

**Pengenalan Huruf Arab Berbasis Pengolahan
Citra**

TESIS

**Karya tulis sebagai salah satu syarat
untuk memperoleh gelar Magister dari
Institut Teknologi Bandung**

Oleh

Ainatul Radhiah

NIM: 23215145

Program Studi Magister Teknik Elektro



**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG**

2017

**PENGENALAN HURUF ARAB BERBASIS PENGOLAHAN
CITRA**

Oleh

Ainatul Radhiah

Bandung,

Menyetujui

Pembimbing I,

Pembimbing II,

Prof. Dr. Ir. Carmadi Machbub

Dr. techn. Ary Setijadi P., S.T., M.T

Pembimbing III,

Dr. Egi Muhammad Idris H.

ABSTRAK

PENGENALAN HURUF ARAB BERBASIS PENGOLAHAN

CITRA

Oleh

Ainatul Radhiah

NIM : 23215145

PROGRAM STUDI MAGISTER TEKNIK ELEKTRO

Dalam proses pembelajaran huruf Arab, mengenali huruf Arab merupakan bagian yang sangat penting. Pada fase ini, pelajar pemula akan mengalami kesulitan. Pembelajaran huruf Arab akan lebih efektif jika ada sebuah sistem yang mampu mengenali huruf Arab, baik dalam bentuk terisolasi maupun yang terdapat dalam kalimat.

Pada tesis ini akan dirancang sebuah sistem yang dapat mengenali huruf Arab dalam bentuk terisolasi dan huruf Arab yang terdapat dalam kalimat. Sistem memiliki lima tahap: binerisasi, segmentasi, penipisan, ekstraksi fitur dan klasifikasi. Dalam tahap binerisasi dilakukan dengan mengubah citra kedalam bentuk biner yang memiliki nilai 0 dan 1. Pada tahap segmentasi dilakukan dengan algoritma *Zidouri* yang memiliki beberapa parameter, fitur dan aturan untuk melakukan segmentasi suatu kata. Pada tahap penipisan dilakukan dengan algoritma *Stentiford* yang memiliki 4 *template*, *end point* dan jumlah konektivitas untuk memeriksa apakah citra piksel dihapus atau tidak. Pada tahap ekstraksi di ekstrak 3 fitur, yang pertama adalah *chain code* yang sudah di normalisasi, yang kedua adalah jumlah titik, yang ketiga adalah posisi titik. Pada tahap klasifikasi dilakukan dengan membandingkan dua metode yaitu metode *Neural Network* dan *Hidden Markov Model*. Hasil dari pengenalan huruf Arab menggunakan metode klasifikasi *Neural Network* untuk huruf terisolasi mencapai akurasi 100% dan hasil pengenalan huruf dalam kalimat mencapai akurasi 69%. Hasil pengenalan huruf Arab menggunakan metode *Hidden Markov Model* untuk huruf terisolasi mencapai akurasi 71% dan hasil pengenalan huruf dalam kalimat mencapai akurasi 50%.

Kata kunci : pengenalan huruf Arab, algoritma *stentiford*, *chain code*, *neural network*, *hidden markov model*

ABSTRACT

ARABIC LETTER RECOGNITION BASED ON IMAGE PROCESSING

By

Ainatul Radhiah

NIM : 23215145

(Electrical Engineering Master Program)

In the process of learning Arabic letters, recognizing Arabic letters is a very important part. In this phase, novice students will experience difficulties. Learning Arabic letters will be more effective if there is a system that is able to recognize Arabic letters, either in isolated form or in the sentence.

In this thesis will be designed a system that can recognize Arabic letters in the form of isolated and Arabic letters in the sentence. The system has five stages: binarization, segmentation, thinning, feature extraction and classification. Binarization stage is done by converting the image into a binary form that has a value of 0 and 1. In the segmentation stage is done with Zidouri algorithm which has several parameters, features and rules to segment a word. In the thinning stage is done by Stentiford algorithm which has 4 templates, end point and number of connectivity to check whether pixel image is deleted or not. In the feature extraction stage three features extracted, the first is the normalized chain code, the second is the number of points, the third is the point position. At the classification stage is done by comparing the two methods of Neural Network and Hidden Markov Model. The result of the recognition of Arabic letters using the Neural Network classification method for isolated letters reaches 100% accuracy and the result of letter recognition in the sentence reaches 69% accuracy. Arabic letters recognition using the method of Hidden Markov Model for isolated letters reaches 71% accuracy and the result of letter recognition in sentence reaches 50% accuracy.

Keywords : Arabic letter recognition , stentiford algorithm, chain code, neural network, hidden markov model.

PEDOMAN PENGGUNAAN TESIS

Tesis S2 yang tidak dipublikasikan terdaftar dan tersedia di Perpustakaan Institut Teknologi Bandung, dan terbuka untuk umum dengan ketentuan bahwa hak cipta ada pada pengarang dengan mengikuti aturan HaKI yang berlaku di Institut Teknologi Bandung. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau peringkasan hanya dapat dilakukan seizin pengarang dan harus disertai dengan kebiasaan ilmiah untuk menyebutkan sumbernya.

Sitasi hasil penelitian Tesis ini ditulis dalam bahasa Indonesia sebagai berikut:

Ainatul Radhiah (2017): *Pengenalan Huruf Arab Berbasis Pengolahan Citra*, Tesis Program Magister, Institut Teknologi Bandung.

Dan dalam bahasa Inggris sebagai berikut:

Ainatul Radhiah (2017): *Arabic Letter Recognition Based On Image Processing*, Master's Program Thesis, Institut Teknologi Bandung.

Memperbanyak atau menerbitkan sebagian atau seluruh tesis haruslah seizin Dekan Sekolah Pascasarjana, Institut Teknologi Bandung.

KATA PENGANTAR

Puji syukur penulis panjatkan ke-hadirat Allah SWT, karena atas segala limpahan rahmat dan karunianya yang diberikan sehingga penulis dapat menyusun dan menyelesaikan tesis ini. Shalawat dan salam tercurah kepada Rasulullah Muhammad SAW beserta keluarganya.

Selama melaksanakan tesis ini, penulis mendapat bantuan dan dukungan dari berbagai pihak. Untuk itu, penulis ingin mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Carmadi Machbub, selaku pembimbing pertama yang telah memberikan bimbingan, arahan dan motivasi dalam menyelesaikan tesis ini.
2. Bapak Dr. techn. Ary Setijadi Prihatmanto, S.T., M.T., selaku pembimbing kedua dan dosen wali yang telah memberikan bimbingan, arahan, dan motivasi dalam menyelesaikan perkuliahan.
3. Bapak Dr. Egi Muhammad Idris Hidayat, selaku pembimbing ketiga yang telah memberikan bimbingan, arahan dan motivasi dalam menyelesaikan tesis ini.
4. Kedua orang tua yang saya cintai, Ibu Desfaharni dan Bapak Yakub terimakasih dukungan dan do'anya, sehingga penulis dapat menyelesaikan tesis ini.
5. Kepada kedua adik saya Husnul dan Aulia yang selalu memberikan do'a dan dukungannya.
6. Kak Hendy Irawan, Arief Rahman, Agis, Sandy, Yuliska, dan mas Sunaryo, yang selalu menyertai penulis selama mengerjakan penelitian dan bekerjasama dalam menyelesaikan tesis ini.
7. Rekan – rekan Teknologi Digital Media dan Game 2015 dan rekan – rekan Teknik Informatika bidang keahlian Sistem Intelegen 2016 yang selalu memberikan semangat, dan diskusi – diskusinya selama menyelesaikan perkuliahan.
8. Dan pihak – pihak yang juga telah banyak memberikan bantuan kepada penulis yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa tesis ini bukanlah tanpa kelemahan, untuk itu kritik dan saran sangat diharapkan.

Akhir kata, semoga tesis ini dapat bermanfaat bagi para pembacanya.

Bandung, November 2017

Penulis

DAFTAR ISI

	Halaman
ABSTRAK	i
ABSTRACT	ii
PEDOMAN PENGGUNAAN TESIS.....	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
Halaman	vi
DAFTAR LAMPIRAN	ix
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
I.1 Latar Belakang Masalah	1
I.2 Tujuan Penelitian.....	2
I.3 Batasan Masalah.....	2
I.4 Metodologi Penelitian	3
I.5 Sistematika Pembahasan	3
BAB II TINJAUAN PUSTAKA.....	5
II.1 Tulisan Arab.....	5
II.2 Pengenalan Huruf Arab.....	6
II.3 Proses Pengenalan Huruf Arab	7
II.3.1 Binerisasi.....	8
II.3.2 Segmentasi	8
II.3.3 Penipisan	10

II.3.4 Tahap Ekstraksi Fitur	13
II.3.4.1 Chaincode.....	13
II.3.4.1.1 Normalisasi Chaincode	14
II.3.4.2 Fitur Jumlah Titik.....	15
II.3.4.3 Fitur Posisi Titik	16
II.4 Artificial Neural Network	16
II.4.1 Jaringan Syaraf Tiruan	16
II.4.2 Model Neuron	17
II.4.3 Arsitektur jaringan	18
II.4.4 Training	19
II.4.5 Fungsi aktivasi	20
II.5 Hidden Markov Model.....	21
II.5.1 Evaluasi	22
II.5.2 Pengkodean (Decoding)	23
II.5.3 Learning	24
BAB III PERANCANGAN DAN EKSPERIMEN	25
III.1 Perancangan Awal Sistem dan Eksperimen Pengenalan Huruf Arab	25
III.2 Perancangan Sistem dan Eksperimen	27
III.2.1 Eksperimen Binerisasi	28
III.2.2 Eksperimen Segmentasi	28
III.2.3 Eksperimen Penipisan.....	28
III.2.4 Eksperimen Ekstraksi Fitur	28
III.3 Perancangan Sistem dan Eksperimen Pengklasifikasi.....	29
III.3.1 Tahap Pelatihan dan Pengujian dengan Metode Neural Network	29
III.3.2 Tahap Pelatihan dan Pengujian dengan Metode Hidden Markov Model	30

BAB IV PENGUJIAN DAN PEMBAHASAN	31
IV.1 Pengujian Sistem	31
IV.1.1 Pengujian Sistem Pengenalan Huruf Arab	32
IV.1.1.2 Hasil Pengujian Binerisasi.....	32
IV.1.1.2 Hasil Pengujian Segmentasi	33
IV.1.1.2.1 Perbandingan Hasil Pengujian Segmentasi	37
IV.1.1.3 Hasil Pengujian Penipisan	38
IV.1.1.4 Ekstraksi Fitur	38
IV.1.1.5 Hasil Pengujian Sistem Pengenalan Huruf Arab dengan Neural Network	39
IV.1.1.6 Hasil Pengujian Sistem Pengenalan Huruf Arab dengan Hidden Markov Model	42
IV. 2 Pembahasan	45
IV.2.1 Perbandingan Pengujian Huruf Arab Terisolasi.....	45
IV.2.2 Perbandingan Pengujian Huruf Arab dalam Kalimat.....	46
IV.2.3 Rekapitulasi Pembahasan dari Seluruh Hasil Pengujian	47
BAB V KESIMPULAN DAN SARAN.....	49
V.1 Kesimpulan	49
V.2 Saran	49
DAFTAR PUSTAKA	51

DAFTAR LAMPIRAN

Lampiran A Hasil Ekstraksi Fitur Data Latih	L1
A.1 Hasil Ekstraksi Fitur dengan Font Arial.....	L1
A.2 Hasil Ekstraksi Fitur dengan Font Tahoma.....	L6
A.3 Hasil Ekstraksi Fitur dengan Font Times New Roman.....	L11

DAFTAR GAMBAR

Gambar II.1 Diagram blok umum sistem pengenalan huruf Arab	7
Gambar II.2 Hasil segmentasi dengan algoritma Zidouri	10
Gambar II.3 Pola huruf dan hasil penipisan	11
Gambar II.4 Perbandingan algoritma penipisan	11
Gambar II.5 Template algoritma Stentiford	12
Gambar II.6 Penomoran nilai k pada algoritma Stentiford	12
Gambar II.7 Jumlah konektivitas 1 algoritma Stentiford	13
Gambar II.8 Arah mata angina chain code	13
Gambar II.9 Fitur jumlah titik	16
Gambar II.10 Cara menentukan posisi titik	16
Gambar II.11 Model neuron	17
Gambar II.12 Perhitungan bobot dan fungsi aktivasi neuron	18
Gambar II.13 Jaringan layer tunggal	19
Gambar II.14 Jaringan layer jamak	19
Gambar II.15 Hidden Markov Model	22
Gambar III.1 Diagram blok utama dalam sistem pengenalan huruf Arab	26
Gambar III.2 Proses pengenalan huruf Arab dengan metode Neural Network	29
Gambar III.3 Proses pengenalan huruf Arab dengan metode Hidden Markov Model	30
Gambar IV.1 Diagram blok utama pengujian pengenalan huruf Arab	31
Gambar IV.2 Citra biner huruf Ain terisolasi	32

Gambar IV.3 Citra biner huruf Ain di awal	33
Gambar IV.4 Citra biner huruf Ain di tengah	33
Gambar IV.5 Citra biner huruf Ain di akhir	33
Gambar IV.6 Hasil segmentasi kalimat satu	34
Gambar IV.7 Hasil Segmentasi kalimat dua	34
Gambar IV.8 Hasil Segmentasi kalimat tiga.....	35
Gambar IV.9 Hasil pengujian segmentasi dengan font Arial Unicode Ms, Tahoma dan Times New Roman.....	37
Gambar IV.10 Hasil penipisan.....	38
Gambar IV.11 Pengujian huruf Arab terisolasi dengan metode Neural Network dan Hidden Markov Model.....	45
Gambar IV.12 Pengujian huruf Arab dalam kalimat dengan metode Neural Network dan Hidden Markov Model	46

DAFTAR TABEL

Tabel II.1 Huruf Arab dan empat bentuknya	5
Tabel IV.1 Hasil segmentasi 10 kalimat dengan font Arial Unicode Ms	35
Tabel IV.2 Hasil segmentasi 10 kalimat dengan font Tahoma	36
Tabel IV.3 Hasil segmentasi 10 kalimat dengan font Times New Roman	36
Tabel IV.4 Data training huruf Ba	38
Tabel IV.5 Data hasil pengujian pengenalan huruf Arab terisolasi dengan metode Neural Network.....	39
Tabel IV.6 Data hasil pengujian pengenalan huruf Arab dalam kalimat dengan metode Neural Network.....	41
Tabel IV.7 Data hasil pengujian pengenalan huruf Arab terisolasi dengan metode Hidden Markov Model.....	42
Tabel IV.8 Data hasil pengujian pengenalan huruf Arab dalam kalimat dengan metode Hidden Markov Model	44
Tabel IV.9 Perbandingan hasil rekapitulasi akurasi.....	47

BAB I

PENDAHULUAN

I.1 Latar Belakang Masalah

Bahasa Arab digunakan lebih dari 1 juta orang di dunia ^[1]. Dalam proses pembelajaran huruf Arab, pengenalan huruf Arab adalah bagian yang sangat penting. Pada proses ini pelajar pemula akan mengalami kesulitan dalam mengenali huruf Arab. Proses pembelajaran akan lebih efektif jika ada sebuah sistem yang dapat mengenali huruf Arab baik dalam bentuk terisolasi maupun dalam kalimat.

Bahasa Arab memiliki 28 huruf inti dan 3 huruf tambahan yang ditulis dari kanan ke kiri secara bersambung, baik dalam tulisan cetak maupun tulisan tangan. Oleh karena itu pengenalan huruf Arab membutuhkan proses segmentasi. Beberapa dari huruf Arab memiliki bentuk yang mirip yang dapat dibedakan dengan jumlah titik dan posisi titik. Setiap huruf Arab memiliki bentuk berbeda bergantung pada posisinya dalam kata, yaitu terisolasi, di awal, di tengah dan di akhir.

Penelitian sebelumnya tentang pengenalan huruf Arab telah dilakukan. Nimas ^[2] dan rekan-rekan (2017), melakukan penelitian tentang pengenalan huruf Arab terisolasi menggunakan neural network dengan metode *learning backpropagation* dan metode *learning vector quantisation*. Hasil dari penelitian dengan *backpropagation* mencapai akurasi 98.81% dan pengenalan dengan *learning vector quantisation* mencapai akurasi 51.19%. M. Albakor ^[3] dan rekan-rekan (2009) telah melakukan penelitian dalam pengenalan huruf Arab yang menerapkan metode segmentasi, penelitian ini mencapai akurasi 98.7%. Albadr ^[4] (2013) mengembangkan pengenalan huruf Arab yang terdapat dalam kalimat dengan menggunakan metode klasifikasi C4.5, penelitian ini mencapai akurasi 48%. Izakian ^[5] (2008), mengembangkan pengenalan huruf Farsi/Arab terisolasi menggunakan *Support Vector Machine*, hasil mencapai akurasi 97.4%.

Berdasarkan dari penelitian sebelumnya, penelitian tentang pengenalan huruf Arab dalam kalimat belum banyak diketahui. Oleh karena itu dalam penelitian ini akan dikembangkan sistem pengenalan huruf Arab dalam bentuk terisolasi maupun yang terdapat dalam kalimat. Pada penelitian ini akan dibandingkan dua metode klasifikasi untuk pengenalan huruf Arab yaitu dengan metode *Neural Network* dan *Hidden Markov Model*.

Tujuan dari pengembangan sistem pengenalan huruf Arab ini adalah untuk membantu proses pembelajaran huruf Arab baik dalam bentuk terisolasi maupun dalam kalimat.

I.2 Tujuan Penelitian

Berdasarkan latar belakang dan rumusan masalah, berikut tujuan yang ingin dicapai dalam penelitian ini.

1. Merancang sistem yang dapat mengenali huruf Arab dalam bentuk terisolasi dan yang terdapat dalam kalimat.
2. Membandingkan sistem pengenalan huruf Arab melalui metode *Neural Network* dan metode *Hidden Markov Model*.

I.3 Batasan Masalah

Hal-hal yang menjadi batasan pada penelitian ini adalah sebagai berikut.

1. Tulisan huruf arab yang digunakan adalah dari tulisan cetak dalam posisi terisolasi dan yang terdapat dalam kalimat.
2. Sistem pengenalan huruf Arab yang dibangun khusus untuk tulisan Arab dengan Bahasa Arab.
3. Font yang digunakan adalah 3 jenis font, yaitu Arial Unicode Ms, Tahoma dan Times New Roman

I.4 Metodologi Penelitian

Tahapan-tahapan kegiatan yang dilakukan dalam penelitian ini dijelaskan sebagai berikut.

1. Studi literatur

Mempelajari dan menganalisis berbagai sumber informasi seperti buku-buku, paper dan literatur referensi yang serupa dengan penelitian yang akan dilakukan

2. Analisis desain

Berdasarkan hasil studi literatur akan dibuat analisis desain dalam menyelesaikan proses perbandingan algoritma untuk setiap metode.

3. Pembuatan sistem

Hasil perancangan diimplementasikan dengan menggunakan Bahasa pemrograman Java sebagai pemrosesan citra.

4. Pengujian dan evaluasi

Berdasarkan aplikasi yang telah dibuat kemudian dilakukan uji coba sistem dan mengevaluasi sistem sesuai dengan tujuan penelitian.

5. Pengambilan data

Setelah pengujian dan didapatkan hasil yang sesuai dengan tujuan penelitian kemudian dilakukan pengambilan data.

6. Penulisan laporan

Penulisan laporan hasil pengujian dan pengambilan data dibukukan dalam laporan penelitian.

I.5 Sistematika Pembahasan

Dalam penulisan laporan tesis ini terdiri dari lima bab dengan masing–masing bab menguraikan beberapa hal yang terkait dengan perancangan yang dilakukan.

Bab I Pendahuluan. Bab ini mengemukakan latar belakang masalah, tujuan penelitian, batasan masalah, metode penelitian, dan sistematika pembahasan.

Bab II Tinjauan Pustaka. Bab ini menjelaskan teori dasar mengenai tulisan Arab, pengenalan huruf Arab, Proses pengenalan huruf Arab, neural network dan hidden markov model.

Bab III Perancangan Sistem. Bab ini membahas perancangan sistem dan eksperimen pengenalan huruf Arab, eksperimen binerisasi, eksperimen segmentasi, eksperimen penipisan, eksperimen ekstraksi fitur, dan eksperimen klasifikasi.

Bab IV Pengujian dan Pembahasan. Bab ini membahas pengujian sistem pengenalan huruf arab yang menghasilkan data yang dibutuhkan sebagai analisis pembahasan perbandingan metode *Neural Network* dan *Hidden Markov Model*.

Bab V Penutup. Bab ini mengemukakan bagian penutup dari laporan penelitian yang meliputi kesimpulan dan saran agar dapat dikembangkan dengan metode lainnya untuk sistem kerja yang sama.

Dalam bab selanjutnya akan disampaikan teori-teori dasar hasil dari peninjauan beberapa literatur yang digunakan. Literatur tersebut dapat berupa buku-buku, makalah penelitian yang telah dipublikasikan, dan sebagainya yang dijadikan sebagai referensi dalam melakukan penelitian ini.

BAB II

TINJAUAN PUSTAKA

II.1 Tulisan Arab

Bahasa Arab digunakan oleh lebih dari 1 juta orang di dunia ^[1], bahasa ini adalah bahasa resmi dari 25 negara, dan merupakan bahasa peribadatan dalam agama Islam karena merupakan bahasa yang dipakai oleh Al-Qur'an.

Huruf Arab terdiri dari 28 huruf dasar dan 3 huruf tambahan. Tulisan Arab ditulis dari kanan ke kiri dan selalu ditulis dalam bentuk bersambung baik dalam tulisan cetak maupun tulisan tangan. Huruf Arab mengalami perubahan bentuk sesuai dengan posisinya di dalam kata, yaitu posisi terisolasi, bentuk tersambung dari kiri atau di posisi awal kata, bentuk tersambung dari kedua sisi atau di posisi tengah kata, dan bentuk tersambung dari kanan saja atau di posisi akhir kata. Terdapat sembilan huruf yang tidak dapat disambung dari kiri, sehingga tidak memiliki bentuk di awal dan di tengah.

Beberapa huruf Arab memiliki kemiripan bentuk, namun bisa dibedakan berdasarkan jumlah titik dan posisi titik. Seluruh huruf Arab dan bentuk dituliskannya dapat dilihat pada tabel II.1

Tabel II.1. Huruf Arab dan empat bentuknya

No	Nama	Alih Aksara	Bentuk Terisolasi	Bentuk Tersambung		
				Awal	Tengah	Akhir
1.	alif	-	ا	-	-	ا
2.	ba'	b	ب	ب	ب	ب
3.	ta'	t	ت	ت	ت	ت
4.	tsa	ts	ث	ث	ث	ث
5.	jim	j	ج	ج	ج	ج
6.	ha	h	ح		ح	ح

No	Nama	Alih Aksara	Bentuk Terisolasi	Bentuk Tersambung		
				ا ح ا	Tengah	Akhir
7.	kha	kh	خ	خ	خ	خ
8.	dal	d	د	-	-	د
9.	zal	z	ذ	-	-	ذ
10.	ra'	r	ر	-	-	ر
11.	za'	z	ز	-	-	ز
12.	sīn	s	س	س	س	س
13.	syīn	sy	ش	ش	ش	ش
14.	shād	sh	ص	ط	ط	ص
15.	dād	d	ض	ظ	ظ	ض
16.	ta	t	ت	ت	ت	ت
17.	za	z	ز	-	-	ز
18.	'ain	'	ع	ع	ع	ع
19.	gaīn	g	غ	غ	غ	غ
20.	fa'	f	ف	ف	ف	ف
21.	qāf	q	ق	ق	ق	ق
22.	kāf	k	ك	ك	ك	ك
23.	lām	l	ل	ل	ل	ل
24.	mīm	m	م	م	م	م
25.	nūn	n	ن	ن	ن	ن
26.	wāwu	w	و	-	-	و
27.	ha'	h	ه	ه	ه	ه
28.	ya'	y	ي	ي	ي	ي
29.	hamzah	-	ء	-	-	-
30.	Tamarbuto	ta	ة	-	-	ة
31.	lām + alif	-	لا	-	-	لا

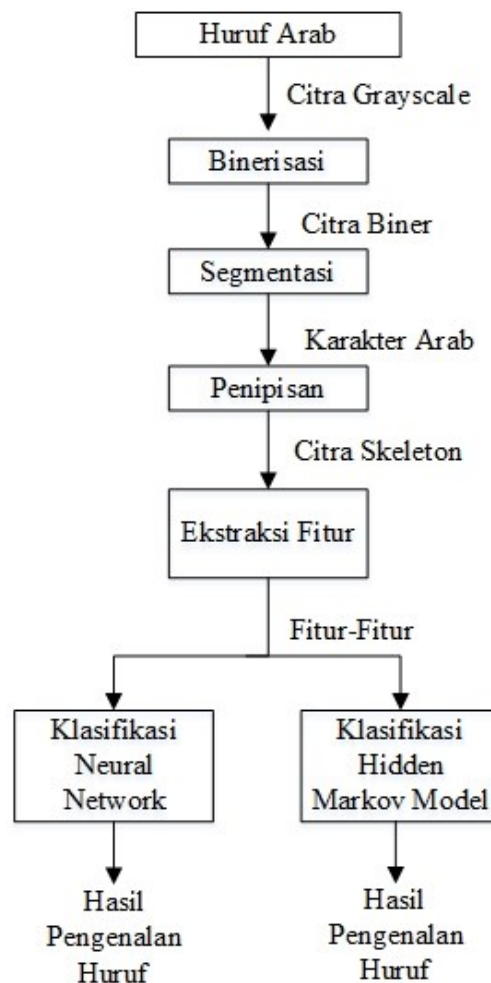
II.2 Pengenalan Huruf Arab

Konsep pengenalan huruf arab secara umumnya dapat dikategorikan dua bagian, yaitu pengenalan huruf terisolasi dan pengenalan huruf dalam kalimat. Pengenalan huruf

terisolasi berkaitan dengan proses mengenali huruf Arab yang terisolasi atau berdiri sendiri. Sedangkan pengenalan huruf dalam kalimat Arab adalah proses mengenali huruf yang terdapat dalam sebuah kalimat Arab. Oleh karena sifat tulisan Arab yang bersambung baik tulisan cetak maupun tulisan tangan, maka proses pengenalan huruf dalam kalimat Arab membutuhkan proses segmentasi, sehingga tulisan yang bersambung dapat terbagi menjadi huruf – huruf yang berdiri sendiri sesuai dengan posisi-nya pada kalimat, yakni di terisolasi, diawal, ditengah dan diakhir.

II.3 Proses Pengenalan Huruf Arab

Proses pengembangan sistem pengenalan huruf Arab terdiri dari lima tahap, 4 tahap akan dijelaskan pada bab ini dan 1 tahap akan dijelaskan pada bab III.



Gambar II.1 Diagram blok umum sistem pengenalan huruf Arab

II.3.1 Binerisasi

Binerisasi pada citra adalah proses merubah citra kedalam bentuk biner yang bernilai 0 dan 1. Citra *grayscale* akan dirubah menjadi hitam dan putih. Konversi citra berwarna ke biner ini dilakukan pada saat awal citra dibuka oleh sistem.

Cara yang dilakukan adalah dengan melakukan pengambangan (*threshold*) pada masing-masing kanal warna. Ambang yang digunakan adalah 150. Jika kanal warna bernilai kurang dari 150 maka akan diubah menjadi warna hitam, dan jika warna lebih dari 150 akan diubah menjadi warna putih.

II.3.2 Segmentasi

Segmentasi atau proses pemotongan adalah proses yang penting dalam sistem pengenalan huruf khususnya huruf Arab, dikarenakan kalimat bahasa Arab yang ditulis secara bersambung. Segmentasi memiliki tujuan untuk memisahkan huruf dalam citra. Setelah terpisah, masing-masing huruf dapat diidentifikasi lebih lanjut.

Segmentasi adalah berusaha memotong kata menjadi huruf per huruf khususnya pada tulisan bersambung seperti tulisan Arab. Dalam pengenalan huruf latin proses segmentasi dapat dilakukan dengan mudah karena huruf – huruf terpisah satu sama lain kecuali tulisan tangan. Berbeda dengan tulisan Arab yang memiliki sifat bersambung, baik ditulis dengan tangan maupun tulisan cetak. Oleh karena itu masalah segmentasi dalam tulisan arab perlu diperhatikan.

Pada penelitian ini segmentasi dilakukan dengan algoritma Zidouri^[7], berikut tahap dari penipisan algoritma zidouri.

Digunakan beberapa parameter dalam segmentasi yaitu sebagai berikut:

Φ = Lebar dari titik tunggal dari dokumen

L_s = lebar dari karakter terkecil

L_s' = Lebar dari karakter terkecil jika muncul bersamaan

L_m = Lebar maksimum karakter dalam bentuk terisolasi

$B(x, y)$ = Lokasi *baseline*

I = Citra dari kata

I' = Citra dari kata tanpa titik

E = Citra kosong dari ukuran I

Langkah – langkah segmentasi:

1. Lakukan penulangan terhadap I
2. Scan dari kanan ke kiri dalam baris-baris untuk menemukan sebuah *band* dari piksel horizontal yang memiliki lebar $\geq L_s$
3. Ambil proyeksi vertikal dari *band* yang di *scan* di langkah 2, jika tidak ada piksel ditemukan, jika tidak ada piksel ditemui gambar *guide band* vertikal pada E .
4. Gunakan tanda khusus untuk *guide bands*, yang mana digunakan pada langkah ke dua yaitu ketika menetapkan fitur keempat, bernilai 0 jika dibawah *baseline* $B(x,y)$
5. Ulangi prosedur untuk seluruh baris

Setelah itu akan diperoleh citra E dengan beberapa *guide band*. Kemudian dilakukan pemilihan *guide band* yang benar untuk segmentasi kata. Oleh karena itu beberapa fitur di ekstrak sebagai berikut.

F_1 : Lebar dari *guide band*

F_2 : Jarak 1 pendahulu dari kanan bernilai 0 dalam kasus *guide band* pertama.

F_3 : Jarak 2 pendahulu dari kanan, bernilai 0 dalam kasus *guide band* pertama dan kedua

F_4 : Posisi *guide band* ditemukan, bernilai 1 jika diatas lini basis dan bernilai 0 jika dibawah lini basis

F_5 : Titik tengah dari *guide band*

Setelah melakukan ekstrak 5 fitur, terdapat 4 aturan tambahan untuk memilih guide band dari masing-masing guide band diuji melalui 4 aturan. Jika memenuhi peraturan maka guide band dipilih, jika tidak maka ditolak. Aturan tersebut adalah sebagai berikut.

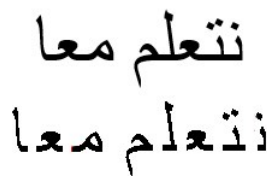
Aturan 1 : *Guide band* dipilih yang memiliki lebar tertinggi relatif untuk fitur F1 dan nilai $F4 = 1$

Aturan 2 : *Guide band* dipilih jika $F2 > Ls$ dan $F4 = 1$

Aturan 3 : *Guide band* dipilih jika $F2 \leq Ls$ dan $F3 > Ls'$ dan *guide band* bukan yang terakhir

Aturan 4 : *Guide band* dipilih jika $F1 \geq Lm$ dan $F4 = 1$

Setelah menetapkan ke-4 aturan tersebut, ada aturan tambahan yaitu jika *guide band* pertama gagal memenuhi aturan 1 sampai 4, jika *guide band* berikutnya memenuhi aturan nomor 2 maka *guide band* ini tetap dipilih. Setelah itu terdapat tambahan aturan lagi yaitu Jika semua *guide band* gagal memenuhi seluruh aturan, maka hapus kondisi F4 kecuali pada aturan nomor 4.



Gambar II.2 Hasil segmentasi dengan algoritma Zidouri

II.3.3 Penipisan

Penipisan adalah proses pengambilan tulang dari suatu pola, salah satu penggunaan penipisan adalah dalam aplikasi pengenalan pola. Citra yang digunakan adalah citra yang telah dilakukan binerisasi terlebih dahulu. Proses ini mengikis piksel sebanyak mungkin tanpa mempengaruhi bentuk umum. Citra hasil dari algoritma penipisan disebut dengan *skeleton*. Pada umumnya suatu algoritma penipisan yang dilakukan terhadap citra biner memiliki kriteria sebagai berikut :

1. *Skeleton* tetap menjaga struktur keterhubungan yang sama dengan citra awal.

2. *Skeleton* dari citra kira-kira berada di bagian tengah dari citra awal sebelum dilakukan penipisan.
3. *Skeleton* memiliki bentuk yang mirip dengan citra awal.
4. *Skeleton* mengandung jumlah piksel yang setipis mungkin namun tetap memenuhi kriteria sebelumnya.



Gambar II.3 Pola huruf dan hasil penipisan

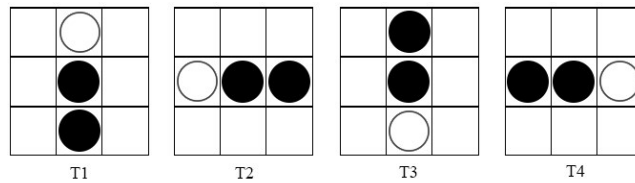
Penipisan berguna apabila tidak diperlukan pada ukuran dari pola melainkan pada posisi relatif goresan dari pola. Ada beberapa algoritma yang dirancang untuk tujuan ini. Terdapat beberapa algoritma penipisan yang populer diantaranya adalah *Zhang Suen* ^[9], *Stentiford* ^[6] dan *Hilditch* ^[10]. Dalam penelitian ini algoritma Stentiford dipilih sebagai algoritma penipisan terbaik. Setelah dilakukan perbandingan diantara algoritma *Zhang Suen*, *Stentiford* dan *Hilditch*. Gambar II.2 menunjukkan perbandingan hasil penipisan dari huruf "ث" dengan algoritma *Zhang Suen*, *Stentiford* dan *Hilditch*.



Gambar II.4 Perbandingan algoritma penipisan

Pada Gambar II.2 dapat dilihat hasil dari penipisan dengan algoritma *Zhang Suen* menghapus bagian kanan huruf "ث" yang seharusnya tidak terhapus, sebagaimana yang terdapat pada hasil penipisan dengan algoritma *Stentiford*. Sedangkan hasil dari penipisan *Hilditch* menghapus dua titik huruf "ث", sehingga huruf "ث" hanya memiliki 1 titik. Hasil penipisan dengan algoritma *Stentiford* terlihat sempurna tanpa ada kesalahan.

Pada algoritma *Stentiford* ada 4 buah *template* yang dipakai , template 3 x 3 yaitu :



Gambar II.5 Template algoritma Stentiford

Berikut adalah langkah – langkah yang dilakukan untuk mendapatkan *skeleton* dari suatu citra dengan algoritma *Stentiford*:

1. Cari lokasi piksel (i, j) yang cocok dengan template T₁. Pencocokkan template ini bergerak dari kiri ke kanan dan dari atas ke bawah.
2. Bila piksel tengah bukan merupakan endpoint dan mempunyai jumlah konektivitas 1, maka tandai pixel untuk kemudian dihapus.
 - a. *Endpoint* adalah piksel yang merupakan batas akhir dan hanya terhubung 1 piksel saja. Artinya, jika piksel hitam hanya memiliki satu tetangga hitam dari delapan kemungkinan tetangga.
 - b. Jumlah konektivitas adalah ukuran berapa banyak objek yang terhubung dengan piksel tertentu. Berikut adalah rumus untuk menghitung jumlah konektivitas.

$$C_n = \sum_{k \in S} N_k - (N_k \cdot N_{k+1} + N_{k+1} \cdot N_{k+2}) \quad \dots(\text{II.1})$$

Dimana:

N_k merupakan nilai dari 8 tetangga di sekitar piksel yang akan dianalisa, dan nilai $S = \{1, 3, 5, 7\}$

N_0 adalah nilai dari piksel tengah.

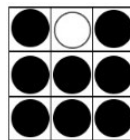
N_1 adalah nilai dari piksel pada sebelah kanan central pixel dan sisanya diberi nomor berurutan dengan arah berlawanan jarum jam

4	3	2
5	0	1
6	7	8

Gambar II.6 Penomoran nilai k pada algoritma Stentiford

3. Ulangi langkah 1 dan 2 untuk semua piksel yang cocok dengan template T₁.

4. Ulangi langkah 1–3 untuk template T2, T3 dan T4.
 - a. Pencocokkan template T2 akan dilakukan pada sisi kiri dari obyek dengan arah dari bawah ke atas dan dari kiri ke kanan.
 - b. Pencocokkan template T3 akan dilakukan pada sisi bawah dari obyek dengan arah dari kanan ke kiri dan dari bawah ke atas.
 - c. Pencocokkan template T4 akan dilakukan pada sisi kanan dari obyek dengan arah dari atas ke bawah dan dari kanan ke kiri.
5. Piksel yang ditandai untuk dihapus diubah menjadi putih.



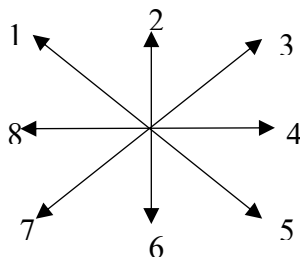
Gambar II.7 Jumlah konektivitas 1 algoritma Stentiford

II.3.4 Tahap Ekstraksi Fitur

Pada penelitian ini terdapat tiga tahap ekstraksi fitur, yaitu fitur *chain code* yang dinormalisasi, jumlah titik dan posisi titik. Berikut adalah penjelasan dari masing – masing fitur.

II.3.4.1 Chaincode

Pada pengenalan pola, chaincode merupakan suatu teknik untuk menggambarkan suatu struktur dari suatu objek. *Chain code* diperoleh dengan cara menelusuri piksel batas objek berdasarkan arah-arah yang telah ditetapkan. Hasil dari *chain code* adalah angka – angka yang menunjukkan arah yang mewakili batas objek. Pencarian *chain code* hanya bisa dilakukan pada citra biner.



Gambar II.8 Arah mata angin chain code

Berikut adalah cara pengambilan chain code dari suatu objek di dalam sebuah citra:

1. Cari piksel hitam yang hanya memiliki 1 tetangga dengan cara menelusuri piksel pada citra dimulai dari pojok kiri atas sampai menemukan piksel yang berwarna hitam, jika tidak ditemukan piksel hitam yang hanya memiliki 1 tetangga maka ambil piksel hitam pertama yang ditemui.
2. Lakukan iterasi pada gambar
 - a. Ubah piksel saat ini menjadi 0
 - b. Ikuti prioritas arah 1 sampai 8
 - c. Pindah posisi piksel
 - d. Tambahkan arah ke chain code

Panjang chain code dari suatu objek dapat berubah – ubah sesuai bentuk dari suatu objek tersebut. Untuk menjaga konsistensi pada penelitian ini akan dilakukan normalisasi chain code, yaitu membuat panjang chain code dari suatu citra tetap jumlahnya menjadi 10 angka untuk setiap huruf.

II.3.4.1.1 Normalisasi Chaincode

Normalisasi suatu chain code dilakukan agar jumlah chain code yang pada mulanya berubah – ubah sesuai bentuk dari suatu objek dapat dijadikan tetap. Pada penelitian ini panjang chain code dari setiap citra huruf Arab akan ditetapkan menjadi 10.

Berikut adalah langkah – langkah dari normalisasi chain code :

1. Chain code diubah menjadi matriks 2 dimensi. Baris pertama adalah nilai dari chain code. Baris kedua adalah frekuensi terjadinya setiap angka dalam chain code^[5].

Seperti chain code berikut : 7777311122222583353333, setelah dilakukan tahap pertama dari normalisasi chain code akan menjadi 2 x 9 matriks :

```
7 3 1 2 5 8 3 5 3
4 1 3 5 1 1 2 1 4
```

2. Hilangkan semua nilai yang memiliki frekuensi hanya 1^[5].

```
7 3 1 2 5 8 3 5 3   ➡   7 1 2 3
4 1 3 5 1 1 2 1 4   ➡   4 3 5 6
```

3. Tampilkan chaincode sesuai dengan frekuensi terjadinya :

777711122222333333

4. Lakukan pemetaan terhadap chaincode:

FTC: 777711122222333333

Rumus Normalisasi Chain Code:

$NC[i] = FTC \lfloor \text{round}(i/9 \times \text{FTC.length}-1) \rfloor$

Keterangan:

FTC = Frekuensi Terjadinya Chain code

NC = Chain code yang ternormalisasi

Chaincode yang ternormalisasi: 7711222333

II.3.4.2 Fitur Jumlah Titik

Fitur jumlah titik merupakan fitur yang penting dalam huruf Arab, karena beberapa huruf Arab memiliki bentuk yang sama namun hanya dibedakan berdasarkan jumlah titik. Seperti huruf ب, ث, dan ت.

Jumlah titik diperoleh dengan cara melakukan iterasi pada gambar huruf dari sudut kiri atas ke kanan, kemudian kebawah, jika ditemukan titik hitam pertama hitung *chain code* dari titik hitam tersebut. Huruf yang memiliki titik akan memiliki *chain code* lebih dari 1. Kemudian akan diperiksa, jika *chain code* yang ditemukan kurang dari 7 maka akan dihitung sebagai *chain code* titik, dan dilakukan penjumlahan terhadap jumlah titik. Jika *chain code* memiliki panjang lebih dari 7 maka akan dihitung sebagai *chain code* dari *body* huruf.

Seperti pada huruf, ث yang memiliki 4 *chain code*. *Chain code* pertama adalah “6” , *chain code* kedua adalah “6”, *chain code* ketiga adalah “6” dan *chain code* keempat adalah “66666667654545444488881818178787880”. Dari 4 *chain code* tersebut didapat 3 *chain code* yang memiliki panjang kurang dari 7. Kemudian setiap kemunculan *chain code* yang kurang dari 7 akan dilakukan perhitungan jumlah titik. Dari 4 *chain code* pada huruf ث didapat jumlah titik adalah 3.



Gambar II.9 Fitur jumlah titik

II.3.4.3 Fitur Posisi Titik

Fitur posisi titik merupakan hal yang penting pada huruf Arab. Beberapa huruf Arab memiliki bentuk dan jumlah titik yang sama, namun dibedakan berdasarkan posisi dari titik tersebut.

Posisi titik diperoleh dengan cara menentukan posisi titik dan tinggi huruf, kemudian gambar akan dibagi menjadi menjadi 5 bagian. Jika titik berada pada posisi kurang dari $\frac{2}{5}$ tinggi gambar maka posisi titik adalah diatas yang diwakili dengan angka 0. Jika titik berada pada posisi kurang dari $\frac{3}{5}$ tinggi gambar maka posisi titik berada ditengah yang diwakili dengan angka 1. Jika titik berada pada posisi lebih dari $\frac{3}{5}$ tinggi gambar maka posisi titik adalah dibawah yang diwakili dengan angka 2.



Gambar II.10 Cara menentukan posisi titik

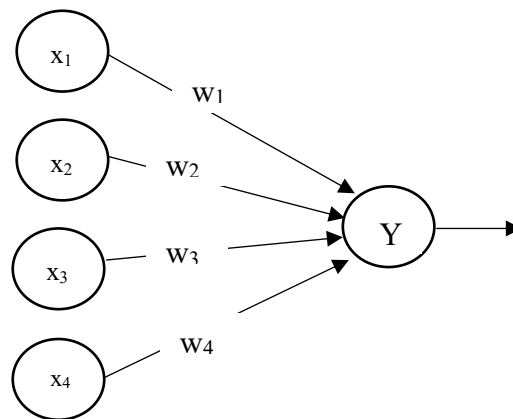
II.4 Artificial Neural Network

II.4.1 Jaringan Syaraf Tiruan

Jaringan syaraf tiruan (*Artificial Neural Network*) merupakan suatu sistem komputasi yang struktur jaringannya meniru sistem syaraf manusia agar dapat menghasilkan respon dan perilaku seperti dengan jaringan syaraf biologis. Sistem pemrosesan

informasi pada jaringan syaraf tiruan memiliki ciri – ciri yang mirip jaringan syaraf biologi.

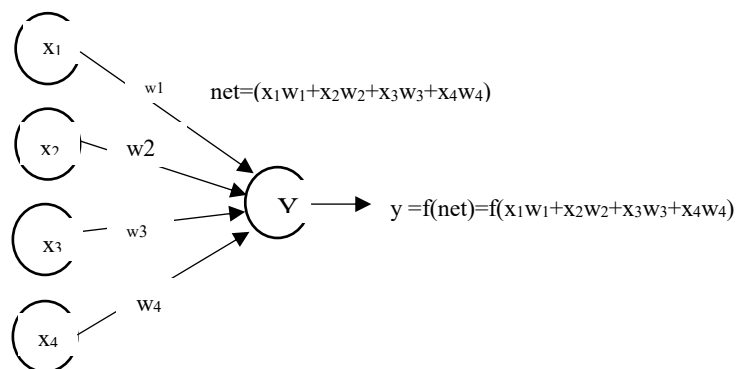
II.4.2 Model Neuron



Gambar II.11 Model neuron

Neuron adalah unit pemproses informasi utama dari jaringan syaraf tiruan yang bekerja berdasarkan impuls atau sinyal yang diterimanya dan diteruskan ke neuron lain. Neuron terdiri dari tiga elemen utama, yaitu :

1. Kumpulan unit-unit yang dihubungkan dengan jalur hubungan
2. Unit penjumlah yang menjumlahkan sinyal masukan yang sudah dikalikan dengan bobotnya. Fungsi aktivasi yang digunakan untuk menentukan keluaran dari suatu neuron, yakni menentukan apakah sinyal dari masukan neuron akan diteruskan ke neuron lain atau tidak



Gambar II.12 Perhitungan bobot dan fungsi aktivasi neuron

Keterangan :

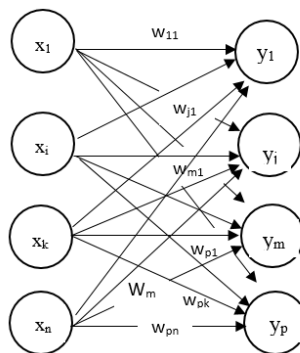
1. Y menerima masukan dari neuron x_1, x_2, x_3 , dan x_4 .
2. Neuron – neuron tersebut akan melakukan transformasi pada informasi yang diterima melalui sambungan keluarnya menuju neuron-neuron lain. Pada jaringan syaraf tiruan hubungan ini disebut dengan bobot. Setiap neuron memiliki bobot yang ditandai dengan w_1, w_2, w_3 dan w_4 . Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut.
3. Informasi masukan akan dikirim ke neuron dengan nilai bobot tertentu. Masukan akan di proses oleh suatu fungsi yang menjumlahkan nilai – nilai semua bobot yang datang. Pada gambar ketiga impuls neuron dijumlahkan :

$$\text{net} = (x_1w_1+x_2w_2+x_3w_3+x_4w_4) \quad \dots(\text{II.2})$$
4. Hasil penjumlahan akan dibandingkan dengan suatu nilai ambang (*threshold*) melalui fungsi aktivasi setiap neuron. Besarnya impuls yang diterima Y mengikuti fungsi aktivasi $y = f(\text{net})$.
5. Jika nilai fungsi aktivasi cukup kuat, yakni melewati nilai ambang tertentu, sinyal akan diteruskan ke neuron lain.

II.4.3 Arsitektur jaringan

Pola hubungan antar neuron pada neural network terbagi menjadi dua yaitu jaringan layer tunggal dan jaringan layer jamak

1. Jaringan layer tunggal (*single layer network*)

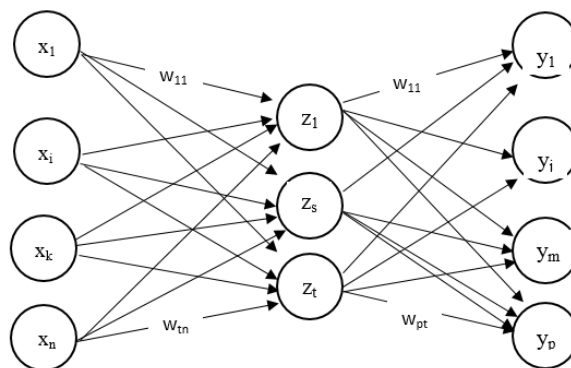


Gambar II.13 Jaringan layer tunggal

Jaringan layer tunggal memiliki satu layer bobot terhubung. Pada layer ini sekumpulan neuron masukan dihubungkan langsung dengan sekumpulan neuron keluaran. Dimana unit masukan merupakan unit yang menerima sinyal sedangkan unit output adalah unit dimana respon dari jaringan terlihat. Arsitektur jaringan tersusun dari sejumlah unit input dan sejumlah unit output. Semua unit input terhubung ke semua unit output dengan bobot berbeda. bobot w_{p1} menyatakan bobot antara unit 1 pada input dengan unit p pada output.

2. Jaringan layer jamak (multi layer network)

Jaringan layer jamak memiliki satu atau lebih diantara lapisan masukan dan keluaran, lapisan ini disebut lapisan tersembunyi (*hidden layer*). Jaringan layer jamak memiliki kelebihan dapat menyelesaikan masalah yang lebih kompleks jika dibandingkan dengan jaringan layer tunggal. Namun memiliki proses pelatihan yang lebih lama dan lebih sulit.



Gambar II.14 Jaringan layer jamak

II.4.4 Training

Training adalah proses pembelajaran yang dilakukan dengan cara mencari nilai dari bobot – bobot yang menjadi penghubung seluruh neuron, sehingga data input dapat menghasilkan keluaran yang diinginkan. Training dilakukan dengan cara berulang – ulang hingga mendapatkan bobot yang paling optimal.

Training dapat dibagi menjadi dua kategori yaitu supervised learning dan unsupervised learning berikut adalah penjelasannya.

1. Supervised Learning

Supervised learning adalah metode pembelajaran yang sudah terdapat data latih sebagai masukan dan terdapat target keluaran. Data masukan dan target digunakan untuk melatih jaringan sehingga didapatkan bobot yang diinginkan. Sehingga jaringan dapat melakukan pemetaan dari input ke output sesuai yang diinginkan.

Tahap – tahap yang dilakukan setiap pelatihan pada *Supervised Learning* :

1. Suatu masukan diberikan ke jaringan
2. Jaringan memproses dan mengeluarkan keluaran
3. Selisih antara keluaran jaringan dan target adalah kesalahan yang terjadi, disebut dengan loss. Semakin kecil nilai dari loss maka proses training semakin bagus
4. Jaringan memodifikasi bobot sesuai dengan loss tersebut, ulangi proses di atas

2. Unsupervised Learning

Pada *unsupervised learning* tidak ada data target yang mengarahkan proses pelatihan. Perubahan bobot jaringan dilakukan berdasarkan parameter tertentu. Jaringan dimodifikasi menurut ukuran parameter tersebut.

II.4.5 Fungsi aktivasi

Fungsi aktivasi digunakan untuk menentukan keluaran suatu neuron. Jika net $\sum x_i w_i$, maka fungsi aktivasinya adalah $f(\text{net}) = f(\sum x_i w_i)$.

Berikut adalah beberapa jenis fungsi aktivasi.

1. Fungsi aktivasi step/threshold :

Digunakan pada awal pengembangan neural network, tidak dapat menyelesaikan masalah yang tidak linier.

Persamaan fungsi aktivasi step :

$$F(x) = \begin{cases} 0; & x \leq 0 \\ 1; & x > 0 \end{cases} \quad \dots(\text{II.3})$$

2. Fungsi Sigmoid

Persamaan fungsi aktivasi sigmoid :

$$F(x) = \frac{1}{(1+e^{-x})} \quad \dots(\text{II.4})$$

Cocok untuk menyelesaikan masalah yang tidak linier

3. Fungsi Identitas

Persamaan fungsi aktivasi identitas :

$$F(x) = x \quad \dots(\text{II.5})$$

Seringkali digunakan pada neuron keluaran

4. Fungsi Softmax

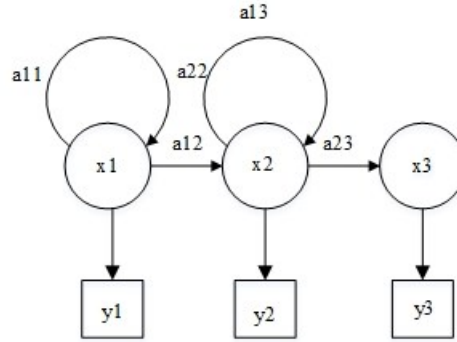
$$F(X_i) = \frac{e^{X_i}}{(\sum_{j=0}^k e^{X_j})} \quad \dots(\text{II.6})$$

Dimana:

$$I = 0, 1, 2 \dots k$$

II.5 Hidden Markov Model

Hidden Markov Model (HMM) adalah perluasan dari rantai Markov dimana statenya tidak dapat diamati secara langsung atau tersembunyi, tetapi hanya dapat diobservasi melalui pengamatan variabel lain. Terdapat tiga masalah utama yang dapat diselesaikan oleh Hidden Markov Model yaitu tahap evaluasi, tahap decoding dan learning.



Gambar II. 15 Hidden Markov Model

II.5.1 Evaluasi

Operasi evaluasi dalam HMM adalah perhitungan probabilitas dari urutan nilai observasi yang diberikan oleh *Hidden Markov Model*. Algoritma yang banyak digunakan untuk penyelesaian masalah observasi adalah algoritma maju (*Forward Algorithm*) yang keadaannya mengalir kedepan. Algoritma mundur (*Backward Algorithm*) yang keadaannya mengalir kebelakang dari observasi terakhir pada saat T, dan algoritma maju – mundur (*forward-backward algorithm*) yang merupakan gabungan dari algoritma forward – backward [8].

1. Algoritma maju

Algoritma forward terdiri dari tiga bagian yakni inisialisasi, rekursi dan terminasi.

Inisialisasi :

$$\alpha_t(i) = \pi_i b_i(O_1), 1 \leq i \leq n \quad \dots(\text{II.7})$$

dimana π = matrik state awal dan $b_i(O_1)$ = matriks pertama yang terobservasi.

Rekursi :

$$\alpha_{t+1}(j) = \sum_{i=1}^{n_i} \alpha_t(i) a_{ij} b_j(\alpha_t), 1 \leq i \leq n \quad \dots(\text{II.8})$$

dimana :

n_i = jumlah state

a_{ij} = matrik transisi.

Terminasi :

$$P(O|\lambda) = \sum_{i=1}^{n_i} \alpha_t(i) \quad \dots(\text{II.9})$$

2. Algoritma mundur

Algoritma mundur terdiri dari tiga tahap yaitu sebagai berikut.

dan dianalogikan dengan prosedur forward $\beta_t(i)$ dengan langkah:

Inisialisasi:

$$\beta_t(i) = 1 \quad 1 \leq i \leq n \quad \dots(\text{II.10})$$

Rekursi:

$$\beta_t(i) = \sum a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad \dots(\text{II.11})$$

$$t = T - 1, T - 2, \dots, 1, \quad 1 \leq i \leq n \quad \dots(\text{II.12})$$

II.5.2 Pengkodean (Decoding)

Decoding dilakukan untuk mencari *state* yang terbaik dari urutan observasi pada model HMM dengan algoritma *viterbi*. Langkah – langkahnya sebagai berikut^[8].

Inisialisasi :

$$\delta_t(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq n \quad \dots(\text{II.13})$$

Rekursi :

$$\delta_t(j) = \max_{1 \leq i \leq n} [\delta_{t-1}(i) a_{ij}] b_j(O_{t+1}) \quad , \quad 1 \leq j \leq n \quad \dots(\text{II.14})$$

Terminasi :

$$P = \max_{1 \leq i \leq n} [\delta_t(i)] \quad \dots(\text{II.15})$$

Dimana :

P = Probabilitas

II.5.3 Learning

Operasi learning dalam HMM adalah untuk memperoleh parameter pada model HMM.

Untuk menyelesaikan proses learning ini digunakan algoritma Baum-Welch.

BAB III

PERANCANGAN DAN EKSPERIMEN

III.1 Perancangan Awal Sistem dan Eksperimen Pengenalan Huruf Arab

Dalam tesis ini akan dibuat sebuah perangkat lunak pengenalan huruf Arab yang bisa mengenali huruf Arab terisolasi dan dalam kalimat. Masukan dari sistem ialah berupa citra yang berisi huruf Arab dalam bentuk terisolasi dan dalam kalimat kemudian melalui lima tahap, tahap pertama yaitu binerisasi. Pada pengenalan huruf Arab terisolasi setelah melalui proses binerisasi akan dilanjutkan ke tahap penipisan, kemudian fitur dari tiap huruf akan di ekstrak melalui chain code, jumlah titik dan posisi titik. Setelah fitur di ekstrak akan dilakukan klasifikasi menggunakan Neural Network dan Hidden Markov Model. Sedangkan pada pengenalan huruf Arab dalam kalimat setelah proses binerisasi akan melalui tahap segmentasi, kemudian dilanjutkan pada tahap penipisan, lalu akan di ekstrak fitur setiap huruf dengan chain code, jumlah titik dan posisi titik. Setelah fitur di ekstrak akan dilakukan klasifikasi menggunakan neural network dan hidden markov model. Gambaran umum tahapan proses pengenalan huruf Arab dapat dilihat pada Gambar III.1

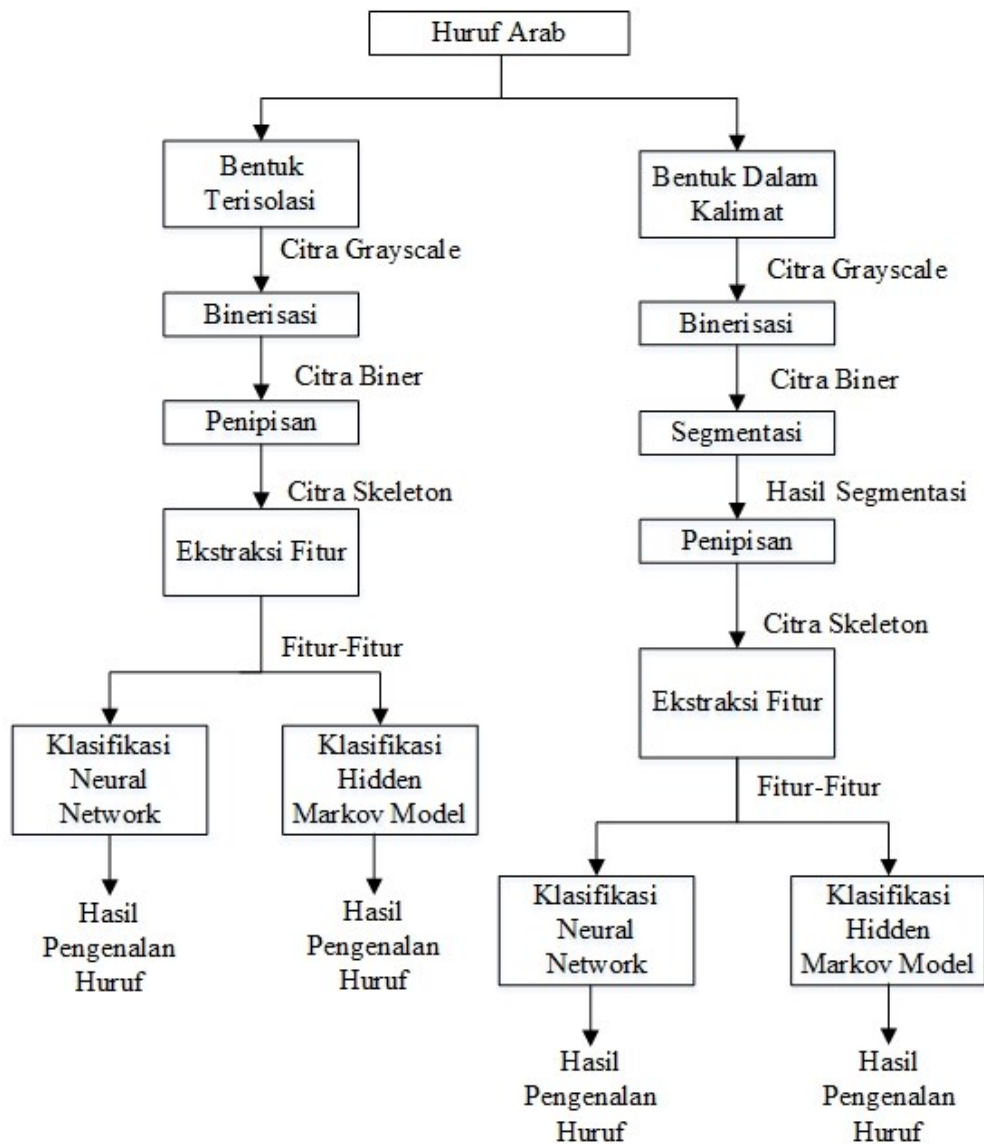
Dalam penelitian ini dikenal data latih dan data uji. Berikut penjelasan dari kedua jenis data tersebut.

1. Data latih

Data latih merupakan kumpulan data huruf arab sesuai sesuai posisinya pada kalimat yakni terisolasi, di awal, di tengah, dan di akhir yang akan dilatih untuk kemudian disimpan sebagai data latih huruf Arab.

2. Data uji

Data uji terbagi dua, yaitu data uji huruf Arab terisolasi dan kumpulan data kalimat Arab yang akan dimasukkan dalam tahap pengujian dan dibandingkan data latih untuk ditentukan hasil pengenalan huruf Arab tersebut.



Gambar III. 1 Diagram blok utama dalam sistem pengenalan huruf Arab

Data latih diambil dari 31 huruf Arab berdasarkan posisinya dalam kata yang memiliki dua sampai empat bentuk, yaitu terisolasi, di awal, di tengah dan di akhir. Masing-masing data latih dan data uji dibuat dalam tiga font, yaitu font Arial Unicode Ms, font Tahoma dan font Time New Roman. Total data latih adalah 318 huruf Arab. Data uji untuk pengenalan huruf Arab terisolasi adalah 31 huruf Arab yang diambil dalam tiga font sehingga total data uji adalah 93 huruf Arab. Sedangkan Data uji untuk pengenalan huruf Arab dalam kalimat adalah 10 kalimat yang diambil dalam tiga font, sehingga total data uji adalah 30 kalimat.

Berikut teknik yang digunakan dalam pengambilan data latih.

1. Ketik huruf 31 huruf Arab menggunakan Microsoft Word , huruf arab ditulis sesuai posisinya pada kalimat, yakni posisi terisolasi, di awal, di tengah dan di akhir.
2. Potong setiap huruf menggunakan *Snipping Tool*
3. Hasil pemotongan huruf disimpan sebagai data latih huruf arab

Saat pengambilan data uji, dilakukan dalam dua tahap, yaitu:

1. Pengujian huruf Arab terisolasi

Saat pengujian huruf ini dilakukan sebanyak 31 huruf arab terisolasi yang terdapat pada tabel II.1. Pangujian huruf Arab terisolasi dilakukan dalam waktu 13 detik.

2. Pengujian kalimat

Saat pengujian kalimat dilakukan dengan memasukkan 10 kalimat yang berbeda kedalam sistem. Pengujian kalimat dilakukan dalam waktu 11 detik. Berikut 10 kalimat Arab yang akan diujikan kedalam sistem :

- a. الاحترام المتبادل بين الاديان
- b. لقمر تبدو جميلة جدا
- c. ف كهوة دوريان ا لا ذواق جيدة
- d. تتخلي ابداء في الحياة
- e. انتظر اي محاكمة
- f. لحفاظ علي صحتك حياة طيبة
- g. المعلمين يعلمون
- h. جدة الارز المطبوخ
- i. يتكلم ببطء مفهومة حتي
- j. الحفاظ علي صحتك حياة طيبة

III.2 Perancangan Sistem dan Eksperimen

Pada penelitian ini terdapat 5 tahap yang dilakukan untuk membuat sistem pengenalan huruf Arab, berikut adalah penjelasan masing – masing tahap.

III.2.1 Eksperimen Binerisasi

Binarisasi gambar adalah proses pengubahan citra menjadi biner yang memiliki nilai 0 dan 1. Citra *grayscale* akan berubah menjadi hitam putih. Proses binerisasi diperlukan untuk melakukan langkah selanjutnya pada pengenalan huruf dan kalimat Arab. Cara melakukannya adalah dengan melakukan ambang atau *threshold* pada setiap kanal warna. Ambang yang digunakan adalah 150. Jika kanal warna kurang dari 150 maka akan dikonversi menjadi hitam, dan jika warnanya lebih dari 150 akan berubah menjadi putih.

III.2.2 Eksperimen Segmentasi

Pada pengenalan huruf Arab dalam kalimat melalui tahap segmentasi terlebih dahulu sebelum tahap penipisan. Segmentasi dilakukan dengan menggunakan algoritma Zidouri ^[7]. Langkah – langkah dari algoritma Zidouri telah dijelaskan pada bab II.3.2

III.2.3 Eksperimen Penipisan

Setelah diperoleh citra biner, dilakukan proses penipisan untuk mendapatkan tulang dari huruf. Proses penipisan dilakukan dengan algoritma Stentiford. Langkah – langkah dari algoritma Stentiford^[6] telah dijelaskan pada bab II.3.3.

III.2.4 Eksperimen Ekstraksi Fitur

Ekstraksi fitur dilakukan dalam 3 tahap yakni sebagai berikut :

1. Chain code

Setelah melalui tahap penipisan fitur chain code di ekstrak untuk setiap huruf Arab kemudian dilakukan normalisasi pada setiap chain code sehingga panjang chain code hanya 10 untuk setiap huruf. Langkah – langkah untuk mengambil chain code pada suatu objek telah dijelaskan pada bab II.4.1, dan langkah – langkah untuk melakukan normalisasi chain code telah dijelaskan pada tahap II.4.1.1

2. Jumlah titik

Setelah fitur chaincode fitur jumlah titik merupakan fitur yang penting karena beberapa huruf arab memiliki kemiripan bentuk namun dapat di bedakan berdasarkan jumlah titik, jumlah titik terbagi menjadi empat , yaitu 0, 1, 2 dan 3.

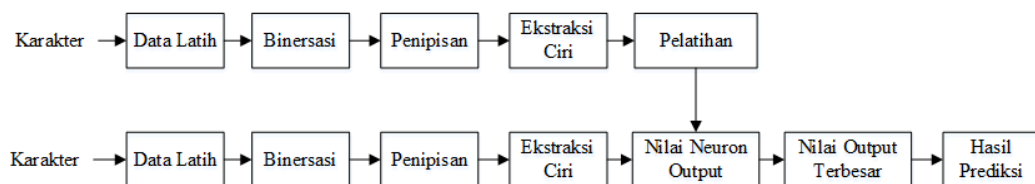
3. Posisi titik

Setelah fitur chaincode dan jumlah titik, fitur posisi titik juga merupakan fitur yang penting karena beberapa huruf arab memiliki kemiripan bentuk namun dapat di bedakan berdasarkan posisi titik, posisi titik terbagi menjadi tiga, yaitu di atas, ditengah dan di bawah.

III.3 Perancangan Sistem dan Eksperimen Pengklasifikasi

III.3.1 Tahap Pelatihan dan Pengujian dengan Metode Neural Network

Proses pengenalan huruf Arab pada tahap klasifikasi dengan metode neural network dapat dilihat pada gambar di bawah ini.



Gambar III.2 Proses pengenalan huruf Arab dengan metode *Neural Network*

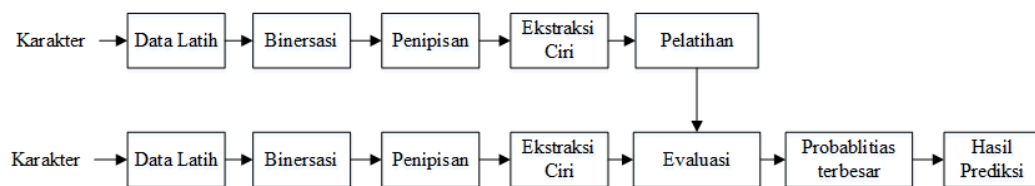
Tahap pelatihan yang dibangun metode *Neural Network* menggunakan metode *Backpropagation* yang merupakan salah satu metode learning dalam *Neural Network*. Pada *Backpropagation* terdapat cara update bobot secara khusus, bobot di update secara terus menerus sampai output neuron mendekati sama dengan target. Update bobot berhenti saat epoch telah mencapai batasnya.

Pada penelitian ini jenis arsitektur neural network yang digunakan adalah jaringan layer jamak dengan 1 *hidden layer*. Fungsi aktivasi yang digunakan untuk *hidden layer* adalah fungsi aktivasi *Sigmoid*, dan fungsi aktivasi yang digunakan untuk *output layer* adalah fungsi aktivasi *Softmax*. Masukan neuron untuk setiap sampel adalah 12, neuron pertama adalah jumlah titik, neuron kedua adalah posisi titik-titik dan neuron

ketiga adalah chain code yang telah dinormalisasi, batas iterasi yang digunakan adalah 10.000.

III.3.2 Tahap Pelatihan dan Pengujian dengan Metode Hidden Markov Model

Proses pengenalan huruf Arab pada tahap klasifikasi dengan metode *Hidden Markov Model* dapat dilihat pada gambar di bawah ini.



Gambar III.3 Proses pengenalan huruf Arab dengan metode *Hidden Markov Model*

Tahap pelatihan dilakukan dengan menentukan parameter estimasi, sehingga terbentuk hidden markov model berupa $\lambda = (A, B, \Pi)$, dilakukan dengan algoritma *Baum Welch*.

Data latih yang disimpan di ekstrak fiturnya menggunakan *chain code*, jumlah titik dan posisi titik. Keluaran dari ekstraksi fitur adalah jumlah titik, posisi titik dan *chain code* yang telah ternormalisasi menjadi 10 angka untuk setiap huruf. Hasil ekstraksi fitur ini menjadi masukan oleh proses pelatihan dengan algoritma *Baum Welch*. *Hidden state* dalam penelitian ini adalah label id atau nomor urut dari huruf Arab. *Observed sequence* adalah jumlah titik, posisi titik dan *chain code* yang telah dinormalisasi.

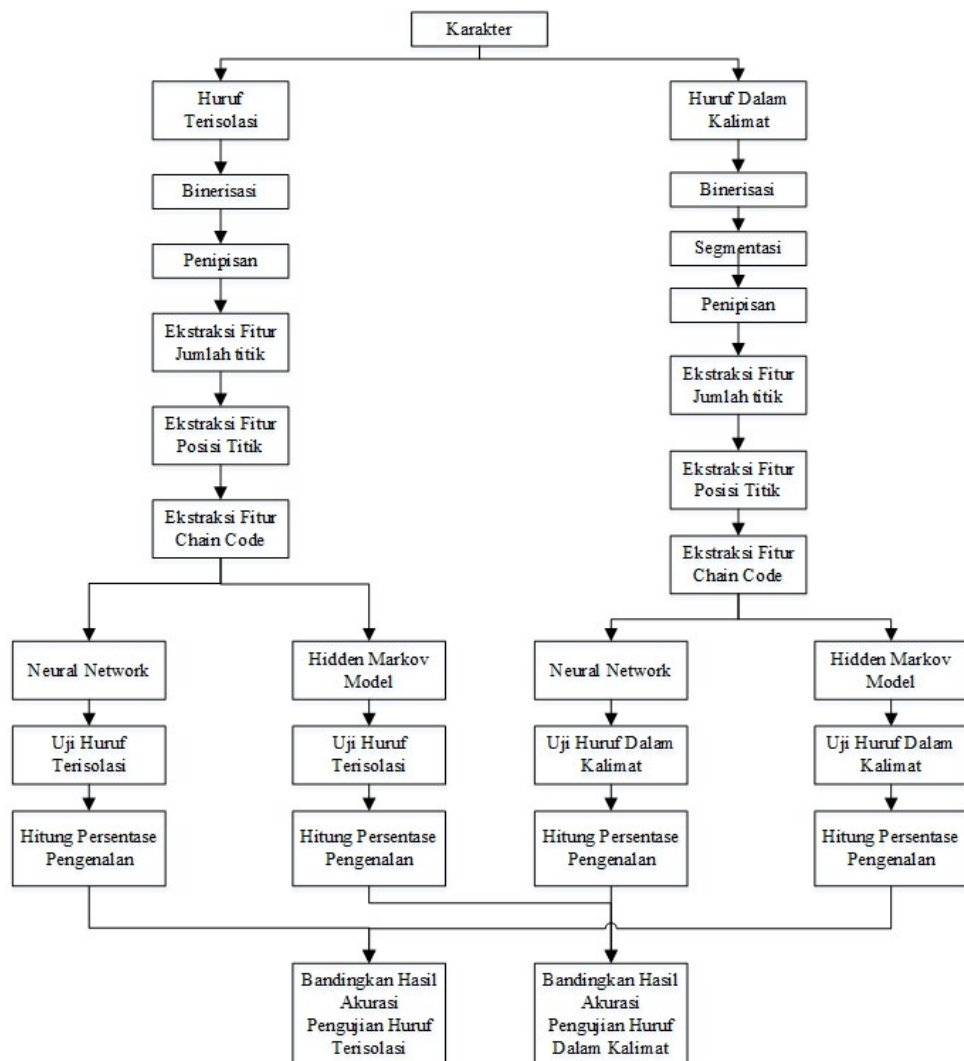
Pada proses Pengujian atau evaluasi menggunakan algoritma maju karena akan dicari nilai peluang yang paling tinggi dan menentukan kecocokan antara data uji dan data latih. Nilai peluang yang paling tinggi akan dipilih sebagai huruf hasil pengenalan dari masukan huruf yang diujikan.

BAB IV

PENGUJIAN DAN PEMBAHASAN

IV.1 Pengujian Sistem

Pengujian dilaksanakan dengan membandingkan hasil eksperimen dari sistem pengenalan huruf arab dengan menggunakan metode *Neural Network* dan *Hidden Markov Model*. Hasil eksperimen berupa nilai akurasi dari masing – masing hasil pengenalan huruf Arab. Berikut ini adalah diagram blok dari sistem pengenalan huruf Arab.



Gambar IV.1 Diagram blok utama pengujian pengenalan huruf Arab

Berikut penjelasan dari diagram blok di atas.

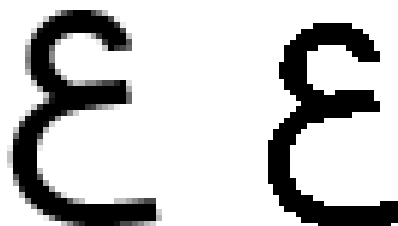
1. Citra huruf Arab *grayscale* dijadikan dalam bentuk biner
2. Huruf Arab terisolasi akan langsung masuk ke proses penipisan sedangkan huruf Arab dalam kalimat akan mengalami proses segmentasi kemudian masuk ke proses penipisan.
3. Huruf arab di ekstrak fiturnya dengan mengambil *chain code* yang di normalisasi, jumlah titik dan posisi titik dari setiap huruf, kemudian dilakukan pengujian dengan metode *Neural Network* dan *Hidden Markov Model*.
4. Masing-masing metode, akan dihasilkan berupa hasil pengenalan yang akan diujikan dalam 3 font berbeda, dan diujikan dalam bentuk huruf Arab terisolasi dan huruf Arab dalam kalimat.
5. Membandingkan tingkat akurasi dari masing-masing metode untuk huruf Arab terisolasi dan huruf Arab dalam kalimat.

IV.1.1 Pengujian Sistem Pengenalan Huruf Arab

Pengembangan sistem pengenalan huruf Arab terdiri dari 5 tahap: binerisasi, segmentasi, penipisan, ekstraksi fitur dan klasifikasi. Berikut adalah hasil pengujian tiap-tiap tahap.

IV.1.1.2 Hasil Pengujian Binerisasi

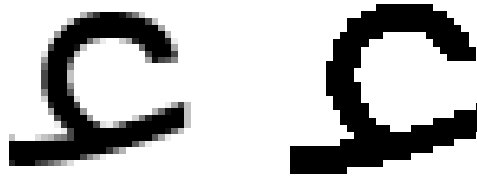
Proses binerisasi adalah proses merubah citra grayscale menjadi citra yang memiliki warna hitam dan putih, yaitu memiliki nilai 0 dan 1. Berikut adalah Hasil pengujian binerisasi untuk huruf Ain dengan font Arial dalam empat bentuk posisi yaitu terisolasi, di awal, di tengah dan diakhir.



Sebelum binerisasi

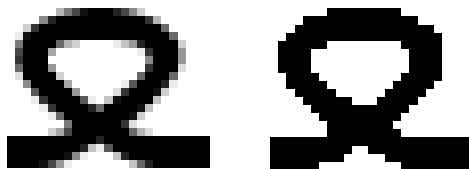
Sesudah binerisasi

Gambar IV.2 Citra biner huruf Ain terisolasi



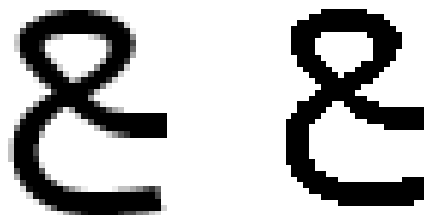
Sebelum binerisasi Sesudah binerisasi

Gambar IV.3 Citra biner huruf Ain di awal



Sebelum binerisasi Sesudah binerisasi

Gambar IV.4 Citra biner huruf Ain di tengah



Sebelum binerisasi Sesudah binerisasi

Gambar IV.5 Citra biner huruf Ain di akhir

Dalam penelitian ini binerisasi berhasil dilakukan pada seluruh data latih dan data uji, sehingga proses pengenalan huruf Arab dapat dilanjutkan ketahap berikutnya.

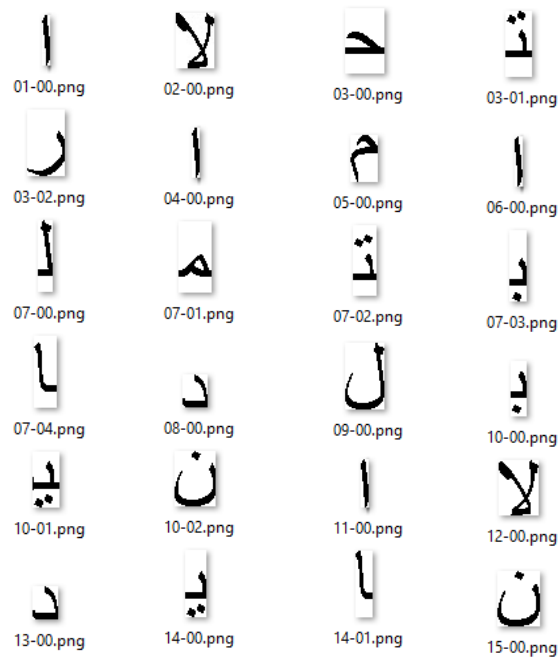
IV.1.1.2 Hasil Pengujian Segmentasi

Segmentasi dilakukan terhadap pengenalan huruf Arab dalam kalimat. Pada penelitian ini segmentasi menggunakan algoritma Zidouri. Berikut Hasil Segmentasi tiga kalimat.

1. Segmentasi kalimat satu

Kalimat: الاحترام المتبادل بين الاديان

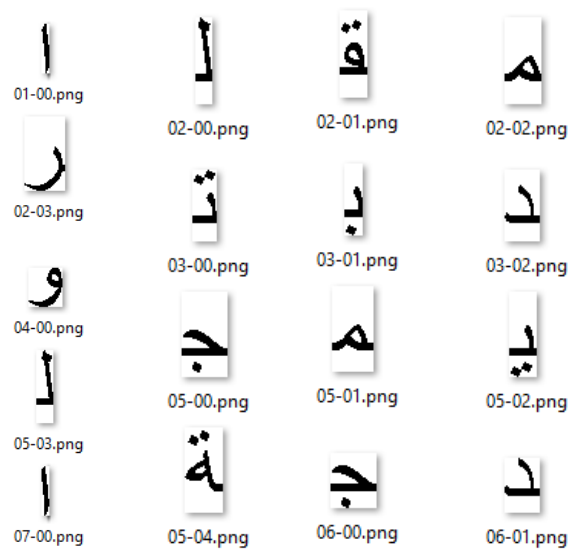
Hasil segmentasi kalimat satu:



Gambar IV.6 Hasil segmentasi kalimat satu

2. Hasil segmentasi kalimat dua

Kalimat : القمر تبدو جميلة جدا

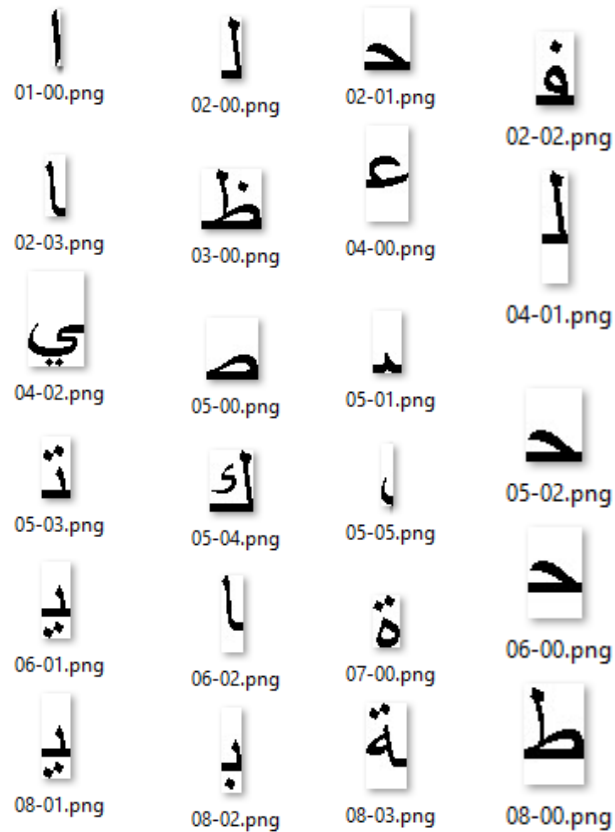


Gambar IV.7 Hasil segmentasi kalimat dua

3. Segmentasi Kalimat tiga

Kalimat : الحفاظ علي صحتك حياة طيبة

Hasil segmentasi:



Gambar IV.8 Hasil segmentasi kalimat tiga

Berdasarkan pengujian segmentasi yang telah dilakukan kepada 10 kalimat Arab didapatkan hasil sebagai berikut

Tabel IV.1 Hasil segmentasi 10 kalimat dengan font Arial Unicode Ms

No	Kalimat	Jumlah Huruf	Berhasil di Segmentasi	Tidak Berhasil Di Segmentasi	Persentase Keberhasilan
1	Kalimat 1	24	20	4	83%
2	Kalimat 2	17	17	0	100%
3	Kalimat 3	21	19	2	90%
4	Kalimat 4	17	15	2	88%

5	Kalimat 5	13	13	0	100%
6	Kalimat 6	21	17	4	80%
7	Kalimat 7	14	10	4	71%
8	Kalimat 8	14	14	0	100%
9	Kalimat 9	18	14	4	77%
10	Kalimat 10	21	19	2	90%
Rata-Rata					88%

Tabel IV.2 Hasil segmentasi 10 kalimat dengan font Tahoma

No	Kalimat	Jumlah Huruf	Berhasil di Segmentasi	Tidak Berhasil Di Segmentasi	Persentase Keberhasilan
1	Kalimat 1	24	22	2	91%
2	Kalimat 2	17	17	0	100%
3	Kalimat 3	21	20	1	95%
4	Kalimat 4	17	17	0	100%
5	Kalimat 5	13	11	2	84%
6	Kalimat 6	21	20	1	95%
7	Kalimat 7	14	13	1	92%
8	Kalimat 8	14	14	0	100%
9	Kalimat 9	18	18	0	100%
10	Kalimat 10	21	20	1	95%
Rata-Rata					

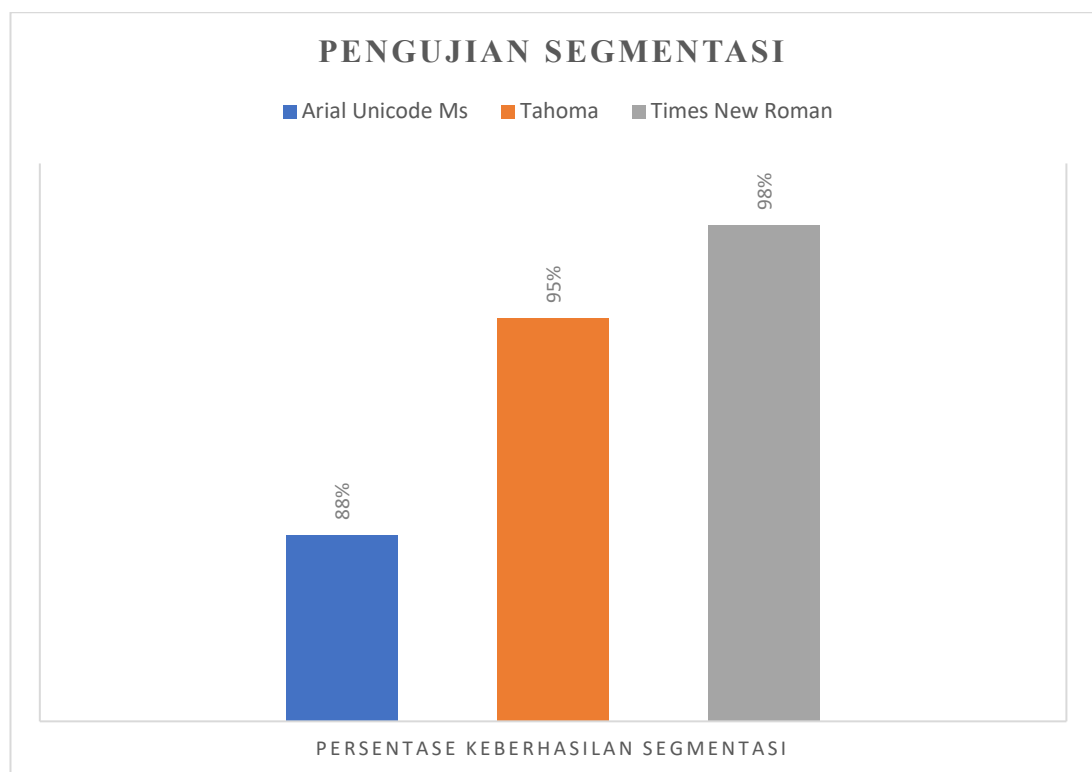
Tabel IV.3 Hasil segmentasi 10 kalimat dengan font Times New Roman

No	Kalimat	Jumlah Huruf	Berhasil di Segmentasi	Tidak Berhasil Di Segmentasi	Persentase Keberhasilan
1	Kalimat 1	24	24	0	100%
2	Kalimat 2	17	17	0	100%
3	Kalimat 3	21	21	0	100%
4	Kalimat 4	17	17	0	100%

5	Kalimat 5	13	13	0	100%
6	Kalimat 6	21	19	2	90%
7	Kalimat 7	14	14	0	100%
8	Kalimat 8	14	14	0	100%
9	Kalimat 9	18	18	0	100%
10	Kalimat 10	21	19	2	90%
Rata- Rata					98%

IV.1.1.2.1 Perbandingan Hasil Pengujian Segmentasi

Pengujian segmentasi terhadap 10 kalimat dalam tiga font telah dilakukan, berikut hasil perbandingan dari ketiga font tersebut.

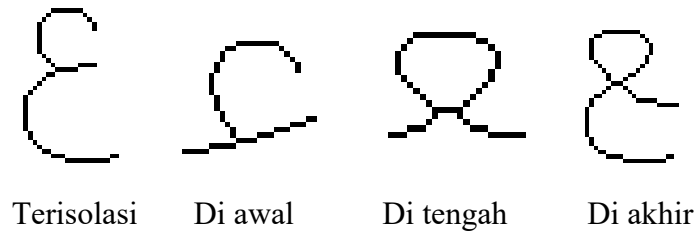


Gambar IV.9 Hasil pengujian segmentasi dengan Font Arial Unicode Ms, Tahoma dan Times New Roman

Berdasarkan hasil pengujian dengan 10 kalimat dengan tiga font, segmentasi dengan font *Times New Roman* mendapatkan hasil akurasi terbaik dengan mencapai 98%, selanjutnya diikuti dengan font *Tahoma* dengan akurasi 95%. Font *Arial Unicode Ms* mendapatkan hasil akurasi yang paling rendah yaitu 88%.

IV.1.1.3 Hasil Pengujian Penipisan

Penipisan dilakukan dengan algoritma Stentiford^[6]. Penipisan dilakukan berguna untuk mendapatkan chain code yang lebih bagus dan lebih mewakili bentuk suatu objek. Berikut Hasil dari Penipisan huruf Ain dengan font Times new roman pada posisi terisolasi, diawal, ditengah dan diakhir.



Gambar IV.10 Hasil Penipisan

Dalam penelitian ini penipisan berhasil dilakukan dengan baik pada seluruh data latih dan data uji.

IV.1.1.4 Ekstraksi Fitur

Huruf Arab yang terdiri dari 4 bentuk yaitu terisolasi, di awal, di tengah dan di akhir di Training menggunakan Java, data training telah mengalami proses binerisasi, penipisan dan ekstraksi fitur. Berikut ditampilkan hasil ekstraksi fitur data training untuk huruf “ba” dalam posisi terisolasi, di awal, ditengah dan di akhir.

Tabel IV.4 Data training huruf “Ba”

Huruf	Fitur			
	Posisi	Jumlah Titik	Posisi Titik	Chain code Ter-Normalisasi
Ba	Terisolasi	1	2	6, 6, 5, 4, 4, 4, 4, 2, 2
Ba	Di awal	1	2	6, 6, 6, 6, 6, 6, 8, 8, 8
Ba	Di tengah	1	2	6, 6, 6, 6, 6, 6, 8, 8, 8

Ba	Di Akhir	1	2	6, 5, 4, 4, 4, 4, 2, 2, 6, 4
----	----------	---	---	------------------------------

Ekstraksi Fitur telah berhasil dilakukan pada seluruh data latih dan data uji. Hasil Ekstraksi fitur seluruh data training dapat dilihat pada lampiran 1.

IV.1.1.5 Hasil Pengujian Sistem Pengenalan Huruf Arab dengan Neural Network

1. Hasil Pengujian Huruf Terisolasi

Pengujian dengan memasukkan 31 huruf Arab dalam tiga font yang berbeda ke dalam sistem. Sistem menghitung persentase masing – masing kemungkinan huruf yang mungkin dikenali, huruf yang memiliki persentase terbesar akan muncul sebagai huruf yang dikenali. Berdasarkan hasil pengenalan dengan *Neural Network*, di dapat hasil pengenalan huruf Arab terisolasi dalam tabel IV.2.

Tabel IV.5 Data hasil pengujian pengenalan huruf Arab terisolasi dengan metode Neural Network

No	Huruf	Font		
		Arial Unicode Ms	Tahoma	Times New Roman
1	Ain	√	√	√
2	alif	√	√	√
3	Ba	√	√	√
4	Dal	√	√	√
5	Dhad	√	√	√
6	Dzal	√	√	√
7	Dzo	√	√	√
8	Fa	√	√	√

9	Ghoin	√	√	√
10	Hamzah	√	√	√
11	Ha	√	√	√
12	Habesar	√	√	√
13	Jim	√	√	√
14	Kaf	√	√	√
15	Kha	√	√	√
16	Lam	√	√	√
17	Mim	√	√	√
18	Nun	√	√	√
19	Qaf	√	√	√
20	Ra	√	√	√
21	Sad	√	√	√
22	Sheen	√	√	√
23	Sin	√	√	√
24	Tamarbuto	√	√	√
25	Ta	√	√	√
26	Tho	√	√	√
27	Tsa	√	√	√
28	Waw	√	√	√
29	Ya	√	√	√

30	Za	√	√	√
31	Lamalif	√	√	√
Akurasi		100%	100%	100%

2. Hasil Pengujian Huruf Arab dalam Kalimat

Dalam pengujian huruf dalam kalimat dilakukan dengan memasukkan 10 kalimat Arab dengan tiga font berbeda kedalam sistem. Hasil pengujian terhadap pengenalan huruf Arab dalam kalimat dengan metode neural network terdapat pada tabel IV.3

Tabel IV.6 Data hasil pengujian pengenalan huruf Arab dalam kalimat dengan metode Neural Network

No	Kalimat	Akurasi Font		
		Arial Unicode Ms	Tahoma	Times New Roman
1	Kalimat 1	68.18%	65.38%	79.17%
2	Kalimat 2	70.59%	64.71%	82.35%
3	Kalimat 3	60.00%	72.73%	76.19%
4	Kalimat 4	75.00%	76.47%	76.47%
5	Kalimat 5	76.92%	53.85%	76.92%
6	Kalimat 6	63.64%	59.09%	73.91%
7	Kalimat 7	50.00%	73.33%	71.43%
8	Kalimat 8	64.29%	71.41%	71.43%
9	Kalimat 9	68.75%	66.67%	72.22%
10	Kalimat 10	63.63	59.09%	68.18%

Rata-Rata	66.10%	66.27%	74.83%
-----------	--------	--------	--------

IV.1.1.6 Hasil Pengujian Sistem Pengenalan Huruf Arab dengan Hidden Markov Model

1. Hasil Pengujian Huruf Terisolasi

Pengujian dengan memasukkan 31 huruf Arab dalam 3 font yang berbeda ke dalam sistem. Kemudian sistem melakukan penghitungan evaluasi dengan menghasilkan nilai peluang. Nilai peluang yang tertinggi menunjukkan tingkat kemiripan data huruf yang diujikan terhadap basis data yang merupakan hasil pelatihan data latih. Berdasarkan hasil pengenalan dengan *Hidden Markov Model*, didapatkan hasil pengenalan huruf Arab terisolasi dalam tabel IV.4.

Tabel IV.7 Data hasil pengujian pengenalan huruf Arab terisolasi dengan metode Hiden Markov Model

No	Huruf	Akurasi Font		
		Arial Unicode Ms	Tahoma	Times New Roman
1	Ain	√	X	√
2	alif	√	√	√
3	Ba	X	√	X
4	Dal	√	√	√
5	Dhad	√	X	√
6	Dzal	X	√	√
7	Dzo	√	√	√
8	Fa	√	√	√
9	Ghoir	X	X	√

10	Hamzah	√	√	√
11	Ha	√	X	X
12	Habesar	X	√	X
13	Jim	√	X	√
14	Kaf	√	X	X
15	Kha	√	√	X
16	Lam	√	√	√
17	Mim	√	√	√
18	Nun	X	X	√
19	Qaf	√	√	X
20	Ra	√	√	√
21	Sad	√	X	√
22	Sheen	√	√	√
23	Sin	√	√	√
24	Tamarbuto	√	√	√
25	Ta	X	X	X
26	Tho	√	√	√
27	Tsa	√	X	√
28	Waw	√	X	√
29	Ya	X	√	√
30	Za	√	√	√

31	Lamalif	X	X	$\sqrt{}$
Akurasi		74%	61%	77%

2. Hasil Pengujian Huruf dalam Kalimat

Pengujian dengan memasukkan 10 kalimat Arab dalam 3 font yang berbeda ke dalam sistem. Kemudian sistem melakukan penghitungan evaluasi dengan menghasilkan nilai peluang. Nilai peluang yang tertinggi menunjukkan tingkat kemiripan data huruf yang diujikan terhadap basis data yang merupakan hasil pelatihan data latih. Berdasarkan hasil pengenalan dengan *Hidden Markov Model*, di dapat hasil pengenalan huruf Arab terisolasi dalam tabel IV.5.

Tabel IV.8 Data hasil pengujian pengenalan huruf Arab dalam kalimat dengan metode Hiden Markov Model

No	Kalimat	Font		
		Arial Unicode Ms	Tahoma	Times New Roman
1	Kalimat 1	54%	53%	58%
2	Kalimat 2	52%	47%	7%
3	Kalimat 3	55%	40%	52%
4	Kalimat 4	43%	47%	52%
5	Kalimat 5	38%	46%	53%
6	Kalimat 6	54%	63%	47%
7	Kalimat 7	33%	40%	57%
8	Kalimat 8	78%	64%	57%
9	Kalimat 9	25%	33%	22%

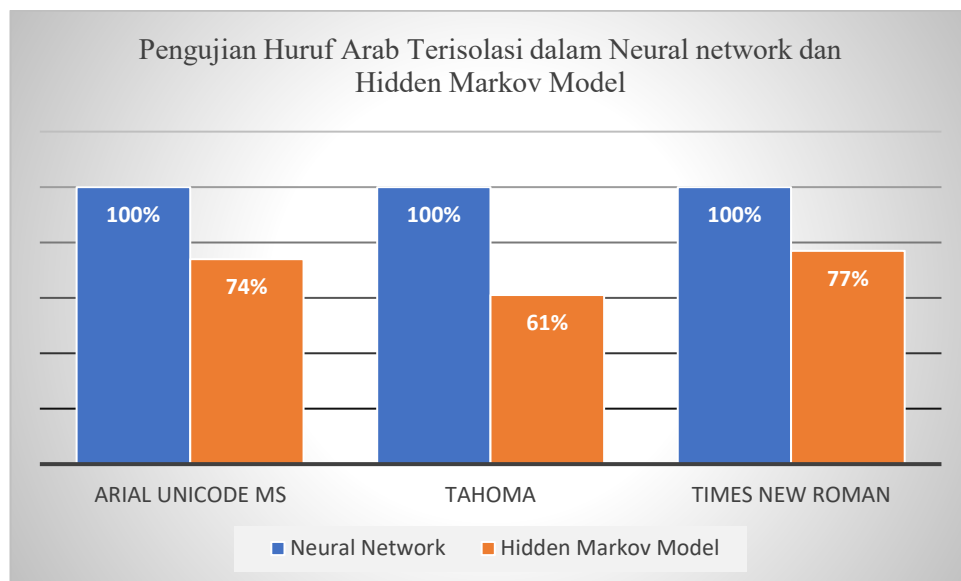
10	Kalimat 10	54%	63%	45%
Rata-Rata		49%	50%	51%

IV. 2 Pembahasan

Pengujian sistem pengenalan huruf Arab dalam bentuk terisolasi dan dalam kalimat menggunakan *Neural Network* dan *Hidden Markov Model* telah dilaksanakan, sehingga menghasilkan data akurasi. Berdasarkan seluruh data yang terhimpun akan ditampilkan grafik yang membandingkan kedua metode tersebut.

IV.2.1 Perbandingan Pengujian Huruf Arab Terisolasi

Berikut ini merupakan grafik dari hasil pengujian huruf Arab terisolasi



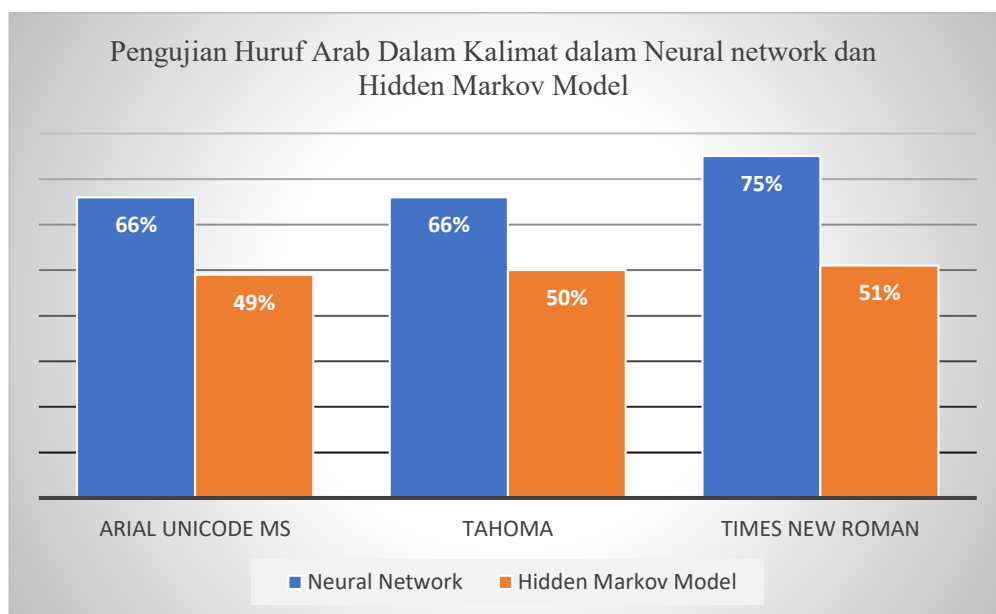
Gambar IV.11 Pengujian huruf Arab terisolasi dengan metode *Neural Network* dan *Hidden Markov Model*

Grafik di atas merepresentasikan hasil pengenalan huruf Arab terisolasi menggunakan *Neural Network* dan *Hidden Markov Model* berdasarkan pengambilan 31 huruf Arab terisolasi dalam 3 font yang berbeda. Maka hasil pengenalan menggunakan *Neural Network* menunjukkan hasil yang lebih baik dibandingkan *Hidden Markov Model* dalam ketiga font. Hasil pengenalan menggunakan *Neural Network* mencapai hasil

100% untuk font *Arial Unicode Ms*, *Tahoma* dan *Times New Roman* sedangkan hasil pengenalan menggunakan *Hidden Markov Model* menghasilkan akurasi 74% untuk font *Arial Unicode Ms*, 61% untuk font *Tahoma* dan 77% untuk font *Times New Roman*.

IV.2.2 Perbandingan Pengujian Huruf Arab dalam Kalimat

Berikut ini merupakan grafik dari hasil pengujian huruf Arab dalam kalimat



Gambar IV.12 Pengujian huruf Arab dalam kalimat dengan metode Neural Network dan Hidden Markov Model

Grafik di atas merepresentasikan hasil pengenalan huruf Arab dalam kalimat menggunakan *Neural Network* dan *Hidden Markov Model*, berdasarkan 10 data kalimat dengan 3 font berbeda yang diambil. Maka hasil pengenalan dengan *Neural Network* menunjukkan hasil yang lebih baik dalam ketiga font. Hasil pengenalan dengan *Neural Network* mencapai akurasi 66% untuk font *Arial Unicode Ms*, 66% untuk font *Tahoma* dan 75% untuk font *Times New Roman*. Sedangkan hasil pengenalan menggunakan *Hidden Markov Model* mencapai akurasi 49% untuk font *Arial Unicode Ms*, 50% untuk font *Tahoma* dan 51%. Hasil pengenalan keseluruhan menunjukkan bahwa font *Times New Roman* menghasilkan pengenalan yang lebih

baik dibandingkan dengan font *Arial Unicode Ms* dan *Thaoma*, baik dalam pengenalan dengan metode *Neural Network* dan *Hidden Markov Model*.

IV.2.3 Rekapitulasi Pembahasan dari Seluruh Hasil Pengujian

Seluruh hasil pengujian kemudian dihitung rata-rata hasil akurasinya, sehingga didapatkan nilai akurasi masing-masing metode berdasarkan pengujian dengan pengenalan huruf terisolasi dan huruf dalam kalimat dalam tabel berikut.

Tabel IV.9 Perbandingan hasil rekapitulasi akurasi

Metode	Pengujian	
	Huruf Terisolasi	Huruf dalam Kalimat
Neural Network	100%	69%
Hidden Markov Model	71%	50%

Dalam pengujian pengenalan huruf Arab terisolasi metode *Neural Network* lebih baik jika dibandingkan dengan metode *Hidden Markov Model*, metode *Neural Network* dapat mengenali huruf Arab dengan mencapai akurasi 100%, sedangkan pengenalan menggunakan *Hidden Markov Model* mencapai 71%. Kemudian berdasarkan pengujian pengenalan huruf Arab dalam kalimat, metode *Neural Network* masih lebih baik dibandingkan dengan metode *Hidden Markov Model*, yakni mencapai akurasi 69%, sedangkan metode *Hidden Markov Model* mencapai 50%.

Jika diperhatikan berdasarkan masing-masing font, font *Times New Roman* memiliki keunggulan tingkat pengenalan dibandingkan font *Arial Unicode Ms* dan font *Tahoma*, baik dalam pengenalan huruf Arab terisolasi maupun dalam pengenalan huruf Arab dalam kalimat. Pada pengenalan huruf Arab terisolasi dengan metode *Neural Network* akurasi pengenalan mencapai 100% untuk seluruh font, sedangkan pada metode *Hidden Markov Model* font *Tahoma* memiliki akurasi pengenalan paling rendah. Pada pengenalan huruf Arab dalam kalimat dengan metode *Neural Network* font *Arial Unicode Ms* dan font *Tahoma* memiliki akurasi yang sama, yaitu 66%, ini lebih rendah dibandingkan dengan font *Times New Roman* yang memiliki akurasi 75%, sedangkan pada *Hidden Markov Model* font *Times new Roman* memiliki akurasi paling tinggi, disusul oleh font *Tahoma* dan font *Arial Unicode Ms*.

Pengenalan huruf Arab dalam kalimat mengalami akurasi yang lebih rendah dibandingkan pengenalan huruf Arab terisolasi. Ini dikarenakan pengenalan huruf Arab dalam kalimat melewati proses segmentasi yang menyebabkan hasil binerisasi huruf data uji sedikit mengalami perbedaan dengan hasil binerisasi data latih, sehingga hasil chain code data latih dan data uji memiliki perbedaan.

Berdasarkan hasil dan analisa didapatkan metode *Neural Network* lebih baik untuk digunakan sebagai pengenalan huruf Arab dibandingkan metode *Hidden Markov Model*, baik dalam huruf terisolasi maupun huruf dalam kalimat.

Bab selanjutnya merupakan bab terakhir yang menjelaskan kesimpulan dari hasil penelitian. Selain itu terdapat saran yang dapat dilakukan oleh peneliti selanjutnya.

BAB V

KESIMPULAN DAN SARAN

V.1 Kesimpulan

Dalam penelitian ini telah dilakukan eksperimen untuk mengenali huruf Arab dalam bentuk terisolasi dan dalam kalimat. Perancangan sistem pengenalan huruf Arab ini telah dapat diujikan kepada tulisan Bahasa Arab. Untuk mendapatkan hasil yang paling baik, maka diperlukan perbandingan terhadap metode yang digunakan untuk mengenali huruf Arab. Metode ekstraksi fitur utama yang digunakan adalah *chain code* yang telah di normalisasi, diikuti dengan jumlah titik dan posisi titik. Kemudian sistem menbandingkan metode klasifikasi Neural Network dan Hidden Markov Model.

Proses latih mempengaruhi hasil pengenalan terhadap pengenalan huruf Arab terisolasi dan dalam kalimat. Untuk pengenalan huruf Arab terisolasi Metode *Neural Network* dapat mencapai akurasi sebesar 100%, sedangkan metode *Hidden Markov Model* dapat mencapai akurasi sebesar 71%. Hasil pengujian pengenalan huruf Arab dalam kalimat menggunakan *Neural Network* mencapai akurasi sebesar 69% sedangkan metode *Hidden Markov Model* mencapai akurasi sebesar 50%.

Setelah dilakukan perbandingan pengenalan huruf Arab pada tulisan cetak dalam posisi terisolasi maupun dalam kalimat dapat disimpulkan bahwa metode *Neural Network* lebih baik dibandingkan metode *Hidden Markov Model*.

V.2 Saran

Dalam penelitian ini akurasi pengenalan huruf Arab dalam kalimat masih lebih rendah dibandingkan dengan akurasi pengenalan huruf Arab terisolasi. Untuk penelitian selanjutnya dapat dikembangkan ekstraksi fitur tambahan untuk menambah ciri dari masing – masing huruf sehingga diharapkan akurasi pengenalan menjadi lebih baik.

Sistem pengenalan pengenalan huruf Arab telah banyak dikembangkan baik dalam tulisan cetak maupun tulisan tangan. Namun masih sedikit penelitian tentang pengenalan tulisan Arab dalam kalimat. Oleh karena itu penelitian ini dapat digunakan sebagai dasar implementasi dalam mengembangkan sistem tersebut dengan perbaikan yang ditemukan peneliti selanjutnya. Dengan pengembangan sistem tersebut, diharapkan dimasa depan akan mempermudah proses pembelajaran huruf Arab pada umumnya.

DAFTAR PUSTAKA

- [1] Ismail, B., Fahd, B., and Yassine, S. (2013): Arabic reading machine for visually impaired people using TTS and OCR, 4th International Conference on Intelligent Systems, Modelling and Simulation, 1.
- [2] Nimas, A. M., Victor, A., and Nashrul H. (2016): Comparative analysis of the accuracy of backpropagation and learning vector quantisation for pattern recognition of hijaiyah letters, 6th International Conference on Information and Communication Technology for The Muslim World, 4.
- [3] M. Albakor, K. Saeed, and F. Sukkar. (2009): Intelligent system for Arabic character recognition, World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), 1
- [4] Iping, S., and Albadr, N. (2013): Arabic character recognition system development, The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013), 1.
- [5] H. Izakian, S. A. Monadjemi, B. Tork, L., K. Zamanifar. (2008): Multi-font farsi/arabic isolated character recognition using chain codes, World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:2, No:7, 1, 3.
- [6] F. W. M. Stentiford., amd R. G. Mortimer. (1983): Some New Heuristics for Thinning Binary Handprinted Characters for OCR, IEEE Transaction On Systems, MAN, AND Cybernetics, VOL. SMC - 13, NO. 1, 3-4
- [7] Zidouri, A. (2010): On multiple typeface arabic script recognition, Research Journal of Applied Sciences Engineering and Technology, 3.
- [8] Devi, H., Hanif, F., Egi, M. I. H., and Carmadi M. (2016): Comparison of Indonesian speaker recognition using vector quantization and hidden markov model for unclear pronunciation problem, IEEE 6th International Conference on System Engineering and Technology (ICSET), 3.

- [9] T. Y. Zhang., and C. Y. Suen. (1984): A fast parallel algorithm for thinning digital patterns, Communications of the ACM Volume 27 Number 3, 1-3.
- [10] C.J. Hilditch. (1968): An application of graph theory on pattern recognition, In Machine Intell. (B. Meltzer and Michie Eds). New York Amer. Elsevier, 3.