suatu mekanisme optik. Proses yang dilakukan adalah mengubah suatu citra yang mengandung karakter-karakter yang dikenali komputer ke dalam teks yang dapat diolah oleh komputer. Optical Character Recognition digunakan untuk mengubah tipe dokumen yang berbeda, misalnya file PDF atau citra hasil rekaman kamera digital kedalam data yang dapat dimanipulasi.*

*Proses jaringan Backpropagation terdiri dari 3 sel neuron pada lapisan input yaitu  $x_1, x_2,$  dan  $x_3$ ; 2 sel neuron pada lapisan tersembunyi yaitu  $z_1$  dan  $z_2$ ; serta 1 sel neuron pada lapisan output yaitu  $y$ . Nilai bias  $b_1$  yang di berikan pada lapisan tersembunyi bertujuan untuk mengolah data input ditambah bobot  $V_{ij}$  yang masuk ke dalam sel-sel pada lapisan tersembunyi dengan bantuan fungsi aktivasi. Begitupula dengan nilai bias  $b_2$  yang diberikan pada lapisan keluaran adalah untuk mengolah data yang berasal dari keluaran sel pada lapisan tersembunyi ditambah bobot  $W_{ij}$  yang masuk kedalam lapisan keluaran dengan bantuan fungsi aktivasi.*

*Setelah dilakukan proses pengujian, diperoleh kesimpulan yaitu tingkat keberhasilan program pada pengujian dengan 10 sampel plat nomor kendaraan 7 sampel berhasil 100% terbaca, sedangkan 3 sampel plat nomor kendaraan dengan hasil yang tak sama dengan citra plat nomor.*

**Kata kunci** : plat nomor kendaraan, Optical character recognition, Jaringan backpropagation,

## LEMBAR PERSEMBAHAN

### Yang Utama Dari Segalanya

*Puji syukur kepada Allah S.W.T atas segala karunia yang telah diberikan kepada saya, sehingga saya dapat berdiri tegar dan menyelesaikan skripsi saya.*

*Kupersembahkan karya sederhana ini kepada orang yang sangat kukasih dan kusayangi.*

### Ibunda dan Ayahanda Tercinta

*Sebagai tanda bakti, hormat, dan rasa terima kasih yang tiada terhingga kupersembahkan karya kecil ini kepada:*

*Ibu (Laminten) dan Ayah (M. Suhadak)*

*yang telah memberikan kasih sayang, segala dukungan, dan cinta kasih yang tiada terhingga yang tiada mungkin dapat kubalas hanya dengan selembar kertas yang bertuliskan kata cinta dan persembahan. Semoga ini menjadi langkah awal untuk membuat Ibu dan Ayah bahagia karna kusadar, selama ini belum bisa berbuat yang lebih. Untuk Ibu dan Ayah yang selalu membuatku termotivasi dan selalu menyirami kasih sayang, selalu mendoakanku, selalu menasehatiku menjadi lebih baik,*

*Terima Kasih Ibu.... Terima Kasih Ayah.....*

### Sahabat Terbaikku

*Sahabat terbaikku "Arif Agung Setiawan, Risky Asvin Weluart, Yusuf Sofian, Wahyudi Hatta, Fifi Safariah, Syaiful Ulum, Mina Djonler, Resti Julianti, Sri Wijayanti Tri Hapsari, Taufiqur Rusda, Ratna Permatasari, Erika Fitriana, Nafi Kaharudin, Bisri Ismail, Riza Juliansyah, Nur Andika Apriyuda, Tri Wahyu, Ana Syariatul Hawa, Badriyatul Mu'azaroh, Gresia Sera Fristantya, Kadek Nuryasa, Cahya Setyawan" terima kasih atas bantuan, doa, nasehat, hiburan, dan semangat yang kalian berikan.*

## **Lab. Pengolahan Citra & Multimedia**

*Seluruh asisten laboratorium pengolahan citra & multimedia "Thio Wangga, Dedik, Nanang, Zulio, Tari, Ning, Erik, Bagus, Aziz, Wisnu, Vanning, Sofina, Toha Indra, Gelar, Rozani" terima kasih atas bantuan, kerjasama, dan semangat yang kalian berikan sampai saat ini. Sangat menyenangkan bergabung satu lab dengan kalian semua.*

### **Dosen Pembimbing Skripsi**

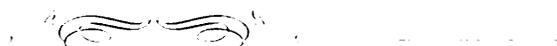
*Bapak Ir. Yudi Limpraptono dan Bapak Ali Mahmudi, B.Eng.P.hd. selaku dosen pembimbing skripsi, terima kasih atas nasehat, bimbingan, dukungan, serta bantuannya selama ini.*

### **Seluruh Dosen Pengajar**

*Ibu Febriana Santi Wahyuni, S.Kom., M.Kom., Ibu Nurlailiy Vendyansyah, ST., Bapak Ali Mahmudi, B.Eng., P.hD., Bapak Yosep Agus Pranoto, ST., MT., Bapak Michael Ardita, ST., MT., Bapak Suryo Adi Wibowo, ST., MT. terima kasih untuk semua ilmu, dan pengalaman yang sangat berarti yang telah Bapak dan Ibu berikan kepada kami.*

### **Staf Akademik**

*Pak Dhe, Ibu Titik dan semua staf akademik di Fakultas Teknologi Industri, terima kasih atas semua bantuan Bapak dan Ibu.*



## **KATA PENGANTAR**

Puji dan Syukur penyusun panjatkan kehadiran Tuhan Yang Maha Esa, karena telah memberikan rahmat-Nya sehingga penyusun dapat menyelesaikan skripsi dengan judul **RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER PADA CITRA PLAT NOMOR KENDARAAN**

sesuai dengan waktu yang ditentukan.

Penyusunan skripsi ini merupakan salah satu persyaratan untuk menyelesaikan program pendidikan Strata Satu (S-1) Teknik Informatika, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Pada kesempatan ini penyusun mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Ir. Soeparno Djiwo, MT. selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. H. Anang Subardi, MT. selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Joseph Dedy Irawan, ST., MT. selaku Ketua Program Studi Teknik Informatika S1 Institut Teknologi Nasional Malang, atas berbagai kebijakan dan arahnya.
4. Bapak Sonny Prasetyo, ST., MT. selaku Sekretaris Program Studi Teknik Informatika S1 Institut Teknologi Nasional Malang yang telah memberikan banyak arahan, motivasi serta bimbingan kepada penulis dalam proses menyusun skripsi.

5. Bapak Ir. F. Yudi Limpraptono, MT. selaku Pembimbing I yang telah banyak memberikan arahan, motivasi dan petunjuk selama penyusunan skripsi ini, sehingga skripsi ini dapat terselesaikan.
6. Bapak Ali Mahmudi, B.Eng, PhD selaku Pembimbing II yang telah banyak memberikan arahan, motivasi dan petunjuk, sehingga skripsi ini dapat terselesaikan.
7. Dosen pengajar dan seluruh karyawan FTI Institut Teknologi Nasional Malang atas segala bantuannya selama perkuliahan.
8. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu berpartisipasi dalam pelaksanaan penelitian ini.

Penyusun menyadari bahwa skripsi masih jauh dari sempurna, oleh karena itu penyusun mengharapkan kritik dan saran dari pembaca. Semoga skripsi ini bisa bermanfaat bagi para pembaca sekalian.

Malang, Maret 2014

Penyusun

## DAFTAR ISI

<b>HALAMAN JUDUL</b>	
<b>LEMBAR PERSETUJUAN DAN PENGESAHAN .....</b>	<b>ii</b>
<b>LEMBAR PERNYATAAN KEASLIAN .....</b>	<b>iii</b>
<b>ABSTRAK .....</b>	<b>iv</b>
<b>LEMBAR PERSEMBAHAN.....</b>	
<b>KATA PENGANTAR .....</b>	<b>v</b>
<b>DAFTAR ISI .....</b>	<b>vii</b>
<b>DAFTAR GAMBAR .....</b>	<b>x</b>
<b>DAFTAR TABEL .....</b>	<b>xiii</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Batasan Masalah .....	3
1.4. Tujuan .....	4
1.5. Metodologi .....	4
1.6. Sistematika Penulisan.....	5
<b>BAB II LANDASAN TEORI .....</b>	<b>7</b>
2.1. <i>License Plate Recognition (LPR)</i> .....	7
2.1.1. Fungsi LPR ( <i>License Plate Recognition</i> ) .....	7
2.1.2. Masalah yang ada dalam LPR .....	10
2.2. <i>Image Processing</i> .....	10
2.3. <i>Visi Compter (Computer Vision)</i> .....	13
2.4. <i>Optical Character Recognition</i> .....	16

2.5. Jaringan Syaraf Tiruan .....	17
2.5.1. Model Sel Syaraf ( <i>Neuron</i> ) .....	18
2.5.2. Fungsi Aktivasi .....	21
2.5.3. Arsitektur Jaringan .....	27
2.5.4. Backpropagation .....	29
2.6. Matlab .....	34
<b>BAB III PERANCANGAN DAN ANALISA SISTEM .....</b>	<b>36</b>
3.1. Citra Plat Nomor .....	36
3.1.1. Spesifikasi Teknis .....	36
3.2. Perancangan Blok Diagram Pada Aplikasi .....	38
3.3. Perancangan Blok Diagram Optical Character Recognition .....	39
3.4. Perancangan Flowchart.....	41
3.4.1. Flowchart Aplikasi .....	41
3.4.2. Flowchart OCR.....	42
3.4.3. Flowchart Metode Backpropagation.....	44
3.5. Perancangan Layout Aplikasi .....	47
3.6. Perancangan Tombol .....	48
3.7. Perancangan Tampilan Hasil .....	49
<b>BAB IV IMPLEMENTASI DAN PENGUJIAN.....</b>	<b>52</b>
4.1. Implementasi .....	52
4.2. Pembuatan tampilan aplikasi .....	52
4.3. Tampilan aplikasi .....	55
4.4. Proses aplikasi penerapan OCR .....	56
4.4.1. Tampilan Input Plat Nomor .....	56
4.4.2. Tampilan Proses Cropping .....	57
4.4.3. Tampilan Proses Binerisasi .....	58
4.4.4. Tampilan Line Segmentation .....	58
4.4.5. Tampilan Character Segmentation .....	59
4.4.6. Tampilan Pembacaan Pola .....	60
4.4.7. Tampilan Proses Learning Backptopagation .....	60

4.4.8. Tampilan Proses .....	61
4.4.9. Tampilan Keluar Aplikasi .....	62
4.5. Pengujian Aplikasi .....	63
<b>BAB V PENUTUP .....</b>	<b>69</b>
5.1. Kesimpulan .....	69
5.2. Saran .....	69
Daftar Pustaka .....	71
Lampiran	

## DAFTAR GAMBAR

Gambar 2.1	Tahap – tahap pengolahan citra yang digunakan .....	11
Gambar 2.2	Skema hubungan visi komputer dengan bidang lain .....	15
Gambar 2.3	Blok diagram kerja OCR .....	16
Gambar 2.4	Model matematis <i>nonlinier</i> dari suatu <i>neuron</i> .....	19
Gambar 2.5	Model matematis <i>nonlinier</i> dari suatu <i>neuron</i> .....	21
Gambar 2.6	Fungsi naik biner hard limit .....	22
Gambar 2.7	Fungsi naik biner threshold .....	22
Gambar 2.8	Fungsi bipolar hard limit .....	23
Gambar 2.9	Fungsi bipolar dengan threshold .....	23
Gambar 2.10	Fungsi linier (identitas) .....	24
Gambar 2.11	Fungsi Saturating linier .....	24
Gambar 2.12	Fungsi symmetric saturating linier .....	25
Gambar 2.13	Fungsi sigmoid biner .....	26
Gambar 2.14	Fungsi sigmoid bipolar .....	27
Gambar 2.15	Jaringan layar tunggal .....	28
Gambar 2.16	Jaringan layar jamak .....	28
Gambar 2.17	Recurrent network .....	29
Gambar 2.18	Arsitektur Jaringan .....	30
Gambar 2.19	Matlab versi 7.12.0 (Matlab R2011a) .....	35
Gambar 3.1	Tanda Nomor Kendaraan Bermotor (TNKB) .....	36
Gambar 3.2	TNKB Kendaraan Bermotor Roda 2 atau 3 .....	37

Gambar 3.3	TNKB Kendaraan Bermotor Roda 4 atau Lebih .....	37
Gambar 3.4	Blok Diagram Aplikasi .....	39
Gambar 3.5	Blok Diagram Proses Optical Character Recognition .....	39
Gambar 3.6	Flowchart Perancangan Aplikasi .....	42
Gambar 3.7	Flowchart OCR .....	43
Gambar 3.8	Flowchart Metode Learning Backpropagation .....	45
Gambar 3.9	Perancangan Layout Aplikasi .....	47
Gambar 4.1	Tampilan Awal Pembuatan <i>Figure</i> .....	53
Gambar 4.2	Tampilan Design .....	54
Gambar 4.3	Tampilan Aplikasi .....	56
Gambar 4.4	Tampilan Proses Input Plat Nomor .....	57
Gambar 4.5	Tampilan Proses Cropping Image .....	57
Gambar 4.6	Tampilan Proses Binerisasi .....	58
Gambar 4.7	Tampilan Line Segmentation .....	59
Gambar 4.8	Tampilan Character Segmentation .....	59
Gambar 4.9	Tampilan Pembacaan Pola .....	60
Gambar 4.10	Tampilan Proses Learning Backpropagation .....	61
Gambar 4.11	Tampilan Proses .....	62
Gambar 4.12	Tampilan Keluar Aplikasi .....	62
Gambar 4.13	Sampel Plat Nomor Kendaraan .....	63

## **DAFTAR TABEL**

Tabel 3.1	Perancangan Nama Tools .....	50
Tabel 4.1	Perancangan Nama Tools .....	54
Tabel 4.2	Hasil Pengujian Terhadap Sampel Citra Berdasarkan Type Gambar .....	64
Tabel 4.3	Hasil Pengujian Terhadap Sampel Berdasarkan Jarak pemotretan .....	65
Tabel 4.4	Hasil Pengujian dengan nilai threshold terbaik .....	67

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Semua kendaraan bermotor di seluruh dunia wajib memiliki plat nomor. Biasanya plat nomor dipasang di depan dan belakang kendaraan. Namun ada yurisdiksi tertentu atau jenis kendaraan tertentu yang hanya membutuhkan satu plat nomor, untuk dipasang di bagian belakang. Plat nomor memiliki nomor seri yakni susunan huruf dan angka yang dikhususkan bagi kendaraan tersebut. Di Indonesia disebut nomor polisi, dan biasa dipadukan dengan informasi lain mengenai kendaraan bersangkutan, seperti warna, merk, model, tahun pembuatan, nomor identifikasi kendaraan atau Vehicle Identification Number (VIN) dan tentu saja nama dan alamat pemiliknya. Semua data ini juga tertera dalam Surat Tanda Nomor Kendaraan Bermotor atau STNK yang merupakan surat bukti bahwa nomor polisi itu memang ditetapkan bagi kendaraan tersebut.

Dalam perkembangan saat ini, banyak teknologi yang sudah berkembang pesat sehingga banyak aplikasi yang sudah berkembang pada masa sekarang. Terutama pada aplikasi yang mengolah suatu karakter angka maupun huruf yang ada saat ini. Aplikasi yang mengolah dan membaca karakter angka dan huruf disebut *Optical Character Recognition*. *Optical Character Recognition* ini digunakan untuk aplikasi penerapan pada plat nomor kendaraan.

Penerapan *optical character recognition* untuk pengenalan plat nomor kendaraan adalah suatu pengenalan huruf dan angka yang terdapat pada plat nomor kendaraan. OCR (*Optical Character Recognition*) sendiri merupakan

suatu teknologi yang memungkinkan mesin atau komputer secara otomatis dapat mengenali karakter-karakter yang terdapat pada suatu citra melalui suatu mekanisme optik. Proses yang dilakukan adalah mengubah suatu citra yang mengandung karakter-karakter yang dikenali komputer ke dalam teks yang dapat diolah oleh komputer. *Optical Character Recognition* digunakan untuk mengubah tipe dokumen yang berbeda, misalnya file PDF atau citra hasil rekaman kamera digital kedalam data yang dapat dimanipulasi.

Untuk mengolah karakter-karakter yang terdapat pada citra menggunakan bantuan sistem jaringan syaraf seperti Jaringan Syaraf Tiruan (JST). Jaringan Syaraf Tiruan sendiri adalah sistem pemrosesan informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi. Jaringan syaraf tiruan dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi. Metode ini menggunakan elemen perhitungan *non-linier* dasar yang disebut dengan neuron yang diorganisasikan sebagai jaringan yang saling berhubungan, sehingga mirip dengan jaringan syaraf manusia. Jaringan syaraf tiruan terdapat 6 metode *learning*, yaitu : *Metode Neuron, Metode Hebb, Metode Perceptron, Metode Adaline, Metode Backpropagation, Metode Kohonen.*

Dengan menggunakan pengenalan OCR dan metode JST yang akan diimplementasikan pada penerapan *optical character recognition* untuk plat nomor kendaraan bermotor. Dengan penerapan *optical character recognition* untuk mengenali setiap karakter yang terdapat pada plat nomor kendaraan.

## **1.2. Rumusan Masalah**

Adapun rumusan masalah dari penelitian ini adalah bagaimana cara menerapkan *optical character recognition* untuk pengenalan plat nomor

kendaraan yang akan mengolah karakter pada plat nomor dengan bentuk citra agar menjadi teks yang sama hasilnya pada plat nomor tersebut.

### 1.3. Batasan Masalah

Adapun batasan-batasan masalah dalam pembuatan program ini adalah sebagai berikut :

1. Gambar digital yang digunakan adalah plat nomor dari kendaraan.
2. Untuk proses deteksi plat nomor ini dengan menggunakan huruf A sampai Z dan angka 0 sampai 9.
3. Plat nomor yang digunakan adalah yang berwarna putih pada karakter dan warna hitam pada dasar.
4. Huruf yang dideteksi sebanyak 3 huruf dan angka yang dideteksi sebanyak 4 angka .
5. Dalam penerapan *optical character recognition* ini berjumlah 7 karakter.
6. Penerapan *optical character recognition* ini dengan menggunakan inputan citra atau gambar berekstensi .JPG.
7. Metode *learning* yang digunakan untuk mendeteksi plat nomor pada kendaraan adalah pengenalan menggunakan JST dengan metode *learning Backpropagation*.
8. *Software* yang digunakan adalah Matlab R2011a (versi 7.12.0).
9. Bahasa pemrograman yang dipakai adalah bahasa pemrograman MATLAB (*Matrix Laboratory*).

#### 1.4. Tujuan

Tujuan pembuatan program aplikasi yang akan dibuat adalah menerapkan sistem *optical character recognition* yang digunakan untuk melakukan pembacaan sebuah karakter pada plat nomor kendaraan.

#### 1.5. Metodologi

Adapun metode yang dipergunakan dalam penyusunan skripsi ini adalah sebagai berikut :

##### 1. Studi literature

Merupakan serangkaian kegiatan mengumpulkan data dan referensi yang berhubungan dengan penyusunan laporan ini. Sumber yang digunakan antara lain adalah buku dan berbagai sumber dari internet. Studi literatur difokuskan penerapan *optical character recognition* untuk pengenalan pada plat nomor kendaraan.

##### 2. Perancangan sistem

Penerapan *optical character recognition* untuk plat nomor ini mengenali dan mendeteksi citra pada huruf dan angka yang berjumlah 7 karakter. Perancangan sistem OCR yaitu : *image acquisition* dengan merubah citra asli menjadi citra yang sudah digitalisasi, kemudian citra akan diperbaiki dengan proses *pre processing*. Setelah itu proses *document and analysis* mendeteksi huruf dan angka pada citra dengan memisah tiap karakter pada huruf dan angka tersebut. Kemudian proses *feature extraction* akan menjadikan citra huruf dan angka yang terbagi untuk dikumpulkan. Setelah itu proses *training and recognition* untuk mencocokkan citra yang sudah terbagi dan mengklasifikasi karakter plat nomor, untuk proses terakhir yaitu proses untuk mencetak karakter plat nomor kendaraan.

### 3. Implementasi

Pada pembuatan program aplikasi ini menerapkan *optical character recognition* untuk plat nomor kendaraan, perangkat lunak yang digunakan adalah Matlab (*Matrix Laboratory*) versi R2011a. Perangkat lunak ini dipilih karena pengolahan citra pada dasarnya merupakan pengolahan matrik, sehingga Matlab adalah perangkat lunak yang tepat dalam hal ini. Pembuatan program dibagi ke dalam dua bagian, yaitu pembuatan interface dan fungsi-fungsi pengolahan citra.

### 4. Pengujian sistem

Tahap pengujian dilakukan jika semua bagian sistem telah selesai dibuat. Pengujian dilakukan ke semua bagian interface sistem untuk mengetahui apakah sistem berjalan sesuai dengan yang diharapkan.

## 1.6. Sistematika Penulisan

Dalam penyusunan skripsi ini agar lebih mudah dipahami maka dibuatlah suatu sistematika penulisan sebagai berikut:

#### BAB I : Pendahuluan

Berisi Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan, Metodologi Penelitian dan Sistematika Penulisan.

#### BAB II : Landasan Teori

Berisi teori-teori yang didapat dari studi literatur dan konsep-konsep yang terkait dengan skripsi ini, beserta dengan penyelesaian masalah yang diambil dalam penyusunan skripsi

**BAB III : Perancangan Aplikasi**

Dalam bab ini berisi mengenai analisa dan perancangan dari sistem yang akan dibangun meliputi analisa sistem, komponen sistem pakar, rancangan basis data dan perancangan antarmuka.

**BAB IV : Implementasi dan Pengujian**

Berisi tentang implementasi dan cara menjalankan aplikasi serta uji coba dari program yang telah dibuat tersebut.

**BAB V : Penutup**

Merupakan bab terakhir yang memuat kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan penelitian selanjutnya.

## BAB II

### LANDASAN TEORI

#### 2.1. License Plate Recognition (LPR)

*License Plate Recognition* (LPR) atau sering disebut juga dengan *Automatic Number Plate Recognition* (ANPR) merupakan salah satu teknologi atau sistem yang menangani pengenalan nomor polisi yang ada pada plat nomor kendaraan bermotor. Pengenalan nomor polisi yang ada pada plat nomor kendaraan tersebut dapat dilakukan secara otomatis dengan menggunakan teknologi seperti OCR (*Optical Character Recognition*).[1]

##### 2.1.1. Fungsi LPR (*License Plate Recognition*)

Berdasarkan kemampuannya, aplikasi LPR dapat mengenali pola nomor polisi secara otomatis dan juga dapat menyimpan gambar atau citra dari plat nomor yang telah diambil pada proses *capturing image*. Kemampuan tersebut tentu saja sangat berguna, menurut Markus Friedriech dkk (2008) dalam paper yang berjudul *Automatic Number Plate Recognition for The Observation of Travel Behavior* bahwa aplikasi ANPR atau LPR dapat menyediakan *data source* yang sangat bermanfaat untuk perencanaan transportasi dan *engineering*, kemudian *multiple task* dapat diselesaikan dengan aplikasi tersebut. ANPR atau LPR juga dapat berguna untuk lima sistem berikut, yaitu: *vehicle classification, travel time measurement, through traffic surveys, route choise observations* dan *estimation of O-D matrices*. Adapun menurut Hofman (2004) dalam skripsi milik Chandra

Yansen dan Arryandra Luis (2006) yang berjudul Pengenalan Plat Nomor Otomatis dengan Metode Morfologi, *Connected Component Labeling* dan *Bidirectional Associative Memory* disebutkan mengenai lingkup bidang aplikasi yang menggunakan nomor plat yaitu sebagai berikut:

1. **Perpajakan** : Nomor plat digunakan untuk identitas pelanggan pra bayar untuk masuk ke dalam parkir. Selain itu digunakan untuk menghitung biaya parkir yang bukan pelanggan dengan membandingkan waktu masuk dan waktu keluar. Citra plat nomor tersebut juga dapat digunakan untuk mencegah pencurian jika terdapat wajah pengemudi di dalam citranya.
2. **Kontrol Akses** : Suatu gerbang atau palang akan terbuka secara otomatis untuk anggotanya sehingga mendapatkan *secure area* dan tidak perlu lagi memerlukan penjaga gerbang.
3. **Jalan Tol** : Nomor plat digunakan untuk menghitung biaya jalan tol atau digunakan untuk memeriksa ulang tiket.
4. **Kontrol Batas Negara** : Sistem pengontrolan batas negara diperlukan untuk mempercepat proses penyebrangan wilayah. Nomor plat dicatat di pintu masuk atau keluar suatu negara dan digunakan untuk memantau batas Negara.
5. **Pencurian mobil** : Daftar dari mobil yang dicuri atau belum dibayar yang terdapat di dalam "*black list*" dapat dilacak. Data "*Black list*" dapat diupdate secara *real time* dan menyediakan *alarm* langsung untuk polisi. Sistem LPR ini dipasang di jalan dan melakukan pengecekan secara *real time* antara mobil yang lewat dengan yang ada dalam daftar, ketika ditemukan, sirene akan dibunyikan dan petugas akan diberitahu.

6. **Enforcement** : Nomor plat dapat digunakan untuk menilang kendaraan yang melebihi batas kecepatan atau lampu merah. Proses otomatis akan lebih menghemat waktu dan tenaga dibandingkan proses konvensional. Selain itu biaya tilang dapat dibayar secara *online* karena secara otomatis data pelanggaran sudah terekam ke dalam sistem.
7. **Pengendalian Lalu Lintas** : Kendaraan dapat diarahkan ke jalur lain berhubungan dengan izin masuk, sistem ini akan mengurangi kemacetan secara efektif. Contohnya aturan ganjil-genap yang akan dilaksanakan di Jakarta sekarang ini. Untuk membedakan plat ganjil dan plat nomor genap LPR sangat diperlukan.
8. **Alat Bantu Pemasaran** : Nomor plat dapat digunakan untuk membuat data untuk keperluan pemasaran atau untuk membangun profil lalu lintas (seperti frekuensi masuk per jam atau hari).
9. **Traffic control** : Beberapa LPR yang diletakkan di beberapa titik di jalan dapat digunakan untuk menghitung kemacetan dan waktu rata-rata antara dua titik jalan dan digunakan untuk memantau kepadatan jalan raya.
10. **Perparkiran Bandara** : Untuk mengurangi kesalahan tiket, LPR digunakan untuk menangkap nomor plat dan gambar mobil. Informasi ini akan digunakan untuk menghitung biaya parkir atau memberikan bukti saat kehilangan tiket.[1]

### **2.1.2. Masalah yang Ada dalam LPR**

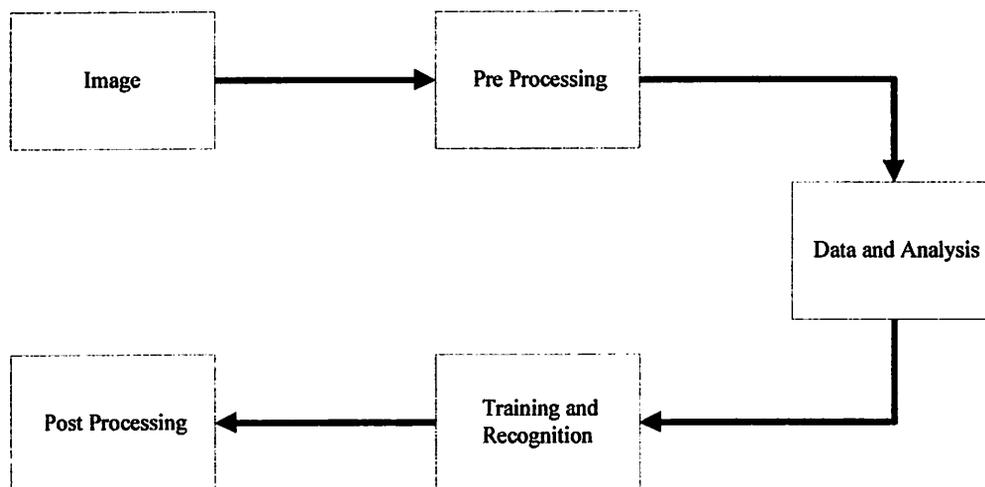
Masalah yang sering muncul dalam pengembangan aplikasi ini yaitu banyak terjadi pada tahap pengambilan gambar atau *capturing image*. Menurut Wikipedia (2012) masalah yang umum dan sering ada yaitu:

1. Resolusi citra input yang tidak sesuai, biasanya disebabkan pengambilan citra plat nomor yang terlalu jauh dari kameranya atau dikarenakan spesifikasi kamera yang kurang.
2. Citra tidak fokus atau blur, bisa dikarenakan kamera yang bergoyang saat pengambilan citra atau lensa kamera yang kotor.
3. Terdapat benda atau objek yang menghalangi plat nomor seperti mur atau yang lainnya.
4. Perbedaan format, stuktur, jenis huruf dari nomor polisi yang ada pada plat nomor di setiap negara.
5. Kurangnya kordinasi antara negara atau wilayah, terkadang terdapat kasus dua mobil dari negara yang berbeda memiliki nomor plat yang sama, meskipun desain yang berbeda.[1]

### **2.2. Image Processing (Pengolahan Citra)**

Pengolahan Citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua

data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh bit-bit tertentu, seperti ditunjukkan pada Gambar 2.1.



**Gambar 2.1** Tahap-tahap Pengolahan Citra Yang Digunakan

Berikut pengertian dari beberapa proses pengolahan citra yang digunakan :

### 1. *Image Acquisition*

Tahapan awal dalam *computer vision* adalah *image acquisition* (pengambilan digital *image*). *Image acquisition* berhubungan dengan sensor yang mengambil *image*. Sensor yang digunakan bisa kamera ataupun *scanner*. Sifat dari sensor dan *image* yang dihasilkan ditentukan dari aplikasinya (kulkarni, 2001, p9).

### 2. Pre Processing

*Preprocessing* adalah tahap dimana citra yang telah didigitalisasi akan diberikan perbaikan-perbaikan seperti *noise removal*, *sharpening*, *skew correction*, dan sebagainya. Pada proses ini pula citra RGB yang telah digitalisasi

akan diubah menjadi citra biner, dimana pada tahap pemrosesan lebih lanjut akan dibutuhkan citra dengan nilai-nilai piksel yang pasti (0 atau 1). Secara umum, nilai nol mewakili intensitas paling gelap dan nilai satu mewakili nilai intensitas paling terang.

### 3. Data and Analysis

Analisa citra merupakan proses dimana bertujuan untuk mendapatkan objek-objek tertentu di dalam citra. Terdapat dua hal pokok proses analisis yang dilakukan. Pertama adalah pengenalan karakter, dan yang kedua adalah analisis format halaman dokumen. Pengenalan karakter berhubungan dengan karakter - karakter penyusun teks, sedangkan analisis format halaman dokumen berhubungan dengan penentuan format suatu halaman serta posisi dan fungsi dari kelompok - kelompok teks penyusun dokumen, misalnya judul, sub judul, catatan kaki, dan lain sebagainya.

Proses yang mungkin dilakukan antara lain adalah edge detection (deteksisi) atau segmentasi. Proses segmentasi yang dilakukan adalah memilah sekumpulan nominal angka di dalam region (area) tertentu ke dalam karakter majemuk (teks). kedua proses analisis tersebut dapat dilakukan secara terpisah atau bisa jadi hasil dari satu proses analisis dipakai untuk melakukan analisis lain. Pada beberapa kasus kadang diperlukan prosedur untuk memperbaiki kecondongan dari dokumen yang diakibatkan oleh peletakan dokumen yang kurang benar pada saat dilakukan proses pembacaan dokumen oleh mesin scan.

#### 4. Training And Recognition

Proses pembelajaran dan pelatihan melibatkan jaringan syaraf tiruan untuk menentukan hasilnya. Banyak tipe jaringan syaraf tiruan yang akan dipakai antara lain *BackPropagation* (Perambatan Balik), *Perceptron*, atau *Kohonen* (Pengaturan mandiri). Tipe yang digunakan pada laporan skripsi ini adalah dengan menggunakan *Backpropagation*.

#### 5. Post Processing

Suatu proses citra yang sudah meliputi berbagai langkah proses sebelumnya, kemudian dimunculkan pada hasil akhir yang akan menentukan jalannya dari proses-proses sebelumnya. Kemudian pada proses akhir ini akan menampilkan hasil dari berbagai proses yang sudah terjadi.

### 2.3. Visi Komputer (*Computer Vision*)

*Computer Vision* adalah ilmu dan teknologi mesin yang melihat, di mana mesin mampu mengekstrak informasi dari gambar yang diperlukan untuk menyelesaikan tugas tertentu. Sebagai suatu disiplin ilmu, visi komputer berkaitan dengan teori di balik sistem buatan bahwa ekstrak informasi dari gambar. Data gambar dapat mengambil banyak bentuk, seperti urutan video, pandangan dari beberapa kamera, atau data multi-dimensi dari scanner medis. Sedangkan sebagai disiplin teknologi, computer vision berusaha untuk menerapkan teori dan model untuk pembangunan sistem computer vision.

*Computer Vision* didefinisikan sebagai salah satu cabang ilmu pengetahuan yang mempelajari bagaimana komputer dapat mengenali obyek yang diamati. Cabang ilmu ini bersama *Artificial Intelligence* akan mampu menghasilkan *Visual Intelligence System*. Perbedaan antara *computer vision* dan

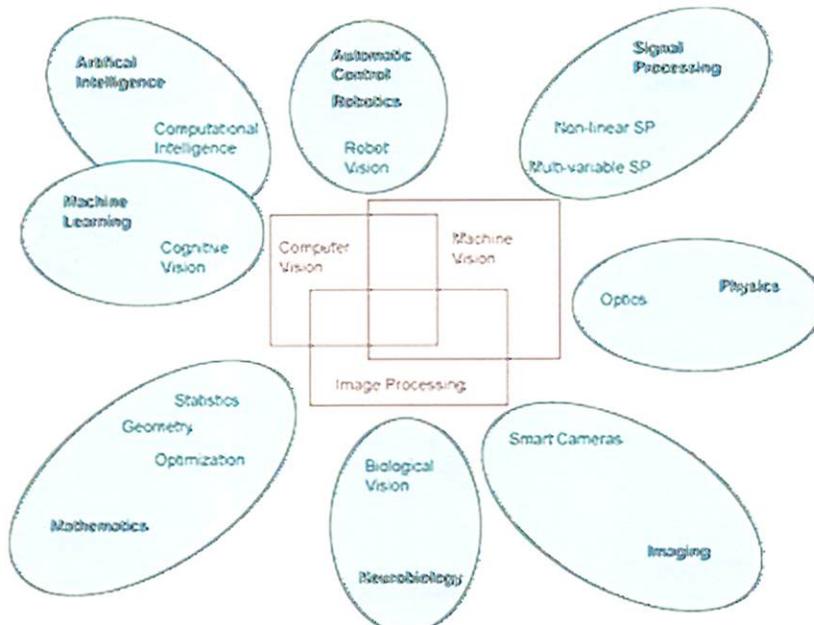
*Artificial Intelligence* pada *Computer Vision* lebih mempelajari bagaimana komputer dapat mengenali obyek yang diamati. Namun komputer grafik lebih ke arah pemanipulasian gambar (visual) secara digital. Bentuk sederhana dari grafik komputer adalah grafik komputer 2D yang kemudian berkembang menjadi grafik komputer 3D, pemrosesan citra, dan pengenalan pola. Grafik komputer sering dikenal dengan istilah visualisasi data.

*Computer vision* merupakan sub disiplin ilmu dari kecerdasan buatan yang mempelajari bagaimana komputer dapat mengenali objek yang diamati atau diobservasi. Dapat juga disebut sebagai *machine vision*, ilmu ini mengembangkan teori-teori dan algoritma dimana informasi yang berguna diekstraksi dan dianalisis dari sebuah citra penelitian, sekumpulan citra, atau citra yang berurutan dari sebuah komputasi yang dibuat oleh computer. *Computer vision* oleh beberapa ahli didefinisikan sebagai berikut :

1. Ballard dan Brown (Ballard & Brown, 1982), *Computer vision* adalah otomatis dan integrasi sebuah range yang luas yang terdiri dari proses-proses dan representasi-representasi terhadap persepsi visual.
2. Adrian Low (Low, 1991), *Computer vision* berhubungan dengan perolehan gambar, pemrosesan, klasifikasi, pengenalan, dan menjadi penggabungan, pengurutan pembuatan keputusan menuju pengenalan.
3. Michael G. Fairhurst (Fairhurst, 1988), *Computer vision* sesuai dengan sifatnya, merupakan suatu subyek yang merangkul berbagai disiplin tradisional secara luas guna mendasari prinsip-prinsip formalnya, dan dalam mengembangkan suatu metodologi yang berlainan dari apa yang

dimilikinya, pertama-tama harus mengembangkan dan secara berurutan membangun materi yang mendasari ini.

4. Shapiro dan Stockman (Shapiro & Stockman, 2001), *Computer vision* adalah suatu bidang yang bertujuan untuk membuat suatu keputusan yang berguna mengenai objek fisik nyata dan keadaan berdasarkan atas sebuah citra. *Computer vision* merupakan kombinasi antara pengolahan citra dan pengenalan pola. Hasil keluaran dari proses *Computer vision* adalah pengertian tentang citra, seperti yang ditunjukkan pada gambar 2.2.

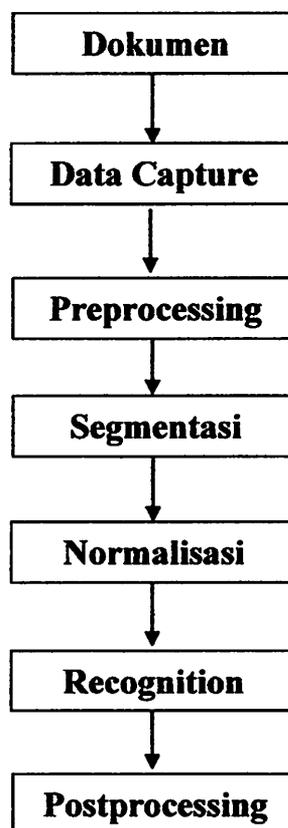


**Gambar 2.2** Skema hubungan *Computer vision* dengan bidang lain

Boyle dan Thomas (1988), mengatakan bahwa *Computer vision* lebih daripada pengenalan, *Computer vision* melakukan operasi “low level processing” sebagai algoritma *image processing* yang murni. Mereka juga yang menggolongkan *image processing* ke dalam *computer vision*. [2]

## 2.4. Optical Character Recognition

Optical character recognition (OCR) adalah sebuah sistem komputer yang dapat membaca huruf, baik yang berasal dari sebuah pencetak (printer atau mesin ketik) maupun yang berasal dari tulisan tangan. OCR adalah aplikasi yang menerjemahkan gambar karakter (image character) menjadi bentuk teks dengan cara menyesuaikan pola karakter per baris dengan pola yang telah tersimpan dalam database aplikasi. Hasil dari proses OCR adalah berupa teks sesuai dengan gambar output scanner dimana tingkat keakuratan penerjemahan karakter tergantung dari tingkat kejelasan gambar dan metode yang digunakan. Secara umum blok diagram kerja OCR dapat dilihat pada gambar 2.3.



Gambar 2.3 Blok diagram kerja OCR.

## 2.5. Jaringan Syaraf Tiruan

Mungkin kita sering memikirkan bagaimana hebatnya sistem syaraf manusia bekerja untuk mengenali berbagai macam pola. Misalkan, si A masih mampu mengenali wajah si B walaupun sudah lama tidak berjumpa dengannya dan meskipun pada periode tersebut si A banyak menemui wajah-wajah baru lainnya. Si A juga mampu membedakan berbagai macam ciri teman-temannya, seperti suara, gerakan tubuh (*gesture*), bahkan irama langkah kakinya, dan sebagainya. Mungkinkah kita bisa menirukan sistem syaraf manusia kedalam suatu teknik komputasi? JST merupakan salah satu upaya manusia untuk memodelkan cara kerja atau fungsi sistem syaraf manusia dalam melaksanakan tugas tertentu. Pemodelan ini didasari oleh kemampuan otak manusia dalam mengorganisikan sel-sel penyusunnya yang disebut *neuron*, sehingga mampu melaksanakan tugas-tugas tertentu, khususnya pengenalan pola dengan efektifitas yang sangat tinggi

Pada [HAY94) dinyatakan bahwa kecepatan sebuah *neuron* manusia dalam memproses suatu kejadian jauh lebih lambat bila dibandingkan dengan kecepatan gerbang logika silikon. Sebuah *neuron* memerlukan waktu  $10^{-3}$  detik (1 *milisecond*) untuk memproses suatu kejadian sedangkan gerbang logika silikon hanya membutuhkan waktu  $10^{-9}$  (1 *nanosecond*). Namun karena kemampuan sistem syaraf manusia yang sangat baik dalam mengorganisir kurang lebih 100 milyar *neuron* penyusunnya, maka kecepatan operasi secara keseluruhan yang dapat dicapai oleh sistem jaringan syaraf manusia menjadi lebih cepat daripada

kecepatan komputer yang paling cepat saat ini. Alexander dan Morton mendefinisikan JST sebagai berikut [HAY94]:

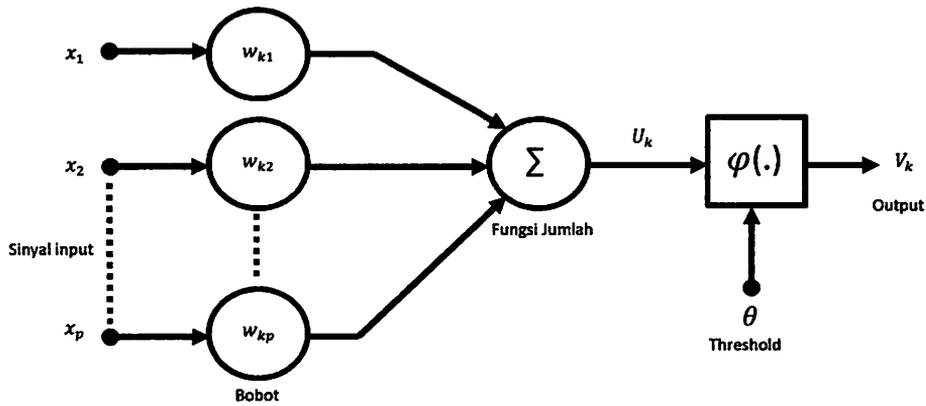
JST adalah prosesor tersebar paralel (*paralel distributed processor*) yang sangat besar yang memiliki kecenderungan untuk menyimpan pengetahuan yang bersifat pengalaman dan membuatnya siap untuk digunakan. JST menyerupai otak manusia dalam dua hal, yaitu: pengetahuan diperoleh jaringan melalui proses belajar; kekuatan hubungan antar sel syaraf (*neuron*) yang dikenal sebagai bobot-bobot sinaptik digunakan untuk menyimpan pengetahuan. [3]

### 2.5.1. Model Sel Syaraf (*Neuron*)

Sel syaraf (*neuron*) adalah unit pemrosesan informasi yang merupakan dasar dari operasi JST. Gambar 2.4 menunjukkan model suatu *neuron*. Terdapat tiga elemen dasar dari model *neuron*, yaitu:

1. Sekumpulan sinapsis atau jalur hubungan, dimana masing-masing sinapsis memiliki bobot atau kekuatan hubungan.
2. Suatu *adder* untuk menjumlahkan sinyal-sinyal *input* yang diberi bobot oleh sinapsis *neuron* yang sesuai. Operasi-operasi yang digambarkan di sini mengikuti aturan *linier combiner*.
3. Suatu fungsi aktivasi untuk membatasi amplitudo *output* dari setiap *neuron*.

Terdapat tiga variasi model *neuron* yang bisa digunakan karena ketiganya sebenarnya ekuivalen. Pertama, model *neuron* pada gambar 2.4. Model ini memasukkan *threshold*  $\theta_k$  (diterapkan secara eksternal) yang memperkecil nilai *input* untuk fungsi aktivasi. Sebaliknya, nilai *input* untuk fungsi aktivasi bisa diperbesar dengan menggunakan bias, yang merupakan kebalikan dari *threshold*.



**Gambar 2.4** Model Matematis Nonlinier Dari Suatu *Neuron*

Pada gambar 2.4 terlihat serangkaian aliran sinyal masukan  $x_1, x_2, \dots, x_p$  yang dipresentasikan oleh sebuah *neuron*. Sebuah *neuron* bisa memiliki banyak masukan dan hanya satu keluaran yang bisa menjadi masukan bagi *neuron-neuron* yang lain. Aliran sinyal masukan ini dikalikan dengan suatu penimbang (bobot sinapsis)  $w_{k1}, w_{k2}, \dots, w_{kp}$  dan kemudian dilakukan penjumlahan terhadap semua masukan yang telah diboboti tadi. Hasil penjumlahan ini disebut keluaran dari *the linier combiner*  $u_k$ .

Secara matematis kita bisa menggambarkan sebuah *neuron*  $k$  dengan menuliskan pasangan persamaan sebagai berikut :

$$u_k = \sum_{j=1}^p w_{kj} x_j \quad (2.1)$$

dan

$$y_k = \varphi(u_k - \theta_k), \quad (2.2)$$

Di mana  $x_1, x_2, \dots, x_p$  adalah sinyal *input*;  $w_{k1}, w_{k2}, \dots, w_{kp}$  adalah bobot-bobot sinaptik dari *neuron*  $k$ ;  $u_k$  adalah *linier combiner output*;  $\theta_k$  adalah *threshold*;  $\varphi(\cdot)$

adalah fungsi aktivasi; dan  $y_k$  adalah sinyal *output* dari *neuron*. Penggunaan *threshold* memberikan pengaruh adanya *affine transformation* terhadap *output*  $u_k$  dari *linier combiner* pada model gambar 2.1 sebagai berikut:

$$v_k = u_k - \theta_k \quad (2.3)$$

Secara khusus, tergantung apakah *threshold*  $\theta_k$  adalah positif atau negatif, hubungan antara tingkat aktivitas atau potensial aktivitas internal efektif  $v_k$  dari *neuron*  $k$  dan *linier combiner output*  $u_k$  dimodifikasikan dalam pola seperti pada gambar 2.2.

*Threshold*  $\theta_k$  adalah suatu parameter eksternal dari *neuron*  $k$ . Kita bisa membenarkan penggambaran ini seperti pada persamaan 2.5. secara ekivalen, kita bisa merumuskan kombinasi persamaan 2.4 dan 2.5 sebagai berikut:

$$v_k = \sum_{j=0}^p w_{kj} x_j \quad (2.4)$$

dan

$$y_k = \varphi(v_k) \quad (2.5)$$

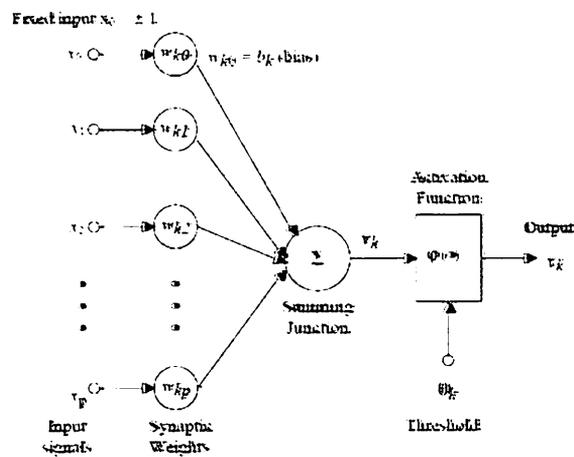
Pada persamaan 2.4 kita telah menambahkan suatu sinapsis baru, yang inputnya adalah

$$x_0 = -1 \quad (2.6)$$

dan bobotnya adalah

$$w_{k0} = \theta_k \quad (2.7)$$

Karena itu kita bisa merumuskan kembali model *neuron k* seperti gambar 2.2. pada gambar tersebut, efek *threshold* dirpresentasikan oleh: penambahan satu sinyal input *input* baru yang tetap yaitu -1; dan penambahan bobot sinaptik yang sama dengan *threshold*  $\theta_k$ .



**Gambar 2.5** Model Matematis Nonlinier Dari Suatu Neuron

## 2.5.2. Fungsi Aktivasi

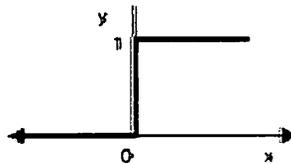
Fungsi aktivasi yang dinotasikan dengan  $\mu(\cdot)$  mendefinisikan nilai *output* dari suatu *neuron* dalam level aktivasi tertentu berdasarkan nilai *output* pengkombinasi *linier*  $u_i$ . Ada beberapa macam fungsi aktivasi yang biasanya digunakan pada JST, diantaranya adalah:

### 1. Fungsi Naik Biner *Hard Limit*

Fungsi naik biner biasanya digunakan oleh jaringan lapisan tunggal untuk mengonversi nilai input dari suatu variabel yang bernilai kontinu ke suatu

nilai *output* biner (0 atau 1) gambar 2.6. secara matematis, fungsi naik biner (*hard limit*) dituliskan sebagai berikut.

$$y = \begin{cases} 0, & \text{jika } x \leq 0 \\ 1, & \text{jika } x > 0 \end{cases} \quad (2.8)$$

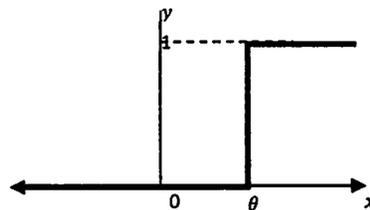


**Gambar 2.6** Fungsi Naik Biner Hard Limit

## 2. Fungsi Naik Biner *Threshold*

Berbeda dengan fungsi naik biner *hard limit*, fungsi naik biner *threshold* menggunakan nilai ambang  $\theta$  sebagai batasnya gambar gambar 2.7. secara matematis, fungsi naik biner *threshold* dituliskan sebagai berikut.

$$y = \begin{cases} 1, & \text{jika } x \geq \theta \\ 0, & \text{jika } x < \theta \end{cases} \quad (2.9)$$

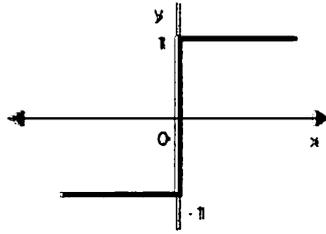


**Gambar 2.7** Fungsi Naik Biner *Threshold*

## 3. Fungsi Bipolar *Symmetric Hard Limit*

Fungsi bipolar *symmetric hard limit* mempunyai *output* yang bernilai 1, 0, dan -1 gambar 2.8 secara matematis, fungsi *symmetric hard limit* dituliskan sebagai berikut.

$$y = \begin{cases} 1, & \text{jika } x > 0 \\ 0, & \text{jika } x = 0 \\ -1, & \text{jika } x < 0 \end{cases} \quad (2.10)$$

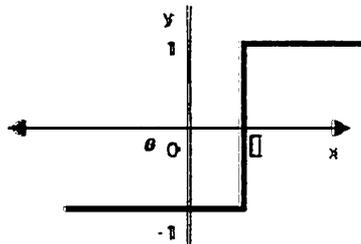


**Gambar 2.8** Fungsi Bipolar *Symetric Hard Limit*

#### 4. Fungsi Bipolar Dengan *Threshold*

Fungsi bipolar dengan *threshold* mempunyai *output* yang bernilai 1, 0, atau -1 gambar 2.9 untuk batas nilai ambang  $\theta$  tertentu. Secara matematis, fungsi bipolar dengan *threshold* dituliskan sebagai berikut.

$$y = \begin{cases} 1, & \text{jika } x > \theta \\ 0, & \text{jika } x = \theta \\ -1, & \text{jika } x < \theta \end{cases} \quad (2.11)$$



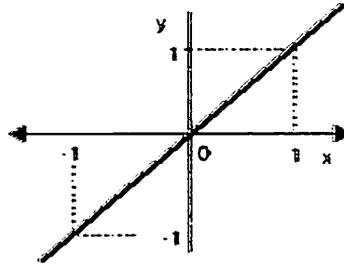
**Gambar 2.9** Fungsi Bipolar Dengan *Threshold*

5. Fungsi *Linear* (identitas)

Nilai *input* dan nilai *output* pada fungsi *linear* adalah sama gambar 2.10.

secara matematis, fungsi *linear* dituliskan sebagai berikut :

$$y = x \quad (2.12)$$



**Gambar 2.10** Fungsi *Linear* (identitas)

6. Fungsi *Saturating Linier*

Fungsi ini akan bernilai 1 jika inputnya lebih dari  $\frac{1}{2}$ . Jika nilai input terletak

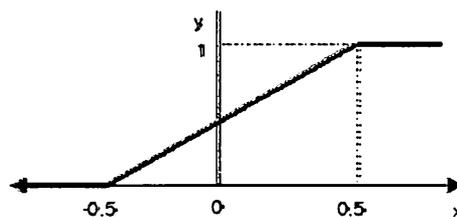
antara  $-\frac{1}{2}$  dan  $\frac{1}{2}$  maka outputnya akan bernilai sama dengan nilai input

ditambah  $\frac{1}{2}$ . Jika nilai inputnya kurang dari  $-\frac{1}{2}$  maka fungsi bernilai 0

gambar 2.11. secara matematis, fungsi *saturating linear* dituliskan sebagai

berikut.

$$y = \begin{cases} 1, & \text{jika } x \geq \frac{1}{2} \\ x + \frac{1}{2}, & \text{jika } -\frac{1}{2} > x > \frac{1}{2} \\ 0, & \text{jika } x \leq -\frac{1}{2} \end{cases} \quad (2.13)$$

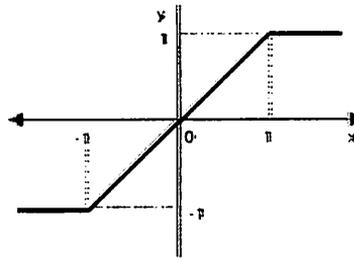


**Gambar 2.11** Fungsi *Saturating Linier*

### 7. Fungsi *Symetric Saturating Liniear*

Fungsi ini akan bernilai 1 jika input-nya lebih dari 1. Jika nilai input terletak antara -1 dan 1 maka output-nya akan bernilai sama dengan nilai input-nya. Sedangkan jika input-nya kurang dari -1 maka fungsi akan bernilai -1 gambar 2.12. Secara matematis, fungsi *symetric saturating linear* dituliskan sebagai berikut :

$$y = \begin{cases} 1, & \text{jika } x \geq 1 \\ x, & \text{jika } -1 \leq x \leq 1 \\ -1, & \text{jika } x \leq -1 \end{cases} \quad (2.14)$$



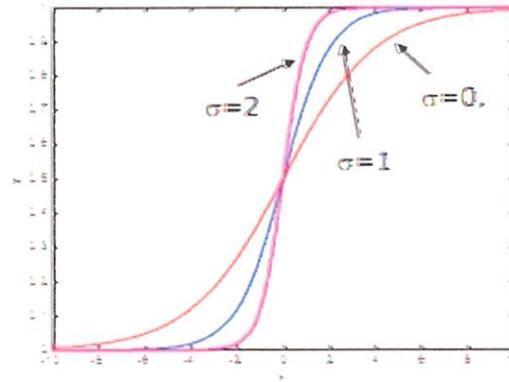
**Gambar 2.12** Fungsi *Symetric Saturating Liniear*

### 8. Fungsi *Sigmoid Biner*

Biasanya fungsi ini digunakan untuk JST yang dilatih menggunakan metode *backpropagation*. JST yang membutuhkan nilai output yang terletak pada interval 0 sampai 1 sering kali menggunakan fungsi *sigmoid biner* karena fungsi ini memiliki nilai pada range 0 sampai 1 gambar 2.13. Secara matematis, fungsi *sigmoid biner* dituliskan sebagai berikut :

$$y = f(x) = \frac{1}{1+e^{-\sigma x}} \quad (2.15)$$

Dengan :  $f'(x) = \sigma f(x)[1 - f(x)]$  (2.16)



**Gambar 2.13** Fungsi Sigmoid Biner

#### 9. Fungsi Sigmoid Bipolar

Output dari fungsi *sigmoid bipolar* memiliki range antara 1 sampai -1 gambar 2.14. Secara matematis, fungsi *sigmoid bipolar* dirumuskan sebagai berikut :

$$y = f(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (2.17)$$

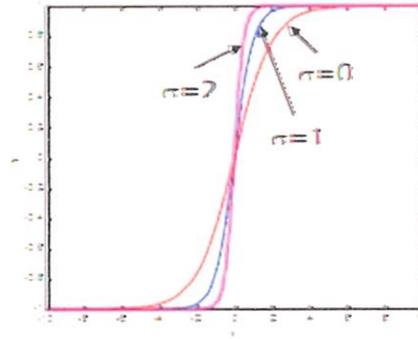
dengan :  $f'(x) = \frac{\sigma}{2} [1 + f(x)][1 - f(x)]$  (2.18)

Fungsi hampir sama dengan fungsi *hyperbolic tangent*. Keduanya memiliki range antara -1 sampai 1. secara matematis, fungsi *hyperbolic tangent* dituliskan sebagai berikut :

$$y = f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.19)$$

atau  $y = f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$  (2.20)

dengan:  $f'(x) = [1 + f(x)][1 - f(x)]$  (2.21)



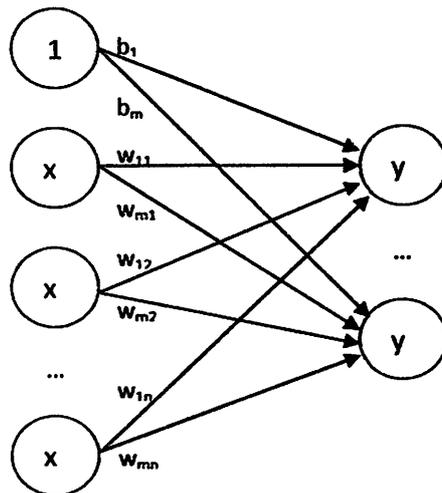
**Gambar 2.14** Fungsi *Sigmoid Bipolar*

### 2.5.3. Arsitektur Jaringan

Pola dimana *neuron-neuron* pada JST disusun berhubungan erat dengan algoritma belajar yang digunakan untuk melatih jaringan. Secara umum, kita bisa membagi arsitektur jaringan menjadi empat, yaitu:

1. Jaringan layar tunggal (*single-layer Feedforward networks*)

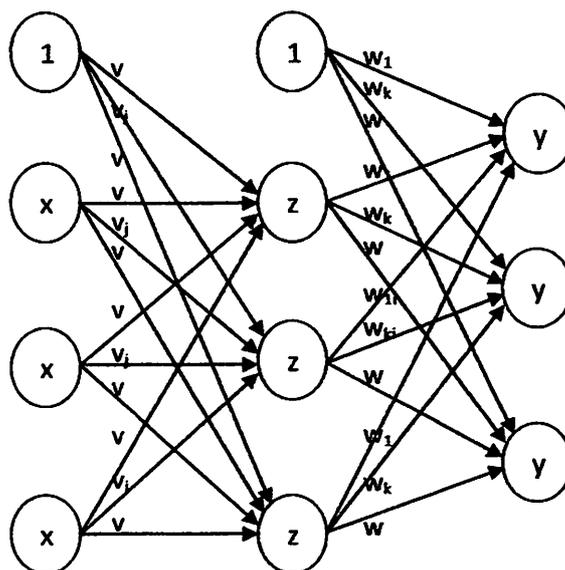
Jaringan dengan lapisan tunggal terdiri dari satu layar *input* dan satu layar *output*. Setiap *neuron/unit* yang terdapat pada lapisan/layar *input* selalu terhubung dengan setiap neuron yang terdapat pada lapisan/layar *output*. Seperti pada gambar 2.15.



**Gambar 2.15 Jaringan Layer Tunggal (Single-layer Feedforward Networks)**

## 2. Jaringan layer jamak (*multi-layer Feedforward networks*)

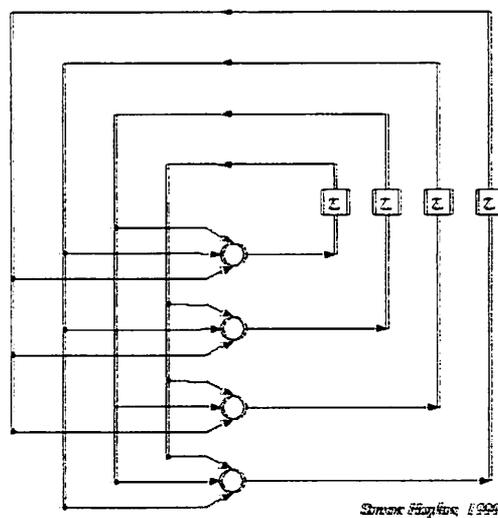
Jaringan dengan layer jamak memiliki layer input, layer tersembunyi (*hidden layer*), dan layer *output*. Jaringan dengan layer jamak mampu menyelesaikan permasalahan yang lebih kompleks dibandingkan dengan jaringan layer tunggal, namun proses pelatihan membutuhkan waktu lama. seperti pada gambar 2.16.



**Gambar 2.16 Arsitektur Jaringan Layer Jamak.**

### 3. *Recurrent Network*

*Recurrent neural network* adalah jaringan yang mempunyai minimal satu *feedback loop*. Sebagai contoh, suatu *recurrent network* bisa terdiri dari satu lapisan *neuron* tunggal dengan masing-masing *neuron* memberikan kembali outputnya sebagai *input* pada semua *neuron* yang lain, seperti diilustrasikan oleh gambar 2.17. Dalam struktur yang digambarkan oleh gambar ini, tidak ada *self-feedback loops* dalam jaringan.



**Gambar 2.17** *Recurrent Network* Tanpa *Self-feedback Loop* dan Tanpa *Hidden Neuron*

#### 2.5.4. Backpropagation

Model *BackPropagation* (perambatan balik) merupakan algoritma pembelajaran / pelatihan dengan :

1. Banyak sel dalam jaringan.
2. Banyak lapisan dalam jaringan.

3. Menggunakan *error* output untuk merubah nilai bobot
4. Perambatan *forward*
5. Perambatan *backward*

Adapun fungsi aktivasi dari model *Backpropagation* ini terdiri dari :

*Sigmoid biner*

$$y = f(x) = \frac{1}{1+e^{-ax}} \quad (2.22)$$

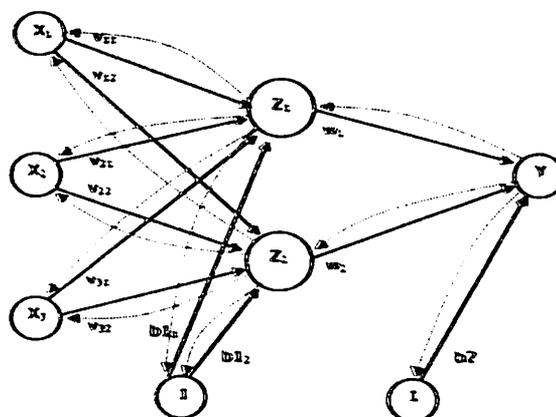
*Sigmoid bipolar*

$$y = f(x) = \frac{1-e^{-x}}{1+e^{-x}} \quad (2.23)$$

Fungsi aktivasi linier

$$y = f(x) = x \quad (2.24)$$

Arsitektur dari jaringan *Backpropagation* diperlihatkan dalam gambar, seperti pada gambar 2.18.



**Gambar 2.18** Arsitektur jaringan *Backpropagation*

Seperti yang diperlihatkan pada gambar diatas bahwa jaringan *Backpropagation* terdiri dari 3 sel *neuron* pada lapisan *input* yaitu  $x_1$ ,  $x_2$ , dan  $x_3$ ; 2 sel *neuron* pada lapisan tersembunyi yaitu  $z_1$  dan  $z_2$ ; serta 1 sel *neuron* pada lapisan *output* yaitu  $y$ . Nilai bias  $b_1$  yang di berikan pada lapisan tersembunyi bertujuan untuk mengolah data *input* ditambah bobot  $V_{ij}$  yang masuk ke dalam sel-sel pada lapisan tersembunyi dengan bantuan fungsi aktifasi. Begitupula dengan nilai bias  $b_2$  yang diberikan pada lapisan keluaran adalah untuk mengolah data yang berasal dari keluaran sel pada lapisan tersembunyi ditambah bobot  $W_{ij}$  yang masuk kedalam lapisan keluaran dengan bantuan fungsi aktifasi.

#### Algoritma Backpropagation

1. *Inisialisasi* bobot (*random* yang kecil, -0.5 s/d 0.5 atau -1 s/d 1).
2. Tetapkan: maks *Epoch*, target error, & *learning rate*  $\alpha$
3. *Inisialisasi*:  $Epoch=0$ ,  $MSE = 1$
4. Berikan harga-harga masukan dan keluaran target yang menjadi paket pelatihan.

$$X_i = [x_1 \ x_2 \ x_3, \dots, \ x_n]^T \quad (2.25)$$

$$T_k = [t_1 \ t_2 \ t_3, \dots, \ t_n]^T \quad (2.26)$$

#### Feedforward

Hitung jumlah data yang masuk ke lapisan tersembunyi

$$Z_i = b_{1j} + \sum_{i=1}^n X_i - V_{ij} \quad (2.27)$$

Hitung keluaran dari sel lapisan tersembunyi dengan menggunakan fungsi aktivasi yang telah ditetapkan

$$Z_j = f(Z_i) \quad (2.28)$$

Hitung jumlah data yang masuk ke lapisan keluaran

$$Y_i = b_{2k} + \sum_{i=1}^n X_i - W_{jk} \quad (2.29)$$

Hitung sinyal keluaran dari sel lapisan *output* dengan menggunakan fungsi aktivasi yang telah ditetapkan

$$Y_k = f(Y_i) \quad (2.30)$$

### *Backpropagation*

Hitung informasi error dari sel pada lapisan output dengan menggunakan data target yang telah diterima.

$$\text{Error sel pada lapisan } output \partial z_k = (t_k - y_k) \cdot f'(y_{k,a}) \quad (2.31)$$

$$\text{Error sel pada lapisan } hidden-output \varphi_{2_{jk}} = \partial z_k \cdot Z_{j,a} \quad (2.32)$$

$$\text{Error pada sel bias output } \beta_{2_k} = \partial z_k \cdot 1 \quad (2.24)$$

Hitung koreksi bobot dan bias , pada tiap sel lapisan *hidden-output*

$$\Delta W_{jk} = \alpha \cdot \varphi_{2_{jk}} \quad (2.25)$$

$$\Delta b_{2_k} = \alpha \cdot \beta_{2_k} \quad (2.26)$$

Hitung jumlah data yang masuk dari setiap sel lapisan *output* ke lapisan hidden.

$$\partial_{inj} = \sum_{k=1}^m \partial z_k \cdot W_{jk} \quad (2.27)$$

Hitung informasi *error*-nya

$$\text{Error sel pada lapisan } \textit{hidden} \partial 1_j = \partial_{inj} \cdot f(Z_{j,a}) \quad (2.28)$$

$$\text{Error sel pada lapisan } \textit{input-hidden} \varphi 1_y = \partial 1_j \cdot X_i \quad (2.29)$$

$$\text{Error sel pada bias } \textit{output} \beta 2_j = \partial 1_j \cdot 1 \quad (2.30)$$

Hitung koreksi bobot dan bias, dari tiap sel pada lapisan *input-hidden*.

$$\Delta b 1_j = \alpha \cdot \quad (2.31)$$

$$\Delta V_{ij} = \alpha \cdot \beta 1_j \quad (2.32)$$

Perbaiki bobot dan bias pada lapisan *hidden-output*.

$$W_{jk(\textit{baru})} = W_{jk(\textit{lama})} + \Delta W_{jk} \quad (2.33)$$

$$b 2_{k(\textit{baru})} = b 2_{k(\textit{lama})} + \Delta b 2_k \quad (2.34)$$

Perbaiki bobot dan bias pada lapisan *input-hidden*.

$$V_{ij(\textit{baru})} = V_{ij(\textit{lama})} + \Delta V_{ij} \quad (2.35)$$

$$b 1_{j(\textit{baru})} = b 1_{j(\textit{lama})} + \Delta b 1_j \quad (2.36)$$

Disamping itu kelebihan – kelebihan yang ada mengenai metode backpropagation ini adalah :

1. Mempunyai sistem pembelajaran sehingga dapat selalu menambah kehandalan dari kemampuannya.

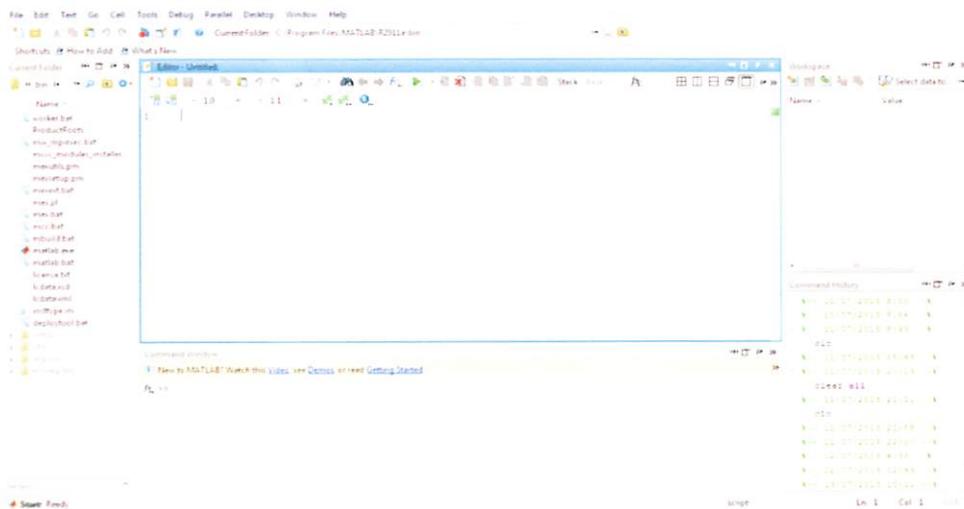
2. Perangkat yang mampu untuk mengenali suatu objek non-linier.
3. Mampu melakukan pengabdatasan terhadap suatu objek.
4. Perangkat yang memiliki toleransi terhadap suatu kesalahan dalam pengenalan suatu objek.
5. *Neural network* yang mampu diimplementasikan terhadap *hardware* atau perangkat keras.
6. Perangkat yang mampu diimplemenasikan secara parallel (dapat digunakan untuk berbagai macam objek).[3]

## 2.6. Matlab

Matlab merupakan bahasa pemrograman yang hadir dengan fungsi dan karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada lebih dahulu seperti Delphi, Basic maupun C++. Matlab merupakan bahasa pemrograman *level* tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan.

Matlab hadir dengan membawa warna yang berbeda. Hal ini karena matlab membawakeistimewaan dalam fungsi-fungsi matematika, fisika, statistik, dan visualisasi. Matlab dikembangkan oleh *MathWorks*, yang pada awalnya dibuat untuk memberikan kemudahan mengakses data matrik pada proyek *LINPACK* dan *EISPACK*. Saat ini matlab memiliki ratusan fungsi yang dapat digunakan sebagai *problem solver* mulai dari simple sampai masalah-masalah yang kompleks dari

berbagai disiplin ilmu. Gambar 2.19 ini menampilkan tampilan antar muka dari matlab versi 7.12.0 (Matlab R2011a). [4]



**Gambar 2.19** Matlab Versi 7.12.0 (Matlab R2011a).

## BAB III

### PERANCANGAN DAN ANALISA SISTEM

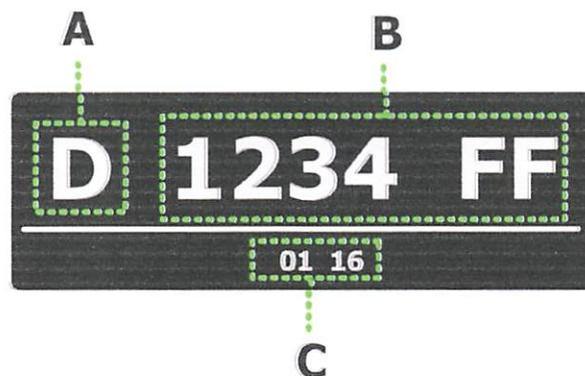
#### 3.1. Citra Plat Nomor

Plat nomor kendaraan selalu dibuat seragam atau sama antara satu jenis kendaraan dengan kendaraan lain, baik dalam ukuran plat, font pada plat nomor pun selalu sejenis. Font yang digunakan untuk plat nomor yaitu *License plated* dengan ukuran font yang sama juga.

##### 3.1.1. Spesifikasi teknis

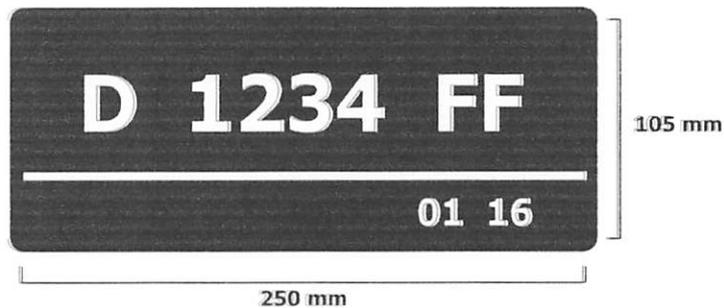
Tanda Nomor Kendaraan Bermotor berbentuk plat aluminium dengan cetakan tulisan dua baris.

- A. Kode wilayah (huruf) dan kode/seri akhir wilayah (huruf).
- B. Nomor polisi.
- C. Baris kedua menunjukkan bulan dan tahun masa berlaku.

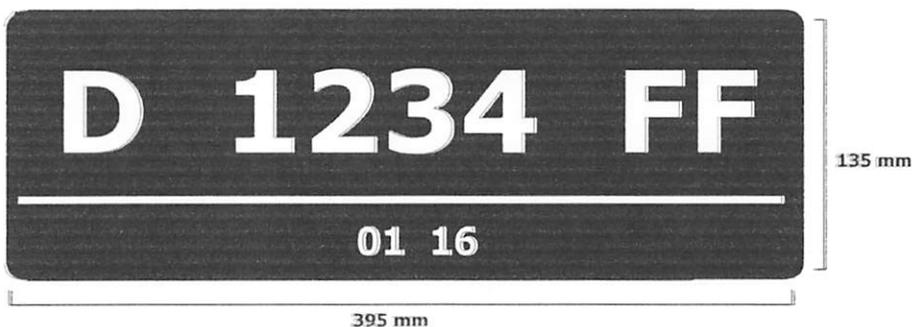


**Gambar 3.1** Tanda Nomor Kendaraan Bermotor (TNKB)

Bahan baku TNKB adalah aluminium dengan ketebalan 1 mm. Ukuran TNKB untuk kendaraan bermotor roda 2 dan roda 3 adalah 250x105 mm, sedangkan untuk kendaraan bermotor roda 4 atau lebih adalah 395x135 mm. Terdapat cetakan garis lurus pembatas lebar 5 mm di antara ruang nomor polisi dengan ruang angka masa berlaku (yang lama), sedangkan yang baru terdapat garis putih di sekitar TNKB dan tidak ada batas pemisah antara nomor polisi dan masa berlaku (dari tahun 2011). Berikut merupakan gambaran plat Nomor kendaraan bermotor roda 2 atau 3 dan kendaraan bermotor roda 4 :



**Gambar 3.2** TNKB Kendaraan bermotor roda 2 atau 3



**Gambar 3.3** TNKB Kendaraan bermotor roda 4 atau lebih

Pada sudut kanan atas dan sudut kiri bawah terdapat tanda khusus (*security mark*) cetakan lambang Polisi Lalu Lintas; sedangkan pada sisi sebelah kanan

dan sisi sebelah kiri ada tanda khusus cetakan "DITLANTAS POLRI" (Direktorat Lalu Lintas Kepolisian RI) yang merupakan hak paten pembuatan TNKB oleh Polri dan TNI.

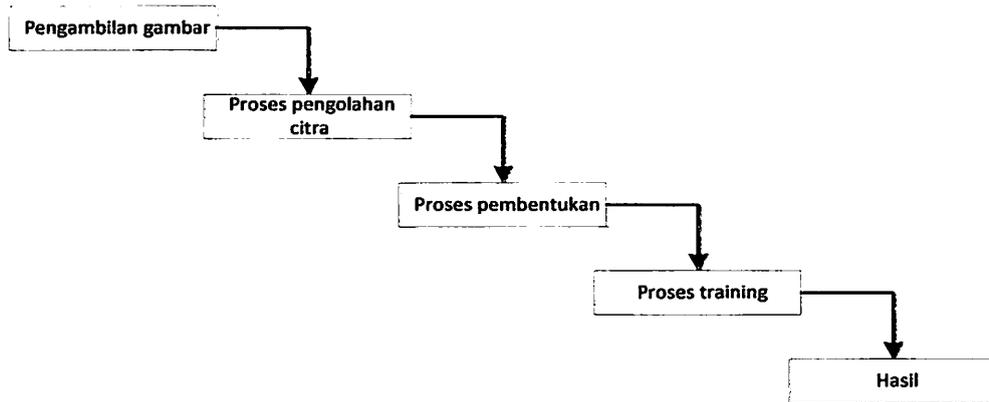
### 3.2. Perancangan Blok Diagram Pada Aplikasi

Pada aplikasi penerapan *optical character recognition* terdapat alur yang akan memproses aplikasi, agar aplikasi dapat memproses penerapan *optical character recognition* pada plat nomor kendaraan.

Pada blok diagram pada aplikasi gambar 3.4. dapat dijelaskan sebagai berikut:

1. Proses awal yaitu proses pengambilan citra yang menampilkan citra plat nomor pada aplikasi.
2. Setelah itu lakukan pemotongan (*cropping*) pada area nomor yang perlu di terjemahkan.
3. Setelah proses pemotongan dapat dilanjutkan langsung pengolahan yang terdapat pada aplikasi ini sampai *recognition*.
4. Proses selanjutnya yaitu *backpropagation* untuk membaca pola plat nomor yang sudah terdeteksi.
5. Setelah sudah melalui prose, kemudian akan tercetak hasil dengan bentuk teks yang sesuai dengan pembacaan pola.

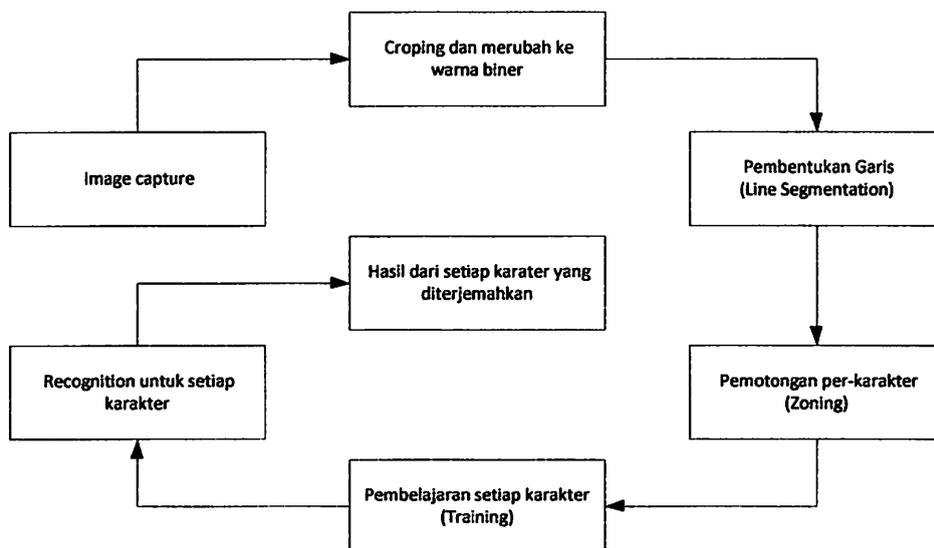
Langkah yang ada pada aplikasi penerapan *optical character recognition* dijelaskan pada gambar 3.4.



**Gambar 3.4** Blok Diagram Aplikasi

### 3.3. Perancangan Blok Diagram Optical Character Recognition

Pada penerapan *optical character recognition* untuk plat nomor kendaraan, terdapat adanya proses *image acquisition*, *pre-processing*, *document and analysis*, *feature extraction*, *training and recognition* dan *post processing* pada citra. Diagram alir dalam penerapan *optical character recognition* untuk plat nomor kendaraan dapat dilihat pada gambar 3.5.



**Gambar 3.5** Blok Diagram Proses *Optical Character Recognition*

Pada blok diagram proses OCR gambar 3.5 dapat dijelaskan sebagai berikut:

1. *Image acquisition* yaitu proses untuk merubah citra asli menjadi citra digital dengan cara *scanning* pada citra.
2. Dalam *image acquisition* terdapat proses *image scanning*, *image scanning* adalah gambar yang sudah dijadikan digital agar bisa diproses.
3. Kemudian proses selanjutnya citra yang sudah menjadi citra digital diperbaiki dengan proses *pre processing*, pada *pre processing* terdapat *noise removal*, *sharpening*, *skew*.
4. *Cropping* adalah proses pemotongan pada citra, *thresholding* merubah citra *grayscale* menjadi *biner*.
5. Proses selanjutnya yaitu *Document And Analysis* yang mendeteksi karakter citra agar dapat terdeteksi dengan benar, sehingga citra dapat dikenali.
6. *Line segmentation* adalah untuk mendeteksi perbaris citra yang sudah terdeteksi.
7. *Character recognition* untuk mendeteksi citra pada setiap huruf yang terdapat pada karakter citra yang terdeteksi.
8. Pada proses *feature extraction* citra yang akan terekstrak akan dikumpulkan kembali, sehingga tersusun karakter yang sudah terdeteksi proses tersebut juga disebut *zoning*.
9. Lanjut pada proses *training*, pada proses ini karakter akan diproses dengan mencocokkan citra karakter yang sudah terdeteksi.
10. Bila pencocokan citra karakter sudah benar, maka proses selanjutnya mengklasifikasi citra dengan cara proses *recognition*.

11. Pengklasifikasi adalah mengelompokkan karakter pada nomor yang sudah mengalami proses pembacaan karakter.
12. Setelah citra karakter cocok setelah melalui proses pencocokan dan pengklasifikasian citra karakter, maka citra akan ditampilkan menggunakan proses *post-processing*.

### **3.4. Perancangan Flowchart**

Perancangan suatu aplikasi harus berdasarkan pada perancangan flowchart yang akan menjadi acuan agar suatu aplikasi dapat dibuat dengan konsep yang sudah ada. Agar perancangan sistem yang dibuat berhasil dan berjalan sesuai dengan keinginan yang ada pada sistem yang sudah dirancang.

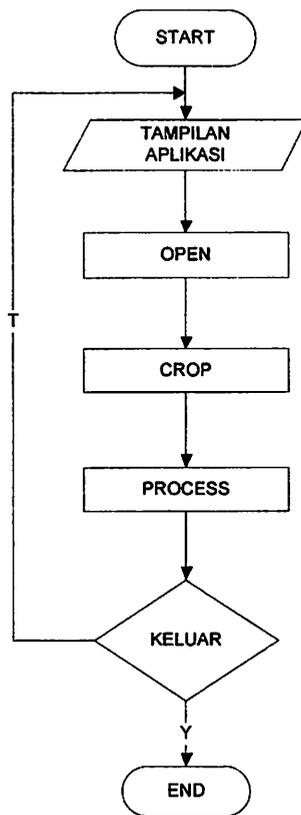
#### **3.4.1. Flowchart Aplikasi**

Pada tahap perancangan aplikasi penerapan *optical character recognition* untuk plat nomor kendaraan. Saat dimulainya aplikasi dengan pengambilan gambar, proses pada aplikasi.

Flowchart aplikasi pada gambar 3.6 dapat dijelaskan sebagai berikut:

1. Mulai dari tampilan aplikasi yang terdapat berbagai tombol proses.
2. Setelah itu memilih tombol *open*, maka akan keluar gambar plat nomor pada kendaraan yang sudah menjadi digital.
3. Setelah itu memilih tombol *crop*, maka akan tampil gambar plat nomor yang sudah terpotong.
4. Jika gambar plat nomor sudah terpotong, setelah itu memilih tombol *execute* untuk menampilkan hasil proses nomor menjadi teks.

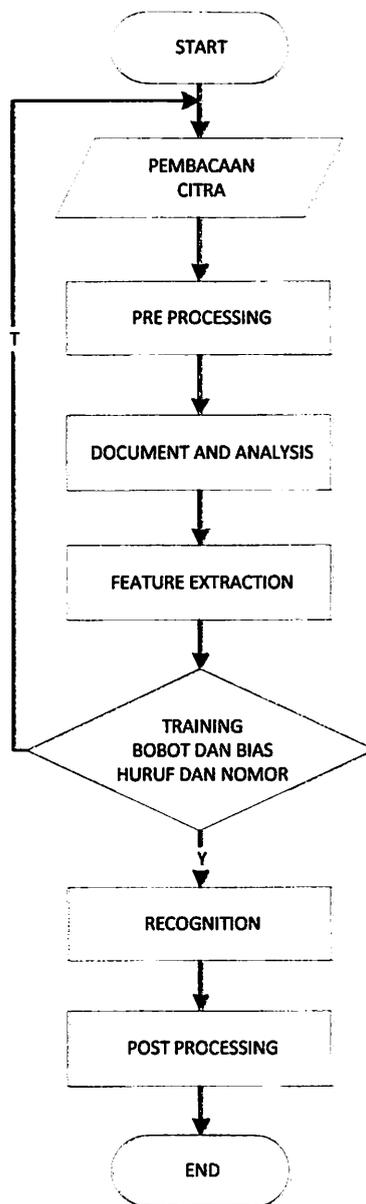
Langkah yang ada pada flowchart aplikasi penerapan *optical character recognition* dijelaskan pada gambar 3.6.



**Gambar 3.6** Flowchart Perancangan Aplikasi

### 3.4.2. Flowchart OCR

Pada tahap perancangan OCR pada penerapan *optical character recognition* untuk plat nomor pada kendaraan. Perancangan OCR pada aplikasi dijelaskan pada gambar 3.7.



**Gambar 3.7** Flowchart OCR

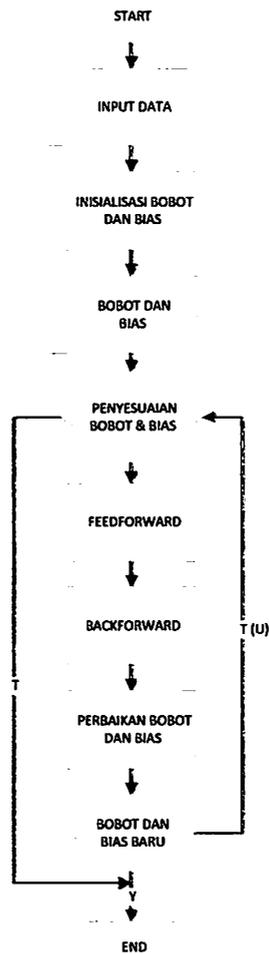
Pada flowchart OCR gambar 3.7 dapat dijelaskan sebagai berikut:

1. Mulai untuk pembacaan karakter pada citra.
2. Kemudian masuk proses *preprocessing*, agar kualitas citra karakter dapat diperbaiki.

3. Setelah itu proses *document and analysis* yang akan men-*segmentasi* karakter perbaris dan perkolomnya, agar karakter dapat terbaca secara jelas.
4. Pada proses selanjutnya *training*, pada proses ini karakter yang sudah ter-*segmentasi* akan dibaca setiap huruf pada karakter tersebut.
5. Jika pembacaan selesai dan benar, maka dikelompokan dengan proses *recognition* agar huruf yang sudah terbaca dapat menjadi karakter dengan benar.
6. Setelah melalui proses *training and recognition*, kemudian karakter akan dicetak sebagai teks biasa dengan proses *post processing*.

#### **3.4.3. Flowchart Metode *Backpropagation***

Pada tahap perancangan aplikasi penerapan *optical character recognition* untuk plat nomor kendaraan. Perancangan metode learning backpropagation untuk training pada sistem aplikasi OCR. Dapat dilihat pada gambar 3.8.



**Gambar 3.8** Flowchart metode learning backpropagation

Pada flowchart metode *learning backpropagation* gambar 3.8 dapat dijelaskan sebagai berikut:

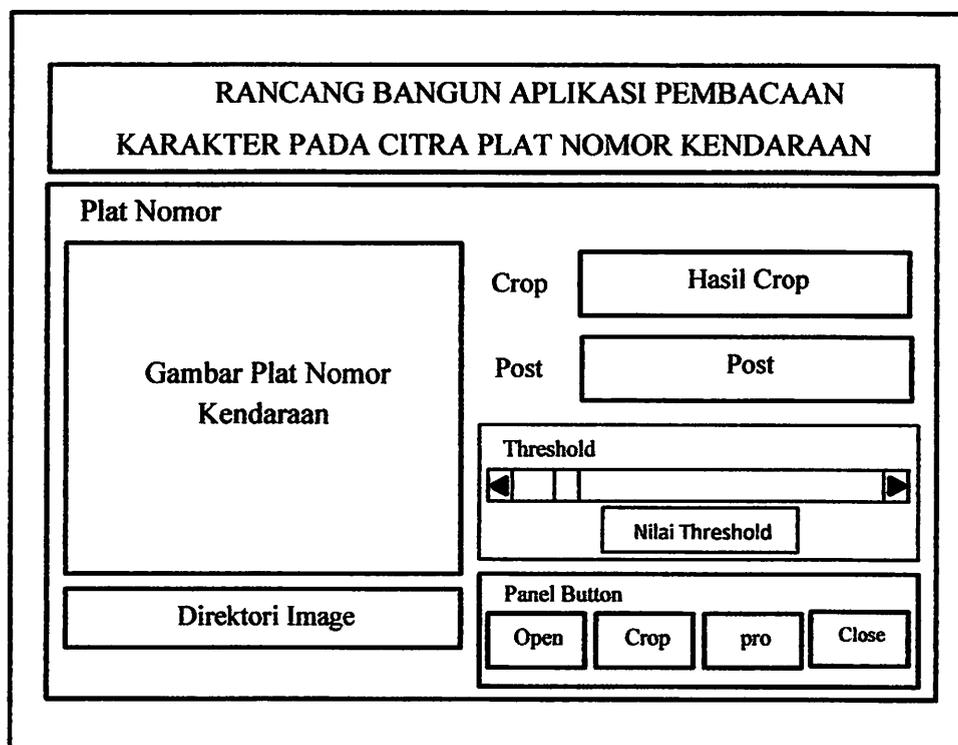
1. Mulai untuk sistem training pada pembacaan karakter.
2. *Input data training* adalah memasukkan data karakter yang sudah terdeteksi agar dapat terbaca.
3. Kemudian diproses pada inialisasi bobot dan bias.
4. Setelah itu meng-*input* bobot dan bias karakter yang akan dibaca.

5. Penyesuaian pada bobot dan bias akan menyesuaikan bobot dan bias pada karakter.
6. Jika penyesuaian bobot dan bias tidak dilaksanakan, maka akan langsung berhenti.
7. Jika penyesuaian bobot dan bias dilaksanakan, maka pembacaan karakter akan dilanjutkan pada proses selanjutnya.
8. Dilanjutkan pada proses umpan maju (*feed forward*) selama propagasi maju, nilai *input* ( $x$ ) dan *output* dari setiap unit lapisan tersembunyi ( $Z_{ji}$ ) akan dipropagasikan ke lapisan tersembunyi menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan nilai *output* pada jaringan ( $y$ ).
9. Kemudian dilanjutkan umpan mundur (*back forward*) berdasarkan kesalahan  $t_k - y_k$ , dihitung faktor  $\delta_k = (k = 1, 2, 3, \dots, m)$  yang dipakai untuk mendistribusikan kesalahan  $y_k$  ke semua unit tersembunyi yang terhubung langsung dengan  $y_k$ .
10. Selanjutnya proses untuk perbaikan bobot dan bias pada karakter.
11. Setelah perbaikan bobot dan bias karakter, maka karakter yang baru akan menjadi bobot dan bias baru.
12. Pada proses bobot dan bias karakter baru ini akan terjadi proses perulangan yang akan membaca karakter terus menerus sehingga karakter yang terbaca sesuai dengan karakter yang terdeteksi.

13. Setelah pembacaan karakter sudah selesai dan sesuai dengan karakter yang dimasukkan, maka akan proses akan selesai.

### 3.5. Perancangan Layout Aplikasi

Perancangan tampilan aplikasi penerapan *optical character recognition* untuk plat nomor kendaraan, seperti ditunjukkan pada gambar 3.9.



**Gambar 3.9** Perancangan Tampilan Aplikasi

Pada perancangan tampilan aplikasi gambar 3.9 dapat dijelaskan sebagai berikut:

1. Terdapat judul pada tampilan aplikasi.
2. *Gambar plat nomor kendaraan* adalah dimana gambar plat nomor akan ditampilkan.
3. *Open* adalah *button* untuk proses mengambil gambar.

4. *Direktori image* adalah tampilan direktori gambar yang muncul pada aplikasi.
5. *Hasil crop* adalah hasil dari citra plat nomor yang terpotong dan menjadi gambar *binner*.
6. *Nilai threshold* didapatkan dari *slider* di atasnya untuk menentukan *threshold* yang baik.
7. *Post* adalah hasil pembacaan setiap karakter pada plat nomor kendaraan.
8. *Button crop* adalah sistem memotong yang akan membuat area pemotongan.
9. *Button Process* adalah dimana proses yang memproses citra nomor kendaraan yang sudah terpotong.
10. Cetak nomor kendaraan adalah hasil dari proses *learning* nomor pada *post*.
11. *Close* adalah tombol keluar untuk aplikasi.

### 3.6. Perancangan Tombol

Perancangan *button* adalah perancangan pembuatan *button* pada aplikasi pada penerapan *optical character recognition* pada plat nomor kendaraan, *button* dirancang sebagai alat guna dalam menjalankan proses aplikasi sesuai fungsinya. adapun perancangan *button* sebagai berikut:

1. Tombol *open* adalah tombol yang digunakan untuk mengambil gambar plat nomor kendaraan dan menampilkan pada tempat yang diberikan dan akan diproses pada aplikasi penerapan *optical character recognition* pada plat nomor kendaraan bermotor.

2. *Slider* adalah pengaturan nilai *threshold* yang akan membersihkan kotoran yang ada pada citra sebelum merubahnya ke dalam biner.
3. *button crop* adalah *button* yang digunakan untuk membuat area pemotongan pada gambar plat nomor kendaraan dan menampilkan hasil *cropping* tersebut. Dalam *cropping* terjadi proses dijadikan ke dalam gambar biner.
4. *Button process* adalah *button* yang digunakan untuk memproses pembacaan pola huruf dan angka pada plat nomor dan menampilkan hasil dari proses dengan mencetak hasilnya yang.
5. *Button close* adalah *button* yang digunakan untuk proses keluar dari aplikasi pembacaan karakter plat nomor kendaraan.

### 3.7. Perancangan Tampilan Hasil

Perancangan tampilan hasil adalah perancangan hasil dari tampilan *button* pada aplikasi pembacaan karakter plat nomor. Tampilan hasil dirancang untuk menenpatkan hasil dari setiap *button* yang membutuhkan. adapun perancangan tampilan hasil sebagai berikut :

1. Tampilan hasil *gambar plat nomor kendaraan* adalah tampilan gambar plat nomor dari *button open* dan tampilan *gambar plat nomor* ini dibuat dari *axes* pada matlab dan terdapat pada aplikasi penerapan *optical character recognition* pada plat nomor kendaraan.
2. Tampilan hasil *direktori image* adalah tampilan nama tempat folder gambar kartu plat nomor yang diproses dari tombol *open* dan tampilan *direktori*

*image* ini dibuat dari *edit text* pada matlab dan terdapat pada aplikasi pembacaan plat nomor kendaraan.

3. Tampilan *hasil crop* adalah tampilan hasil dari seleksi pada plat yang diproses dari *button crop* menjadi citra *biner* dan tampilan *cropping* ini dibuat dari *axes* pada matlab dan terdapat pada aplikasi pembacaan karakter plat nomor kendaraan.
4. Tampilan hasil *post* adalah tampilan hasil dari proses seleksi plat nomor kemudian diproses dengan pembacaan pola pada tombol *process* tampilan *post* ini dibuat dari *text* pada matlab dan terdapat pada aplikasi pembacaan karakter pada plat nomor kendaraan.
5. Perancangan nama *tools* yang akan digunakan pada aplikasi pembacaan karakter pada plat nomor kendaraan. Perancangan nama *tools* dapat dilihat pada tabel 3.1.

**Tabel 3.1** Perancangan Nama *Tools*

Tool yang dibutuhkan	Nama Tools
Background	Figure
Judul Aplikasi	Static Text
Text <i>crop</i>	Static text
Text <i>post</i>	Static text
Kotak "plat nomor"	Panel
Kotak " <i>panel button</i> "	Panel
Kotak " <i>Threshold</i> "	Panel
Tampilan <i>output</i> Gambar plat nomor	Axes

Tampilan <i>hasil crop</i>	Axes
<i>Threshold</i>	Slider
Tampilan <i>post</i>	Text
Tampilan <i>direktori</i>	Text
Nilai <i>Threshold</i>	Text
Tombol <i>open</i>	Push Button
Tombol <i>crop</i>	Push Button
Tombol <i>process</i>	Push Button
Tombol <i>close</i>	Push Button

## BAB IV

### IMPLEMENTASI DAN PENGUJIAN

#### 4.1 Implementasi

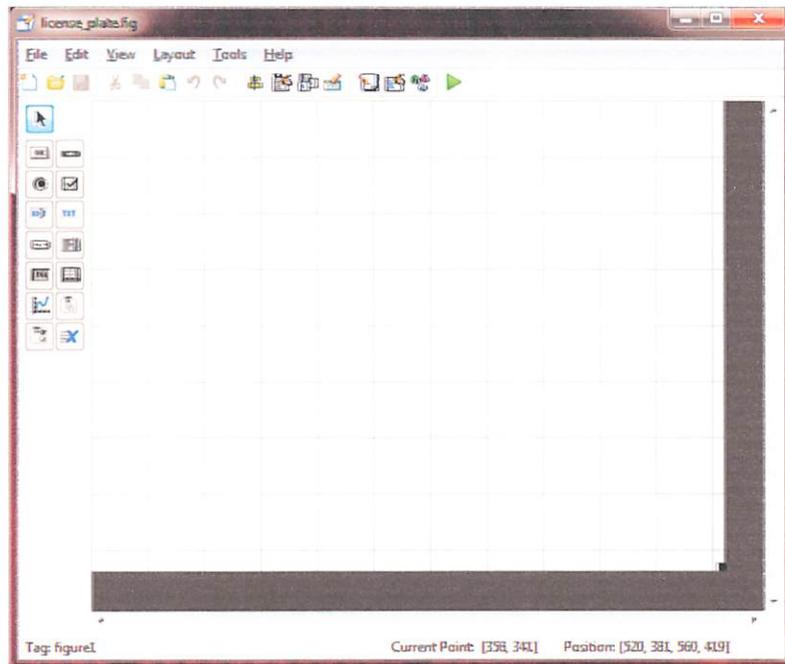
Pada bagian implementasi ini akan dilakukan proses pengambilan gambar plat nomor kendaraan. Kemudian akan dilakukan proses recognition yang akan menampilkan teks huruf dan nomor dari gambar plat nomor tersebut. Untuk menampilkan teks huruf dan angka dari plat nomor tersebut aplikasi ini menggunakan metode *learning backpropagation* agar menampilkan sesuai dari hasil pembacaan.

#### 4.2 Pembuatan Tampilan Aplikasi

Pembuatan tampilan proses untuk aplikasi pada plat nomor kendaraan. Tahapan-tahapan yang diperlukan dalam pembuatan tampilan aplikasi ini, antara lain :

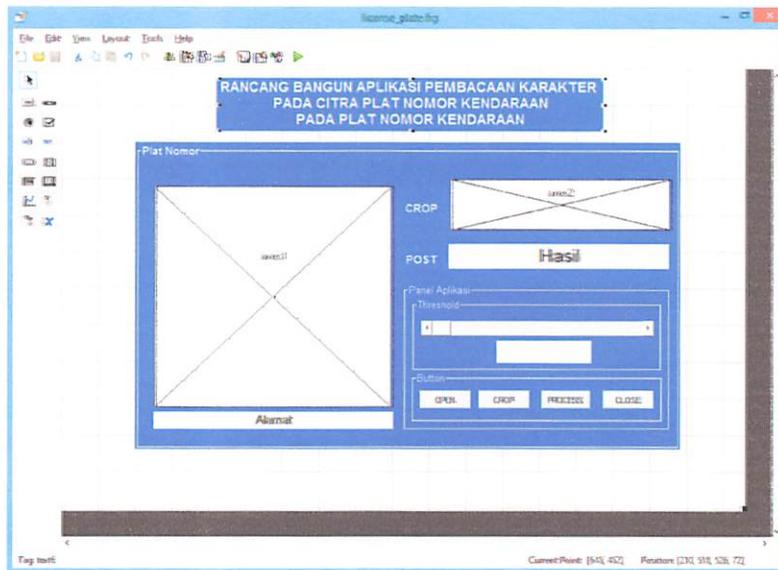
1. Menjalankan perangkat lunak *Matlab R2011a (versi 7.12.0)*.
2. Membuat *figure* baru dengan nama "*license plate.fig*" agar dapat digunakan pemrograman *Matlab*.
3. Melakukan konfigurasi pada dokumen yang masih kosong, dengan menampilkan proses awal. Proses pembuatan *figure* tampilan aplikasi

pembacaan karakter pada plat nomor kendaraan. Proses tersebut dapat dilihat pada gambar 4.1.



**Gambar 4.1** Tampilan Awal Pembuatan *Figure*

Setelah membuat proses awal pada pembuatan aplikasi pembacaan karakter pada plat nomor kendaraan. Kemudian membuat tampilan yang sudah terdapat desain aplikasi pembacaan plat nomor kendaraan bermotor. Tampilan yang sudah terdesain bisa dilihat pada gambar 4.2.



**Gambar 4.2** Tampilan Design Pembacaan Karakter  
Pada Plat Nomor Kendaraan

Dalam setiap desain yang ada pada aplikasi pembacaan karakter pada plat nomor, terdapat berbagai jenis *tools* yang terdapat pada *figure*. Sehingga pada tampilan terlihat menarik dan lebih jelas untuk mengetahui proses penerapan. *Tools* pada tampilan aplikasi pembacaan karakter pada plat nomor kendaraan dapat dilihat pada tabel 4.1.

**Tabel 4.1** Perancangan Nama *Tools*

Tool yang dibutuhkan	Nama Tools
Background	Figure
Judul Aplikasi	Static Text
Text <i>crop</i>	Static text
Text <i>post</i>	Static text
Kotak “plat nomor”	Panel
Kotak “ <i>panel button</i> ”	Panel

Kotak " <i>Threshold</i> "	Panel
Tampilan <i>output</i> Gambar plat nomor	Axes
Tampilan <i>hasil crop</i>	Axes
<i>Threshold</i>	Slider
Tampilan <i>post</i>	Text
Tampilan <i>direktori</i>	Text
Nilai <i>Threshold</i>	Text
Tombol <i>open</i>	Push Button
Tombol <i>crop</i>	Push Button
Tombol <i>process</i>	Push Button
Tombol <i>close</i>	Push Button

### 4.3 Tampilan Aplikasi

Tampilan aplikasi untuk proses pembacaan karakter pada plat nomor ini dibuat berdasarkan perancangan yang sudah ada dan sudah terbentuk. Tampilan aplikasi pembacaan karakter pada plat nomor kendaraan ini terbuat dari *figure* yang terdapat pada matlab. *Figure* aplikasi ini bisa disebut juga GUI (*graphical user interface*) yang menjadi *interface* untuk aplikasi satu-satunya pada pembacaan karakter pada plat nomor. Tampilan aplikasi dapat dilihat pada gambar 4.3.



**Gambar 4.3** Tampilan Aplikasi Pembacaan Karakter Pada Plat nomor Kendaraan

#### 4.4 Proses Aplikasi Penerapan OCR

Proses pada aplikasi penerapan *optical character recognition* pada plat nomor kendaraan ini menjelaskan proses jalannya program *optical character recognition* untuk menerapkan huruf dan angka pada plat nomor kendaraan bermotor

##### 4.4.1 Tampilan Input Plat Nomor

Tampilan *input* plat nomor pada aplikasi penerapan *optical character recognition* pada plat nomor kendaraan ini untuk menampilkan pengambilan proses gambar plat. Selain menampilkan gambar plat nomor, juga menampilkan direktori gambar plat nomor. Tampilan proses *input* gambar plat nomor dapat dilihat pada gambar 4.4.



**Gambar 4.4** Tampilan *Input Image* ke Dalam Aplikasi

#### 4.4.2 Tampilan Proses Cropping

Tampilan proses *cropping* pada aplikasi pembacaan karakter pada plat nomor kendaraan ini untuk menyeleksi huruf dan angka. Setelah huruf dan angka terseleksi, maka citra huruf dan angka pada plat akan diproses menjadi citra plat nomor. Tampilan proses *cropping* pada aplikasi ini dapat dilihat pada gambar 4.5.



**Gambar 4.5** Tampilan *Cropping Image*

#### 4.4.3 Tampilan Proses Binerisasi

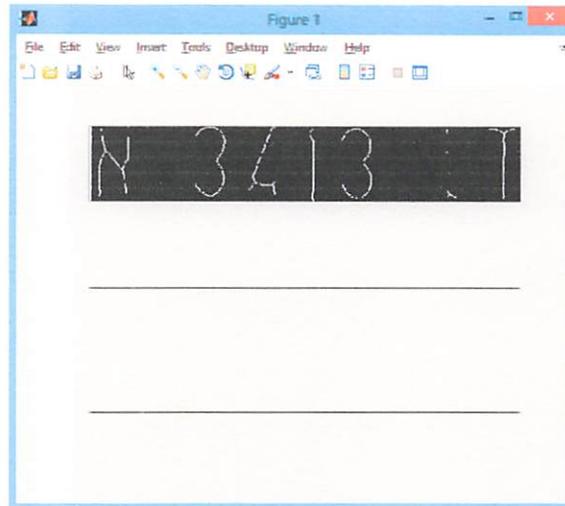
Tampilan proses *binerisasi* pada aplikasi pembacaan karakter pada plat nomor ini merubah gambar plat nomor yang *tercroping* menjadi *biner*. Tampilan proses *biner* pada aplikasi ini dapat dilihat pada gambar 4.6.



Gambar 4.6 Tampilan Proses Binerisasi

#### 4.4.4 Tampilan Line Segmentation

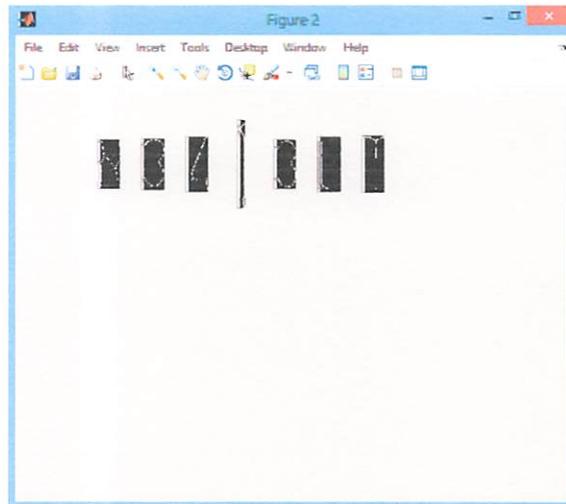
Tampilan *line segmentation* pada aplikasi pembacaan karakter pada plat nomor ini untuk mendeteksi pola huruf dan angka. Tampilan proses *line segmentation* pada aplikasi ini dapat dilihat pada gambar 4.7.



**Gambar 4.7** Tampilan *Line Segmentation*

#### 4.4.5 Tampilan Character Segmentation

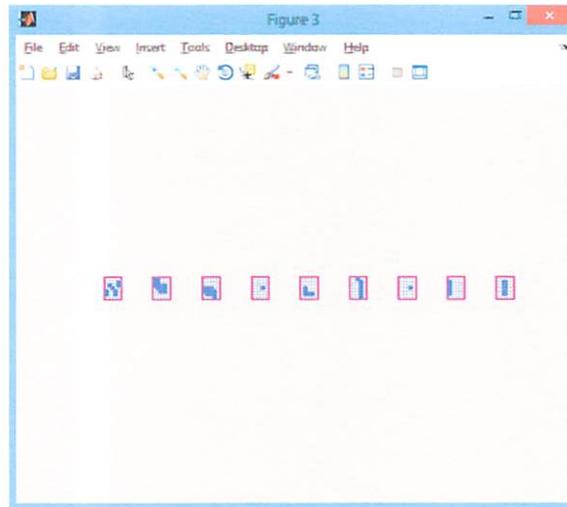
Tampilan proses *character segmentation* pada aplikasi penerapan *optical character recognition* pada pembacaan plat nomor kendaraan ini untuk mendeteksi pola huruf dan nomor per karakter pada plat nomor tersebut. Tampilan proses *chracter segmentation* pada aplikasi ini dapat dilihat pada gambar 4.8.



**Gambar 4.8** Tampilan *Character Segmentation*

#### 4.4.6 Tampilan Pembacaan Pola

Tampilan Pembacaan pola dalam penerapan *optical character recognition* pada plat nomor kendaraan ini membaca pola citra huruf dan angka pada plat. Tampilan pembacaan pola ini dapat dilihat pada gambar 4.9.



**Gambar 4.9** Tampilan Pembacaan Pola

#### 4.4.7 Tampilan Proses Learning Bacpropagation

Tampilan proses *learning backpropagation* pada aplikasi pembacaan karakter pada plat nomor ini proses. Tampilan proses aplikasi ini dapat dilihat pada gambar 4.10.

```
word =  
N  
  
word =  
N3  
  
word =  
N34  
  
word =  
N341  
  
word =  
N3413  
  
word =  
N3413L  
  
word =  
N3413LT
```

**Gambar 4.10** Tampilan *Learning Backpropagation*

#### 4.4.8 Tampilan Process

Tampilan *process* pada aplikasi pembacaan karakter pada plat nomor kendaraan ini proses akhir untuk membaca citra huruf dan nomor. Proses akhir ini terdapat proses *learning backpropagation* yang akan membaca pola citra huruf

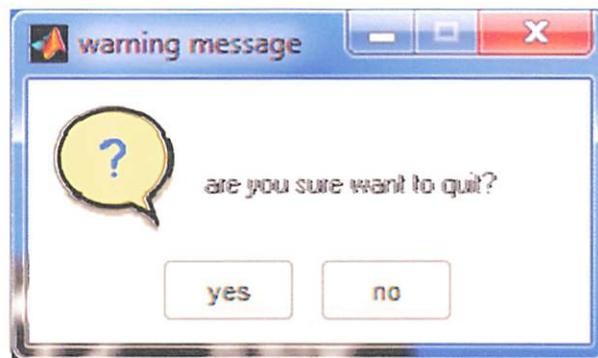
dan nomor menjadi teks. Tampilan *process* pada aplikasi dapat dilihat pada gambar 4.11



**Gambar 4.11** Tampilan Proses Aplikasi

#### 4.4.9 Tampilan Keluar Aplikasi

Tampilan keluar pada aplikasi pembacaan karakter pada plat nomor kendaraan ini dapat dilihat pada gambar 4.12.



**Gambar 4.12** Tampilan Keluar dari Aplikasi

#### 4.5 Pengujian Aplikasi

Salah satu teknik yang digunakan pada pengujian aplikasi ini adalah plat nomor kendaraan. Plat nomor itu sendiri adalah salah satu jenis identifikasi kendaraan bermotor. Bentuknya berupa potongan plat logam atau plastik yang dipasang pada kendaraan bermotor sebagai identifikasi resmi. Karakter yang terdapat pada plat nomor adalah nomor dan huruf, *output* yang diharapkan dan validasi yaitu nomor dan huruf pada plat di setiap kendaraan. Hal tersebut dilakukan untuk menguji kesesuaian identitas dari plat nomor dalam implementasi sistem.

Proses pengujian adalah proses yang penting karena dalam proses ini akan diketahui apakah program yang dibuat sudah sesuai dengan apa yang diharapkan. Dalam pengujian ini dilakukan beberapa langkah yang berbeda untuk memastikan bahwa program berjalan sesuai keinginan. Sebelumnya telah di sediakan plat nomor, seperti yang terlihat pada gambar 4.13.



**Gambar 4.13** Sampel plat nomor kendaraan

Pengujian dilakukan dengan cara pembacaan pada beberapa sampel plat nomor kendaraan bermotor. Dalam pengujian ini dilakukan 10 kali pengujian plat nomor kendaraan bermotor yang berbeda-beda dan dilakukan dalam *type image* yang berbeda. Dengan pengujian ini bisa diperoleh berapa tingkat akurasi pembacaan plat nomor kendaraan bermotor. Hasil pengujian akan ditampilkan pada sebuah tabel untuk mengetahui hasilnya seperti tabel 4.2.

**Tabel 4.2** Pengujian terhadap sampel citra berdasarkan type gambar

No	Plat	Type	Hasil
1	N 4998AA	.jpg	N 4998 AA
		.bmp	N 4998 AA
		.tiff	N 4998 AA
2	DA 2285	.jpg	DA 2285 PH
		.bmp	DA 2285 PH
		.tiff	DA 2285 PH
3	N 3359 IE	.jpg	N 3359 IE
		.bmp	N 3359 IE
		.tiff	N 3359 IE
4	N 3413 LT	.jpg	N 3413 LT
		.bmp	N 3413 LT
		.tiff	N 3413 LT
5	S 4444 ZZ	.jpg	S 4444 ZZ
		.bmp	S 4444 ZZ
		.tiff	S 4444 ZZ
6	W 799 XB	.jpg	W 799 XB
		.bmp	W 799 XB
		.tiff	W 799 XB
7	B 63 NIT	.jpg	B 63 NIT
		.bmp	B 63 NIT
		.tiff	B 63 NIT
8	N 4334 CT	.jpg	N 4334 CT
		.bmp	N 4334 CT
		.tiff	N 4334 CT
9	P 5555 AL	.jpg	H 5555 AL
		.bmp	PL 5EB5 ALL
		.tiff	PL EEEE4 AALLL
10	W 799 XB	.jpg	N 7U9 XB
		.bmp	W 799 XB
		.tiff	W 799 XB

Pengujian selanjutnya dengan menggunakan sampel yang sama dengan jarak pemotretan yang berbeda. Hal ini bertujuan untuk mengetahui keakurasian dalam pengekstrasian karakter pada citra plat nomor kendaraan. Seperti yang terlihat pada tabel 4.3.

**Tabel 4.2** Pengujian terhadap sampel berdasarkan jarak pemotretan

No	Plat	Type	Hasil	Persentase
1	N 4998AA	1.5m	N 4998 AA	100%
		1m	N 4998 AA	100%
		70cm	N 4998 AA	100%
2	DA 2285	1.5m	DA 2285 PH	100%
		1m	DA 2285 PH	100%
		70cm	DA 2285 PH	100%
3	N 3359 IE	1.5m	N 3359 IE	100%
		1m	N 3359 IE	100%
		70cm	N 3359 IE	100%
4	N 3413 LT	1.5m	N 3413 LT	100%
		1m	N 3413 LT	100%
		70cm	N 3413 LT	100%
5	S 4444 ZZ	1.5m	S 4444 ZZ	100%
		1m	S 4444 ZZ	100%
		70cm	S 4444 ZZ	100%
6	W 799 XB	1.5m	W 799 XB	100%
		1m	W 799 XB	100%
		70cm	W 799 XB	100%
7	B 63 NIT	1.5m	B 63 NIT	100%
		1m	B 63 NIT	100%
		70cm	B 63 NIT	100%
8	N 4334 CT	1.5m	N 4334 CT	100%
		1m	N 4334 CT	100%
		70cm	N 4334 CT	100%
9	P 5555 AL	1.5m	H 5555 AL	85%
		1m	PL 5EB5 ALL	71%
		70cm	PL EEEE4 AALLL	42%
10	W 799 XB	1.5m	N 7U9 XB	50%
		1m	W 799 XB	100%
		70cm	W 799 XB	100%

Nilai 100% didapatkan dari :

$$\frac{7}{7} \times 100 \%$$

Keterangan :

7 : adalah jumlah karakter yang bisa dibaca.

7 : adalah jumlah total karakter.

Nilai 85% didapatkan dari :

$$\frac{6}{7} \times 100 \%$$

Keterangan :

6 : adalah jumlah karakter yang bisa dibaca.

7 : adalah jumlah total karakter.

Nilai 71% didapatkan dari :

$$\frac{5}{7} \times 100 \%$$

Keterangan :

5 : adalah jumlah karakter yang bisa dibaca.

7 : adalah jumlah total karakter.

Nilai 50% didapatkan dari :

$$\frac{3}{6} \times 100 \%$$

Keterangan :

3 : adalah jumlah karakter yang bisa dibaca.

6 : adalah jumlah total karakter.

Nilai 42% didapatkan dari :

$$\frac{3}{6} \times 100 \%$$

Keterangan :

3 : adalah jumlah karakter yang bisa dibaca.

7 : adalah jumlah total karakter.

Pengujian selanjutnya dengan menggunakan sampel yang sama dengan nilai threshold yang berbeda. Hal ini bertujuan untuk mengetahui nilai berapa yang bagus dan sesuai dalam kebersihan karakter pada citra plat nomor kendaraan. Seperti yang terlihat pada tabel 4.3.

**Tabel 4.3** Pengujian dengan nilai threshold terbaik

No	Plat	Hasil	Threshold
1	N 4998 AA	N 4998 AA	0.88
2	DA 2285 PH	DA 2285 PH	1.14
3	N 3359 IE	N 3359 IE	0.8
4	N 3413 LT	N 3413 LT	1.2
5	S 4444 ZZ	S 4444 ZZ	0.91
6	W 799 XB	W 799 XB	0.96
7	B 63 NIT	B 63 NIT	1
8	N 4334 CT	N 4334 CT	1.24
9	P 5555 AL	H 5555 AL	1.5
10	W 799 XB	W 799 XB	1.08

Dari hasil pengujian pada Tabel 4.1 dengan menggunakan 10 sampel plat nomor kendaraan bahwasannya penerapan *optical character recognition* pada plat nomor kendaraan berhasil 80% pada sampel yang digunakan. Pada pengujian terhadap jarak yang dibutuhkan pada Tabel 4.2 dari 10 sampel plat nomor

kendaraan yang diuji, dalam pengujian tersebut terdapat 2 sampel terjadi kesalahan saat pembacaan karakter plat nomor kendaraan. Pada pengujian terhadap nilai threshold yang dibutuhkan pada Tabel 4.3 setiap citra yang diekstrak memiliki nilai yang berbeda dikarenakan citra plat nomor itu sendiri dalam keadaan bias dibaca atau sudah rusak.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Setelah membuat aplikasi penerapan *optical character recognition* pada plat nomor kendaraan dan membuat laporan, maka dapat disimpulkan :

1. Berdasarkan pengujian aplikasi semua fungsi tombol berjalan dengan baik.
2. Penerapan *Optical Character Recognition* dapat diterapkan pada citra plat nomor kendaraan
3. Untuk jarak pemotretan yang paling baik antara 70cm – 150cm agar citra plat nomor kendaraan dapat dikenali
4. Dilihat dari nilai threshold semua citra berbeda dikarenakan, dari plat nomor itu sendiri dalam keadaan baik dan bisa dibaca
5. Pada 10 sampel citra plat nomor kendaraan yang diuji, didapatkan hasil 8 sampel terbaca dan 3 sampel gagal.

#### **5.2 Saran**

Setelah melakukan pembuatan membuat aplikasi penerapan *optical character recognition* pada plat nomor kendaraan, maka pengembangan yang dapat dilakukan :

1. Untuk kedepannya proses input gambar plat nomor dapat dikembangkan dengan menggunakan kamera *webcam / CCTV* secara *real time*.
2. Fitur pemotongan citra lebih otomatis tanpa memotong secara manual

3. Proses ekstraksi pada sistem dapat diterapkan menggunakan metode lain seperti metode *Neuron*, metode *Hebb*, metode *Perceptron*, metode *Adaline*, dan metode *Kohonen*.
4. Evaluasi dapat dilakukan antar metode algoritma yang lainnya

## DAFTAR PUSTAKA

- [1] Drs. Jong Jek Siang, M.Sc. (2005). *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Yogyakarta: Andi Publisher.
- [2] Hofman. (2013, Desember). License Plate Recognition – A Tutorial. *License Plate Recognition - A Tutorial*. Retrieved from <http://www.licenseplaterecognition.com>
- [3] Munir R. (2004). *Computer Vision dan pengolahan citra*. Bandung: INFORMATIKA.
- [4] Prijono, A., & Wijaya, M. C. (2007). *Pengolahan Citra Digital Menggunakan MatLAB*. INFORMATIKA.
- [5] Suyanto, ST, Msc. (2007). *Artificial Intelligence (Revisi.)*. Bandung: Andi Publisher.

# LAMPIRAN

### *Source Code Untuk Tombol Open*

```
function open_Callback(hObject, eventdata, handles)
clc;
reset_callback(hObject, eventdata, handles)

open=guidata(gcbo);
[namafile,direktori]=uigetfile({'*.jpg;*.bmp;*.tif'},
'OpenImage');
I=imread(namafile);

set(open.figure1,'CurrentAxes',open.axes1);
set(imagesc(I));colormap('gray');
set(open.axes1,'Userdata',I);
set(handles.text,'string',strcat(direktori));
```

### *Source Code Untuk Tombol Cropping*

```
function CROP_Callback(hObject, eventdata, handles)
reset_callback1(hObject, eventdata, handles)
open=guidata(gcbo);
I=get(open.axes1,'Userdata');
a = imcrop(I);

set(open.figure1,'CurrentAxes',open.axes7);
set(imagesc(a));colormap('gray');
set(open.axes7,'Userdata',a);

guidata(hObject, handles);
```

### *Source Code Untuk Tombol Close*

```
function CLOSE_Callback(hObject, eventdata, handles)
button=questdlg('are you sure want to quit?','warning
message','yes','no','');
if strcmp(button,'yes'),
close all;
end
```

### *Source Code Untuk Slider*

```
function slider_Callback(hObject, eventdata, handles)
open=guidata(gcbo);
I=get(open.axes7,'Userdata');
```

```

c=get(hObject,'Value');
e=num2str(c);
set(handles.edit,'String',e);

Igray = rgb2gray(I);

b1=str2double(get(handles.edit,'String'));

threshold = graythresh(Igray);
handles.Ibw = im2bw(Igray,threshold*b1);

imagesc(handles.Ibw,'parent',handles.axes2);
axis(handles.axes2,'off'); colormap(gray);
guidata(hObject, handles);

```

### Source Code Untuk Process

```

function PROCESS_Callback(hObject, eventdata, handles)
load set
handles.output = hObject;
handles.limit = 10;
%----- Thinning -----
handles.skel = bwmorph(handles.Ibw,'thin',inf);
%----- Blob Analysis ----
handles.stat = regionprops(handles.Ibw,'boundingbox');
for x = 1 : length(handles.stat)
    handles.bb = handles.stat(x).BoundingBox;
    handles.segmen{x} =
imcrop(handles.skel,handles.bb);
end
%----- Line Segmentation -----
handles.lineImage = lineSegmentation(handles.skel);
%function linesegmentation
figure(1),
for x = 1 : size(handles.lineImage,1)
    handles.part{x} =
lineSegment(handles.skel,handles.lineImage(x,:));%funct
ion linesegment
    subplot(size(handles.lineImage,1),1,x),
imshow(handles.part{x});
    pause(0.5);
end
%----- Line Segmentation -----
figure(2),
for x = 1 : size(handles.lineImage,1)
    handles.characterImage =

```

```

characterSegmentation(handles.part{x}); %function
characterSegmentation
    array = handles.characterImage;
    for y = 1 : size(array,1)
        handles.bag{y} =
characterSegment(handles.part{x},array(y,:)); %function
characterSegment
        number = (x-1)*handles.limit + y;

subplot(size(handles.lineImage,1),handles.limit,number)
, imshow(handles.bag{y});
    pause(0.1);
    end
%----- Zoning -----
figure (3),
row = 7;    col = 5;
% Load templates
load learning
global learning
% Compute the number of letters in template file
num_letras=size(learning,2);
for x = 1 : length(handles.stat)
    handles.feature = imresize(handles.segmen{x},
[row*10 col*10]);
    for r = 1 : row
        for c = 1 : col
            handles.zone = sum(handles.feature((r *
10 - 9 : r * 10), (c * 10 - 9 : c * 10)));
            handles.featurevector((r - 1) * 5 + c)
= sum(handles.zone);
        end
    end
    handles.featurevector = ((100 -
handles.featurevector) / 100);
    for j = 1 : length(handles.featurevector)
        if handles.featurevector(j) <= 0.9,
handles.featurevector(j) = 1;
        else
            handles.featurevector(j) = 0;
        end
    end
    handles.objfeature{x} = handles.featurevector';
%----- Recognition -----
handles.numvar(x) =
back_propagation(handles.objfeature{x},num_letras);
%function back_propagation
    subplot(1,length(handles.stat),x),
plotchar(handles.objfeature{x});
end

```

```

imagen = bwareaopen(handles.Ibw,30);
%Storage matrix word from image
word=[ ];
re=imagen;
% Load templates
load templates
global templates
% Compute the number of letters in template file
num_letras=size(templates,2);
while 1
    %Fcn 'lines' separate lines in text
    [fl re]=lines(re); %function line
    imgn=fl;
    %Uncomment line below to see lines one by one
    %imshow(fl);pause(0.5)
    %-----
    % Label and count connected components
    [L Ne] = bwlabel(imgn);
    for n=1:Ne
        [r,c] = find(L==n);
        % Extract letter
        n1=imgn(min(r):max(r),min(c):max(c));
        % Resize letter (same size of template)
        img_r=imresize(n1,[42 24]);
        %Uncomment line below to see letters one by one
        %imshow(img_r);pause(0.5)
        %-----
        % Call fcn to convert image to text
        letter=read_letter(img_r,num_letras);
        % Letter concatenation

nume=['1','2','3','4','5','6','7','8','9','0','A','B','
C','D','E','F','G','H','I','J','K','L','M','N','O','P',
'Q','R','S','T','U','V','W','X','Y','Z'];
        for j=1:36
            if letter==nume(j)
                word = [word letter]
            end
        end
    end
end
word
if isempty(re) %See variable 're' in Fcn 'lines'
    break
end
end
end

```

```
handles.hasil=strcat(word);  
set(handles.text2,'string',handles.hasil);  
guidata(hObject, handles);  
end
```

**BERKAS**

**LAIN**

STUDI  
tentang  
the Best  
the Best  
the Best

1998

General Form

Nama

Nia

Program Studi

Desain Interior  
dan  
Interior

Sistem Perencanaan

Desain dan Perencanaan

Desain  
Desain  
Desain  
Desain

INSTITUT TEKNOLOGI NASIONAL  
Jln. Bendungan Sigura-gura No. 2  
Jln. Raya Karanglo Km2  
MALANG

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : SYAIFUL ULUM

Nim : 1018022

Program Studi : Teknik Informatika

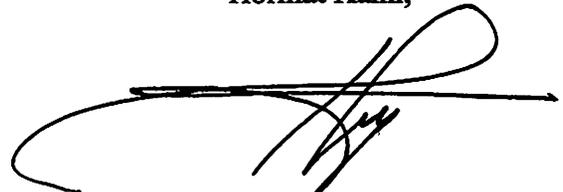
Dengan ini menyatakan bersedia / tidak bersedia \*) membimbing skripsi dari mahasiswa tersebut dengan judul :

**Sistem Pembuka Pintu Portal Dengan Penerapan Optical Character Recognition Pada Plat Nomor Kendaraan**

Demikian Surat Pernyataan ini kami buat agar dipergunakan seperlunya.

Malang, 25/6/2013

Hormat Kami,



Ir. Yudi Limpraptono, MT  
NIP.Y.1039500274

Catatan :  
Setelah disetujui agar formulir ini diserahkan mahasiswa/i yg bersangkutan kepada Jurusan untuk diproses lebih lanjut  
\*) coret yang tidak perlu

Form S-3b

INSTITUT  
INFORMATIKA  
BANDUNG

Departemen Sistem Informasi

Nama

NIM

Program Studi

Dengan ini menyatakan  
dengan judul

Sistem Informasi

dan

Ditentukan sebagai Petugas

Catatan :  
Setelah selesai  
yg bersangkutan  
(\* coreng tidak)

INSTITUT TEKNOLOGI NASIONAL  
Jln. Bendungan Sigura-gura No. 2  
Jln. Raya Karanglo Km2  
MALANG

### PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : SYAIFUL ULUM

Nim : 1018022

Program Studi : Teknik Informatika

Dengan ini menyatakan bersedia / tidak bersedia \*) membimbing skripsi dari mahasiswa tersebut dengan judul :

### Sistem Pembuka Pintu Portal Dengan Penerapan Optical Character Recognition Pada Plat Nomor Kendaraan

Demikian Surat Pernyataan ini kami buat agar dipergunakan seperlunya.

Malang, \_\_\_\_\_

Hormat Kami,



Ali Mahmudi, B.Eng.P.hd  
NIP.P.1031000429

Catatan :  
Setelah disetujui agar formulir ini diserahkan mahasiswa/i yg bersangkutan kepada Jurusan untuk diproses lebih lanjut  
\*) coret yang tidak perlu

Form S-3b



## FORMULIR BIMBINGAN SKRIPSI

Nama : SYAIFUL ULUM  
NIM : 10.18.022  
Masa Bimbingan : 21 OKTOBER 2013 S/D 21 MARET 2014  
Judul Skripsi : RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER  
PADA CITRA PLAT NOMOR KENDARAAN

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	16 November 2013	(Revisi) BAB I BAB II	
2	30 November 2013	(Revisi) BAB I, II, III	
3	10 Desember 2013	(Fix) BAB I dan II	
4	25 Januari 2014	(Revisi) BAB III	
5	30 Januari 2014	(Fix) BAB III	
6	5 Februari 2014	(Revisi) BAB IV dan V	
7	16 Februari 2014	(Fix) BAB IV dan V, (Revisi) Makalah, Demo Program	
8	20 Februari 2014	(Fix) BAB IV dan V, (Revisi) Makalah	
9	6 Februari 2014	(Fix) Makalah	
10	15 Februari 2014	(Fix) Laporan	

Malang, Februari 2014  
Dosen Pembimbing

**Ir. F. Yudi Limpraptono, MT**  
NIP.Y.1039500274



## FORMULIR BIMBINGAN SKRIPSI

Nama : SYAIFUL ULUM  
NIM : 10.18.022  
Masa Bimbingan : 21 OKTOBER 2013 S/D 21 MARET 2014  
Judul Skripsi : RANCANG BANGUN APLIKASI PEMBACAAN KARAKTER  
PADA CITRA PLAT NOMOR KENDARAAN

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	14 November 2013	(Revisi) BAB I BAB II	
2	29 November 2013	(Revisi) BAB I, II, III	
3	30 November 2013	(Fix) BAB I dan II	
4	23 Januari 2014	(Revisi) BAB III	
5	28 Januari 2014	(Fix) BAB III	
6	29 Januari 2014	(Revisi) BAB IV dan V	
7	30 Januari 2014	(Fix) BAB IV dan V, (Revisi) Makalah, Demo Program	
8	3 Februari 2014	(Fix) BAB IV dan V, (Revisi) Makalah	
9	5 Februari 2014	(Fix) Makalah	
10	12 Februari 2014	(Fix) Laporan	

Malang, Februari 2014  
Dosen Pembimbing

Ali Mahmudi, B.Eng, PhD  
NIP.P. 1031000425



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
PROGRAM STUDI TEKNIK INFORMATIKA S-1

---

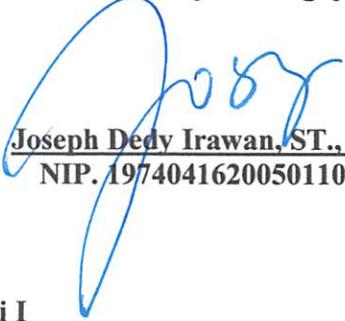
**BERITA ACARA UJIAN SKRIPSI  
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Syaiful Ulum  
NIM : 10.18.022  
Program Studi : Teknik Informatika S-1  
Judul : Rancang Bangun Aplikasi Pembacaan Karakter Pada Citra  
Plat Nomor Kendaraan

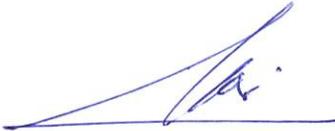
Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada:

Hari : Selasa  
Tanggal : 18 Februari 2014  
Nilai : A

**Panitia Penguji Skripsi,  
Ketua Majelis Penguji**

  
**Joseph Dedy Irawan, ST., MT.**  
NIP. 197404162005011002

**Dosen Penguji I**

  
**Karina Auliasari, ST, M.Eng.**  
NIP.P.1031000426

**Dosen Penguji II**

  
**Sandy Nataly Mantja, S.Kom.**  
NIP.P.1030800418



## FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Program Studi Teknik Informatika, perlu adanya perbaikan skripsi untuk mahasiswa:

Nama : Syaiful Ulum  
NIM : 10.18.022  
Program Studi : Teknik Informatika S-1  
Judul : Rancang Bangun Aplikasi Pembacaan Karakter Pada Citra Plat Nomor Kendaraan

No.	Penguji	Tanggal	Uraian	Paraf
1	Penguji I	18 Februari 2014	- Perbaikan pada sitasi - Perbaiki pada <i>coding</i>	
2	Penguji II	18 Februari 2014	- Jarak <i>capture</i> citra - Perbaiki pada blog diagram	

Dosen Penguji I

Karina Auliasari, ST, M.Eng.  
NIP.P.1031000426

Dosen Penguji II

Sandy Nataly Mantja, S.Kom.  
NIP.P.1030800418

Mengetahui

Dosen Pembimbing I

Ir. Yudi Limpraptono. MT  
NIP.Y.1039500274

Dosen Pembimbing II

Ali Mahmudi, B.Eng.P.hd  
NIP.P. 1031000429