

**PENERAPAN ALGORITMA
LEARNING VECTOR QUANTIZATION
WITH ADAPTIVE PROTOTYPE ADDITION AND REMOVAL
UNTUK PENGENALAN HURUF BAHASA ISYARAT
(STUDI KASUS: SLB ISLAM YASINDO MALANG)**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV
Politeknik Negeri Malang

Oleh:

YUSLIANA GADIS BUATA

NIM. 1341180150



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
AGUSTUS 2017**

**PENERAPAN ALGORITMA
LEARNING VECTOR QUANTIZATION
WITH ADAPTIVE PROTOTYPE ADDITION AND REMOVAL
UNTUK PENGENALAN HURUF BAHASA ISYARAT
(STUDI KASUS: SLB ISLAM YASINDO MALANG)**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV
Politeknik Negeri Malang

Oleh:

YUSLIANA GADIS BUATA

NIM. 1341180150



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
AGUSTUS 2017**

HALAMAN PENGESAHAN

PENERAPAN ALGORITMA
LEARNING VECTOR QUANTIZATION
WITH ADAPTIVE PROTOTYPE ADDITION AND REMOVAL
UNTUK PENGENALAN HURUF BAHASA ISYARAT
(STUDI KASUS: SLB ISLAM YASINDO MALANG)

Disusun oleh:

YUSLIANA GADIS BUATA NIM. 1341180150

Skripsi ini telah diuji pada tanggal 10 Agustus 2017

Disetujui oleh:

- | | | | |
|------------------|---|---|-------|
| 1. Penguji I | : | <u>Dr. Eng. Cahya Rahmad, ST., M.Kom</u>
NIP. 19720202 200501 1 002 | |
| 2. Penguji II | : | <u>Dr. Eng. Rosa Andrie Asmara, ST., MT</u>
NIP. 19801010 200501 1 001 | |
| 3. Pembimbing I | : | <u>Ariadi Retno Tri Hayati Ririd, S.Kom., M.Kom</u>
NIP. 19810810 200501 2 002 | |
| 4. Pembimbing II | : | <u>Yoppy Yunhasnawa, SST., M.Sc</u> | |

Mengetahui,

Ketua Jurusan
Teknologi Informasi

Ketua Program Studi
Teknik Informatika

Rudy Ariyanto, S.T., M.Cs.
NIP. 19711110 199903 1 002

Ir. Deddy Kusbianto PA, M.MKom.
NIP. 19621128 198811 1 001

PERNYATAAN

Dengan ini saya menyatakan bahwa Skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Malang, Agustus 2017

Yusliana Gadis Buata

ABSTRAK

Buata, Yusliana Gadis, “Penerapan Algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal* untuk Pengenalan Huruf Bahasa Isyarat (Studi Kasus: SLB Islam Yasindo Malang)”. **Pembimbing (1) Ariadi Retno Tri Hayati Ririd, S.Kom., M.Kom (2) Yoppy Yunhasnawa, S.ST., M.sc**

Skripsi, Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, 2017.

Para tuna rungu dan tuna wicara menggunakan komunikasi non verbal dalam berkomunikasi. Komunikasi ini menggunakan bahasa isyarat baik berupa gerakan isyarat tangan, isyarat tubuh dan ekspresi wajah. Agar dapat mengerti bahasa isyarat diperlukan suatu media komunikasi penerjemah bahasa isyarat tersebut menjadi bahasa verbal yang mudah dipahami. Dalam upaya memudahkan komunikasi orang tuna rungu dan tuna wicara dengan orang normal di SLB Islam Yasindo Malang dibuatlah sistem pengenalan huruf bahasa isyarat untuk membantu menerjemahkan citra tangan huruf bahasa isyarat menjadi tulisan alfabet.

Tahapan yang dilakukan dalam implementasi sistem pengenalan huruf bahasa isyarat ini adalah dengan membedakan objek dan *background* dari citra tangan untuk menentukan citra yang mencerminkan huruf Sistem Isyarat Bahasa Indonesia (SIBI). Citra tangan dimuat dari direktori kemudian dilakukan proses *preprocessing*, pelatihan data dan klasifikasi tangan. Metode klasifikasi yang digunakan dalam pengenalan huruf bahasa isyarat tangan ini menggunakan metode *Learning Vector Quantization with Adaptive Prototype Addition and Removal*.

Sistem ini dapat mengenali 24 huruf bahasa isyarat yang ditargetkan. Berdasarkan hasil dari perancangan dan pengujian sistem pada penelitian ini, persentase akurasi baca huruf terbaik adalah sebesar 88,75%. Hasil dari pengenalan ini dipengaruhi oleh proses validasi, teknik pengambilan citra dan faktor pencahayaan.

Kata Kunci: *Huruf Bahasa Isyarat SIBI, Preprocessing, Pelatihan Data, Klasifikasi Tangan, Learning Vector Quantization, Adaptive Prototype Addition and Removal.*

ABSTRACT

***Buata, Yusliana Gadis, “Implementation of Learning Vector Quantization Algorithm with Adaptive Prototype Addition and Removal on Sign Language Letter Recognition (Case Study: Islam Yasindo Extraordinary School Malang)”.
Advisors: (1) Ariadi Retno Tri Hayati Ririd, S.Kom., M.Kom (2) Yoppy Yunhasnawa, S.ST., M.sc***

Thesis, Informatics Engineering Study Program, Department of Information Technology, State Polytechnic of Malang, 2017.

The deaf and speech impaired use non-verbal communication for communicating. This communication uses sign language either in the form of gestures, body cues and facial expression. In order to understand the sign language, it is required a media communication such sign language translator into an easily understood verbal language. In order to make communication between the deaf and speech impaired with normal persons easily at Islam Yasindo Extraordinary School Malang, a sign language letter recognition system was created to translate image of hand gesture into alphabetical form.

The steps of this system implementation is by differentiate foreground and background of the image to determine which image that reflects the sign language letters based on Indonesian Sign Language System (ISLS). The hand image loaded from the directory then the system do preprocessing, data training and hand classification. The classification method used in the system is Learning Vector Quantization Algorithm with Adaptive Prototype Addition and Removal.

This system can recognize 24 sign language letters targeted. Based on the results of the system design and testing, the best success rate of letters recognition is 88,75%. The recognition result is based on validation process, image acquisition techniques and exposure factor.

Keywords: *Indonesian Sign Language, Preprocessing, Data Training, Hand Classification, Learning Vector Quantization, Adaptive Prototype Addition and Removal.*

KATA PENGANTAR

Puji Syukur kami panjatkan kehadiran Allah SWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan Skripsi dengan judul “PENERAPAN ALGORITMA LEARNING VECTOR QUANTIZATION WITH ADAPTIVE PROTOTYPE ADDITION AND REMOVAL UNTUK PENGENALAN HURUF BAHASA ISYARAT (STUDI KASUS: SLB ISLAM YASINDO MALANG)”. Skripsi ini penulis susun sebagai persyaratan untuk menyelesaikan studi program Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Penulis menyadari tanpa adanya dukungan dan kerja sama dari berbagai pihak, kegiatan Skripsi ini tidak akan dapat berjalan baik. Untuk itu, penulis ingin menyampaikan rasa terima kasih kepada:

1. Allah SWT yang selalu memberi rahmat dan hidayah-Nya sehingga penulis diberi kemampuan dan kemudahan selama penyelesaian Skripsi ini.
2. Seluruh keluarga, orang tua tercinta yang selalu mendoakan demi kelancaran Skripsi.
3. Bapak Rudy Ariyanto, ST., M.Cs., selaku Ketua Jurusan Teknologi Informasi.
4. Bapak Ir. Deddy Kusbianto PA, M.Mkom., selaku Ketua Program Studi Teknik Informatika
5. Ibu Ariadi Retno Tri Hayati Ririd, S.Kom., M.Kom., selaku Dosen Pembimbing Skripsi I.
6. Bapak Yoppy Yunhasnawa, SST., selaku Dosen Pembimbing Skripsi II.
7. Rian Wahyu Sahadah yang selalu menyemangati dan memberikan motivasi selama pembuatan Skripsi.
8. Ayundha Wulan Kurniawati, Dhike Almira Ramadiyah, Annisa Latifahtul Qalby, Meilinda Ika Zuliyati, dan Achmad Prayogi selaku teman-teman dan Yul'amalya Gladys Buata selaku kakak yang telah membantu dalam pengambilan data citra tangan.

9. Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Skripsi dari awal hingga akhir yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan Skripsi ini, masih banyak terdapat kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika penulisan maupun penggunaan bahasa. Penulis mengharapkan saran dan kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan Skripsi ini. Semoga Skripsi ini berguna bagi pembaca secara umum dan penulis secara khusus. Akhir kata, penulis ucapkan banyak terima kasih.

Malang, Agustus 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN PENGESAHAN.....	ii
PERNYATAAN.....	iii
ABSTRAK.....	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xi
DAFTAR LAMPIRAN.....	xii
BAB I. PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	2
1.5 Sistematika Penulisan Laporan.....	3
BAB II. LANDASAN TEORI.....	5
2.1 Bahasa Isyarat.....	5
2.2 Pengolahan Citra Digital	7
2.2.1 Citra Digital.....	7
2.3 Operasi Penskalaan (<i>Scalling</i>).....	7
2.4 <i>Preprocessing</i>	8
2.4.1 <i>Skin Detection</i>	8
2.4.2 Pengambangan (<i>Thresholding</i>)	9
2.4.3 Dilasi	9
2.4.4 Erosi	10
2.4.5 <i>Opening</i>	11
2.5 Ekstraksi Fitur	12
2.6 Jaringan Saraf Tiruan.....	12
2.7 <i>Learning Vector Quantization</i>	13
2.7.1 Arsitektur <i>LVQ</i>	13
2.7.2 Algoritma <i>LVQ</i>	14
2.7.3 <i>Adaptive LVQ</i>	15
2.7.4 Contoh Perhitungan.....	16
BAB III. METODOLOGI PENELITIAN	19
3.1 Identifikasi Masalah	19
3.2 Studi Literatur.....	20
3.3 Analisa Kebutuhan	20
3.4 Perancangan Sistem.....	21
3.5 Implementasi Sistem	22
3.6 Pengujian dan Analisis	22
3.7 Kesimpulan dan Saran.....	22
3.8 Penulisan Laporan	22
BAB IV. ANALISIS DAN PERANCANGAN	23
4.1 Analisa Perangkat Lunak.....	23

4.1.1	Analisa Kebutuhan Data	23
4.1.2	Analisa Kebutuhan Perangkat Lunak.....	24
4.1.3	Teknik Pengambilan Data	27
4.2	Perancangan Diagram Alur Sistem	27
4.2.1	<i>Load Image</i>	28
4.2.2	<i>Skin Detection</i>	28
4.2.3	<i>Thresholding</i>	30
4.2.4	<i>Denoising</i>	30
4.2.5	<i>Extraction</i>	31
4.2.6	<i>Adaptive LVQ Classification</i>	32
4.2.7	<i>Output Alphabet</i>	34
4.3	Perancangan Antarmuka Sistem.....	34
4.4	Perancangan Pengujian.....	38
BAB V.	IMPLEMENTASI.....	40
5.1	Spesifikasi Sistem.....	40
5.1.1	Spesifikasi <i>Hardware</i>	40
5.1.2	Spesifikasi <i>Software</i>	40
5.2	Implementasi Kode Program	40
5.2.1	Kode Program <i>Load Image</i>	41
5.2.2	Kode Program <i>Skin Detection</i>	41
5.2.3	Kode Program <i>Thresholding</i>	42
5.2.4	Kode Program <i>Denoising</i>	43
5.2.5	Kode Program <i>Extraction</i>	43
5.2.6	Kode Program <i>Adaptive LVQ Classification</i>	44
5.3	Implementasi <i>User Interface</i>	48
5.3.1	<i>UI Home Page</i>	48
5.3.2	<i>UI Training Page</i>	48
5.3.3	<i>UI Testing Page</i>	49
5.3.4	<i>UI Help Page</i>	50
5.3.5	<i>UI About Page</i>	51
BAB VI.	PENGUJIAN DAN PEMBAHASAN	52
6.1	Pengujian	52
6.1.1	Pengujian pada Data <i>Training</i>	52
6.1.2	Pengujian pada Data <i>Testing</i>	52
6.2	Analisis	53
BAB VII.	PENUTUP.....	52
7.1	Kesimpulan.....	55
7.2	Saran	56
DAFTAR PUSTAKA	57
LAMPIRAN	59

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Huruf pada Sistem Isyarat Bahasa Indonesia	6
Gambar 2.2 Susunan Piksel Citra Biner	9
Gambar 2.3 Proses Dilasi Menggunakan SE 3 X 3	10
Gambar 2.4 Proses Erosi.....	11
Gambar 2.5 Operasi <i>Opening</i> (a) Citra Asli (b) Citra Hasil	11
Gambar 2.6 Arsitektur Jaringan <i>Learning Vector Quantization</i>	14
Gambar 2.7 <i>Flowchart</i> Algoritma <i>Adaptive LVQ</i>	16
Gambar 3.1 Diagram Alir Metode Penelitian	19
Gambar 3.2 Arsitektur Sistem Pengenalan Huruf Bahasa Isyarat.....	21
Gambar 4.1 Data Masukan Bentuk Pola Tangan Huruf Bahasa Isyarat.....	23
Gambar 4.2 Diagram <i>Use Case</i> Sistem	25
Gambar 4.3 Diagram Kelas Sistem	26
Gambar 4.4 <i>Flowchart</i> Sistem Pengenalan Huruf Bahasa Isyarat	28
Gambar 4.5 <i>Flowchart Skin Detection</i>	29
Gambar 4.6 <i>Flowchart Set of Rules (RGB Color Space)</i>	29
Gambar 4.7 <i>Flowchart Thresholding</i>	30
Gambar 4.8 <i>Flowchart</i> Proses Erosi dan Dilasi.....	31
Gambar 4.9 <i>Flowchart Extraction</i> dengan Fitur Biner.....	32
Gambar 4.10 <i>Flowchart Adaptive LVQ Classification</i>	33
Gambar 4.11 <i>Flowchart</i> Proses <i>Training Adaptive LVQ</i>	33
Gambar 4.12 <i>Flowchart</i> Proses <i>Testing Adaptive LVQ</i>	34
Gambar 4.13 Rancangan <i>Home Page</i>	35
Gambar 4.14 Rancangan <i>Training Page</i>	35
Gambar 4.15 Rancangan <i>Testing Page</i>	36
Gambar 4.16 Rancangan <i>Help Page</i>	37
Gambar 4.17 Rancangan <i>About Page</i>	38
Gambar 5.1 Implementasi <i>Home Page</i>	48
Gambar 5.2 Implementasi <i>Training Page</i>	49
Gambar 5.3 Implementasi <i>Testing Page</i>	50
Gambar 5.4 Implementasi <i>Help Page</i>	50
Gambar 5.5 Implementasi <i>About Page</i>	51
Gambar 6.1 Grafik Hasil Akurasi Data <i>Testing</i>	53

DAFTAR TABEL

	Halaman
Tabel 2.1 <i>Input</i> Vektor Data <i>Training</i>	17
Tabel 4.1 Daftar Kebutuhan Sistem.....	24
Tabel 4.2 Perancangan Pengujian Data <i>Testing</i>	39
Tabel 5.1 Spesifikasi <i>Hardware</i>	40
Tabel 5.2 Spesifikasi <i>Software</i>	40
Tabel 6.1 Hasil Pengujian terhadap Pengaruh Jumlah <i>Dataset Testing</i>	53

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Hasil <i>Training</i> Sistem	59
Lampiran 2. Hasil <i>Testing</i> Setiap Data	63
Lampiran 3. Hasil <i>Testing</i> Keseluruhan	74
Lampiran 4. Uji Coba Sistem di SLB Islam Yasindo Malang	75
Lampiran 5. Lembar Bimbingan Skripsi Pembimbing I	80
Lampiran 6. Lembar Bimbingan Skripsi Pembimbing II	81
Lampiran 7. Lembar Revisi Skripsi Penguji I	82
Lampiran 8. Lembar Revisi Skripsi Penguji II	83
Lampiran 9. Profil Penulis	84

BAB I. PENDAHULUAN

1.1 Latar Belakang

Menurut hasil Survey Sosial Ekonomi Nasional (Susenas) yang dilaksanakan Biro Pusat Statistik (BPS) tahun 2012, jumlah penyandang disabilitas di Indonesia sebanyak 6.008.661 orang. Dari jumlah tersebut sekitar 1.780.200 orang adalah penyandang disabilitas netra, 472.855 orang penyandang disabilitas rungu wicara, 402.817 orang penyandang disabilitas grahita/intelektual, 616.387 orang penyandang disabilitas tubuh, 170.120 orang penyandang disabilitas yang sulit mengurus diri sendiri, dan sekitar 2.401.592 orang mengalami disabilitas ganda [1]. Sedangkan menurut WHO, diperkirakan anak Indonesia dengan kebutuhan khusus mencapai sekitar 7-10% dari total jumlah anak di Indonesia [2].

Anak berkebutuhan khusus yang lain jika dibandingkan dengan anak tuna rungu dan tuna wicara, apabila dilihat dari segi fisik tidaklah berbeda. Anak tuna rungu dan tuna wicara memiliki keterbatasan dalam berkomunikasi secara verbal [2]. Sehingga para tuna rungu dan tuna wicara umumnya menggunakan komunikasi secara visual seperti bahasa isyarat sebagai alat komunikasi utamanya. Penggunaan bahasa isyarat ini menggunakan orientasi bentuk dan gerakan tangan, lengan, tubuh serta ekspresi wajah untuk mengungkapkan pikiran mereka [3].

Tetapi cara komunikasi tersebut sering menyulitkan/ membatasi komunikasi dengan orang normal. Kebanyakan masyarakat umum masih belum memahami cara menerjemahkan bahasa isyarat, dan juga masih terbatasnya *volunteer* yang dapat membantu penderita tuna rungu dan tuna wicara dalam kehidupan sosial. Sehingga adanya keterbatasan komunikasi tersebut menjadi permasalahan sosial antar keduanya. Untuk dapat mengatasi permasalahan sosial antara penderita tuna rungu dan tuna wicara dengan orang normal, maka diperlukan penerjemah bahasa isyarat menjadi bahasa umum atau mudah dipahami orang normal. Dalam rangka berusaha mewujudkan upaya tersebut, maka dibutuhkan suatu sistem yang dapat menerjemahkan huruf bahasa isyarat tersebut menjadi tulisan alfabet [3].

Pembuatan sistem penerjemah bahasa isyarat salah satunya dapat menggunakan pola tangan pengguna yang dimuat dari direktori. Setelah citra

tangan telah dimuat dan ditampilkan pada sistem, kemudian dilakukan klasifikasi pola dari bentuk citra tangan tersebut [4].

Salah satu metode yang dapat digunakan untuk mengklasifikasi pola bentuk tangan pada citra adalah *Learning Vector Quantization* (LVQ). LVQ adalah algoritma klasifikasi yang memanfaatkan vektor yang terhubung dan bekerja secara kompetitif namun terbimbing untuk menyelesaikan suatu masalah. LVQ banyak digunakan dalam pengenalan pola dan klasifikasi data. Metode ini cukup sederhana namun sangat efektif [5]. Pada penelitian terdahulu oleh Mihajlo Grbovic dan Slobodan Vucetic terdapat pengembangan pada metode LVQ dengan *Adaptive Prototype Addition and Removal* dimana menghasilkan persentase lebih baik dibandingkan dengan metode lainnya [6].

Berdasarkan latar belakang yang telah diuraikan, maka pada penelitian kali ini penulis mengambil judul “Penerapan Algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal* untuk Pengenalan Huruf Bahasa Isyarat (Studi Kasus: SLB Islam Yasindo Malang)”.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka diperoleh rumusan masalah yaitu bagaimana cara mengenalkan huruf bahasa isyarat kepada warga SLB Islam Yasindo Malang yang belum mengenal huruf bahasa isyarat sehingga memudahkan komunikasi antara tuna rungu dan tuna wicara dengan orang normal.

1.3 Tujuan

Tujuan dari penelitian ini adalah menerapkan algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal* untuk sistem pengenalan huruf bahasa isyarat pada SLB Islam Yasindo Malang dalam rangka memudahkan komunikasi antara tuna rungu dan tuna wicara dengan orang normal.

1.4 Batasan Masalah

Dari latar belakang, rumusan masalah dan tujuan yang telah diuraikan, maka batasan masalah yang digunakan sebagai berikut:

- a. Sistem yang dibuat hanya menentukan huruf bahasa isyarat statik saja yaitu citra 2 dimensi dimana tangan dan jari sebagai komponen utama.

- b. Sistem hanya menerjemahkan huruf bahasa isyarat dari A sampai Z, kecuali J dan Z.
- c. Bahasa isyarat yang digunakan sebagai acuan adalah Sistem Isyarat Bahasa Indonesia (SIBI).
- d. Penderita tuna rungu yang menggunakan sistem ini adalah bukan penderita sejak lahir yang belum memahami tata Bahasa Indonesia.
- e. Sistem ini menggunakan bahasa pemrograman Visual Basic.NET.
- f. Proses normalisasi data secara manual.

1.5 Sistematika Penulisan Laporan

Berikut ini merupakan sistematika penulisan pada Sistem Pengenalan Huruf Bahasa Isyarat:

i. **Bab I: Pendahuluan**

Pada bab ini membahas tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan yang digunakan.

ii. **Bab II: Landasan Teori**

Pada bab ini berisi penjelasan tentang landasan teori yang meliputi studi penelitian terdahulu dan studi pustaka yang selanjutnya digunakan dalam bagian pembahasan dan sebagai dasar dalam pembuatan sistem.

iii. **Bab III: Metodologi Penelitian**

Pada bab ini berisi penjelasan tentang bagaimana metode penelitian yang dilakukan, penjelasan dari metode penelitian dan penjelasan kerangka konseptual penelitian.

iv. **Bab IV: Analisis dan Perancangan**

Pada bab ini berisi tentang analisis kebutuhan yang diperlukan untuk pembuatan skripsi ini, serta perancangan mengenai Sistem Pengenalan Huruf Bahasa Isyarat.

v. **Bab V: Implementasi**

Pada bab ini membahas tentang implementasi dari analisis dan perancangan sistem meliputi implementasi sistem dan implementasi metode *Learning Vector Quantization with Adaptive Prototype Addition and Removal*.

vi. Bab VI: Pengujian dan Pembahasan

Pada bab ini berisi hasil pengujian akurasi implementasi Sistem Pengenalan Huruf Bahasa Isyarat.

vii. Bab VII: Penutup

Pada bab ini berisi kesimpulan dari seluruh proses analisis hingga uji coba dan saran yang dapat membantu dalam pengembangan sistem ini untuk masa yang akan datang.

BAB II. LANDASAN TEORI

2.1 Bahasa Isyarat

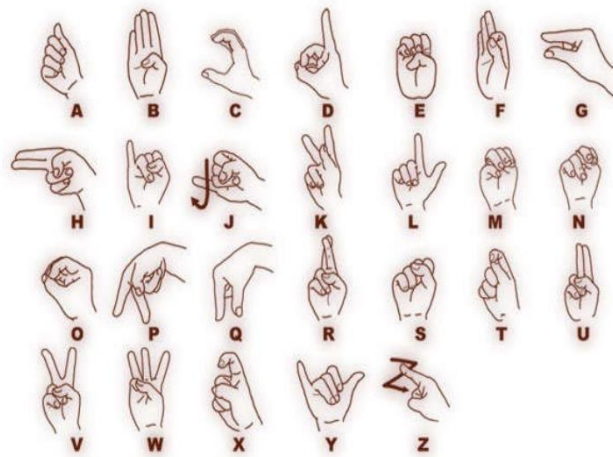
Bahasa isyarat merupakan bahasa tubuh dimana komunikasi menggunakan alat gerak tubuh untuk membentuk simbol tertentu yang membentuk makna tertentu. Penggunaan bahasa tubuh tersebut diaplikasikan ke dalam bentuk bahasa isyarat sebagai bentuk komunikasi kaum tuna rungu dan tuna wicara. Kaum tuna rungu dan tuna wicara tidak mampu memanfaatkan alat bicara mereka sehingga mereka akan menggunakan alat gerak tubuh yang lain untuk mengekspresikan maksud mereka, dan penerima akan menerima simbol-simbol tubuh tersebut sebagai sebuah pesan.

Bahasa isyarat berkembang dan memiliki karakteristik yang berbeda pada tiap negara. Di Indonesia, ada dua bahasa isyarat yang digunakan yaitu SIBI atau Sistem Isyarat Bahasa Indonesia dan Bahasa Isyarat Indonesia atau BISINDO. Sistem Isyarat Bahasa Indonesia dikembangkan menurut kaidah-kaidah pengembangan sistem yang isyarat yang merupakan salah satu kriteria untuk membuat sistem isyarat yang tepat guna bagi pelajar tuna rungu dan tuna wicara, yaitu:

- a. Sistem isyarat harus secara akurat dan konsisten mewakili tata bahasa/sintaksis Bahasa Indonesia yang paling banyak digunakan oleh masyarakat Indonesia.
- b. Tiap isyarat dalam sistem yang disusun harus mewakili satu kata dasar yang berdiri sendiri atau tanpa imbuhan. Tanpa menutup kemungkinan adanya beberapa pengecualian bagi dikembangkannya isyarat yang mewakili satu makna.
- c. Sistem isyarat yang disusun harus mencerminkan situasi sosial, budaya, dan ekologi Bangsa Indonesia.
- d. Sistem isyarat harus disesuaikan dengan perkembangan kemampuan dan kejiwaan siswa.
- e. Sistem isyarat harus disesuaikan dengan perkembangan bahasa siswa, termasuk metodologi pengajaran.

- f. Sistem isyarat harus memperhatikan isyarat yang sudah ada dan banyak dipergunakan oleh kaum tuna rungu.
- g. Sistem isyarat harus mudah dipelajari dan digunakan oleh siswa, guru, orang tua siswa, dan masyarakat.
- h. Isyarat yang dirancang harus memiliki kelayakan dalam wujud dan maknanya. Artinya wujud isyarat harus secara visual memiliki unsur pembeda makna yang jelas, tetapi sederhana dan indah/menunjukkan sifat yang luwes (memiliki kemungkinan untuk dikembangkan), jelas dan mantap (tidak berubah-ubah artinya) [7].

Berikut adalah contoh dari 26 huruf alfabet pada SIBI yang ditunjukkan pada gambar 2.1.



Gambar 2.1 Huruf pada Sistem Isyarat Bahasa Indonesia

Sumber: [8]

Sedangkan BISINDO merupakan bahasa isyarat yang diperjuangkan oleh Gerakan Kesejahteraan Tunarungu Indonesia (GERKATIN). BISINDO merupakan bahasa isyarat alami budaya asli Indonesia yang dengan mudah dapat digunakan dalam pergaulan isyarat kaum tuna rungu sehari-hari [9]. Perbedaan mendasar antara SIBI dan BISINDO adalah SIBI menggunakan abjad sebagai panduan bahasa isyarat tangan satu, sementara BISINDO menggunakan gerakan tangan (dua tangan) sebagai upaya komunikasi antar pengguna bahasa isyarat [10].

2.2 Pengolahan Citra Digital

Pengolahan citra digital merupakan sebuah teknologi yang menerapkan sejumlah algoritma komputer yang digunakan untuk memproses suatu citra digital. Keluaran yang dihasilkan dari proses ini dapat berupa citra ataupun sifat/karakteristik dari citra aslinya. Pengolahan citra digital pada umumnya telah digunakan dalam dunia robotika/sistem cerdas, biomedis, penginderaan jarak jauh, fotografi, forensik dan lain sebagainya. Tujuan dari pengolahan citra digital adalah untuk memungkinkan manusia memperoleh citra berkualitas tinggi atau karakteristik dari suatu citra agar mudah diinterpretasi [11]. Pengolahan citra digital mempelajari tentang hal-hal seperti perbaikan kualitas citra (peningkatan kontras, transformasi warna, restorasi citra), transformasi (rotasi, translasi, skala, transformasi geometrik), ekstraksi ciri (*feature extraction*) untuk tujuan analisis, pengambilan informasi atau deskripsi/pengenalan objek pada citra, kompresi/reduksi untuk/tujuan penyimpanan, transmisi data dan waktu proses data [12].

2.2.1 Citra Digital

Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan *picture elements*, *image elements*, *pels*, atau *pixels*. Istilah terakhir (*pixels*) paling sering digunakan pada citra digital [13].

2.3 Operasi Penskalaan (*Scaling*)

Penskalaan adalah sebuah operasi geometri yang memberikan efek memperbesar atau memperkecil ukuran citra input (*resizing*) sesuai dengan variabel penskalaan citranya. Ukuran baru hasil penskalaan didapat melalui perkalian antara ukuran citra *input* dengan variabel penskalaan.

Proses penskalaan dapat dilakukan dengan rumus:

$$P_0 = S_p \times P_i \quad (2.1)$$

$$L_0 = S_l \times L_i \quad (2.2)$$

Dimana (P_i, L_i) adalah ukuran citra *input*, (P_0, L_0) adalah ukuran citra *output*, dan (S_p, S_l) adalah variabel penskalaan yang diinginkan. Jika variabel penskalaan bernilai lebih besar dari 1 maka hasil penskalaannya akan memperbesar ukuran citra, sebaliknya apabila variabel penskalaannya lebih kecil dari 1 maka hasilnya akan memperkecil ukuran citra [13].

2.4 *Preprocessing*

Preprocessing adalah proses pengolahan citra yang dilakukan untuk membuat citra agar lebih berkualitas atau melakukan modifikasi citra untuk menonjolkan aspek/informasi yang terkandung di dalam citra, yang dibutuhkan pada tahap pengolahan citra berikutnya. Tujuan dari *preprocessing* ini adalah untuk memberikan kemungkinan keberhasilan yang lebih besar pada pengolahan selanjutnya atau memaksimalkan pengambilan informasi citra yang digunakan untuk ekstraksi ciri [14].

2.4.1 *Skin Detection*

Skin detection adalah proses mencari dan mengenali piksel warna kulit pada citra. Proses ini biasanya digunakan sebagai langkah *preprocessing* untuk menemukan daerah yang berpotensi memiliki wajah manusia dan anggota badan pada citra. Beberapa *computer vision* telah mengembangkan pendekatan-pendekatan untuk mendeteksi warna kulit pada citra [15]. Salah satu teknik yang digunakan dalam penelitian ini adalah dengan teknik *thresholding*, yaitu dengan menggunakan model aturan-aturan (*rules*) yang di dalamnya terdapat *range* warna atau nilai *threshold* untuk mendeteksi warna kulit. Model aturan yang dibangun menggunakan ruang warna RGB.

Pada ruang warna RGB, menggunakan model aturan yang dikembangkan oleh Peer, dkk. [16]. Untuk warna kulit dengan pencahayaan siang hari didefinisikan sebagai berikut.

$$(R > 95) \text{ AND } (G > 40) \text{ AND } (B > 20) \text{ AND } (\max\{R, G, B\} - \min\{R, G, B\} > 15) \text{ AND } (|R - G| > 15) \text{ AND } (R > G) \text{ AND } (R > B) \quad (2.3)$$

Sedangkan untuk warna kulit dengan cahaya lampu atau pencahayaan minimum menggunakan aturan sebagai berikut.

$$(R > 220) \text{ AND } (G > 210) \text{ AND } (B > 170) \text{ AND } (|R - G| \leq 15) \text{ AND } (R > B) \text{ AND } (G > B) \quad (2.4)$$

Kemudian kedua aturan tersebut digabungkan dengan operator logika OR. Setelah itu, setiap piksel yang memenuhi kedua aturan tersebut akan diklasifikasikan sebagai piksel warna kulit. Jika nilai piksel mengandung warna kulit maka nilai piksel tersebut akan tetap atau dibiarkan, dan jika tidak termasuk warna kulit maka warna nilai piksel akan dihitamkan sebagai *background*.

2.4.2 Pengambangan (*Thresholding*)

Proses pengambangan merupakan proses yang dilakukan untuk mengubah citra digital menjadi citra biner atau biasa juga disebut binerisasi. Citra biner adalah citra yang hanya memiliki dua kemungkinan nilai *pixel* yaitu hitam dan putih, atau juga biasa disebut citra B&W (*Black and White*) [13]. Proses pengambangan dilakukan agar objek yang diinginkan dalam citra dapat terpisah dengan latar belakangnya berdasarkan nilai ambang yang ditentukan. Pada citra biner biasanya objek ditandai dengan warna hitam bernilai 1 dan latar belakangnya dengan warna putih bernilai 0 [14]. Contoh dari susunan piksel citra biner ditunjukkan pada gambar 2.2 berikut.

					=	1	1	1	1	1
					=	0	0	1	0	0
					=	0	0	1	0	0
					=	0	0	1	0	0
					=	0	0	1	0	0

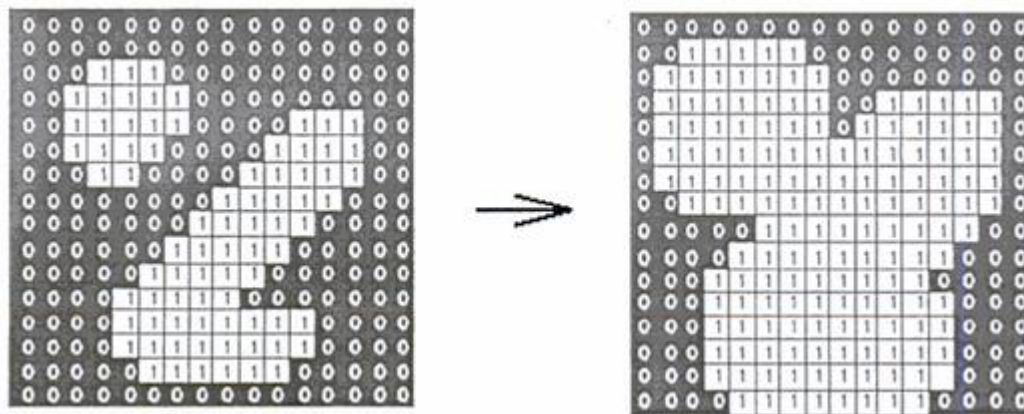
Gambar 2.2 Susunan Piksel Citra Biner

Sumber: [12]

2.4.3 Dilasi

Proses dilasi dilakukan dengan membandingkan setiap piksel citra *input* dengan nilai pusat SE (*Structuring Elements*) dengan cara melapiskan SE dengan citra sehingga pusat SE tepat dengan posisi piksel citra yang diproses. Jika paling

sedikit ada 1 piksel pada SE sama dengan nilai piksel objek (*foreground*) citra maka piksel *input* di-set nilainya dengan nilai piksel *foreground* dan bila semua piksel yang berhubungan adalah *background* maka *input* piksel diberi nilai piksel *background*. Proses serupa dilanjutkan dengan menggerakkan (translasi) SE piksel demi piksel pada citra *input*. Gambar 2.3 berikut ini mengilustrasikan suatu citra sebelum dan sesudah proses dilasi dengan menggunakan SE berukuran 3 x 3 dengan setiap elemen SE bernilai 1.



Gambar 2.3 Proses Dilasi Menggunakan SE 3 X 3

Sumber: [13]

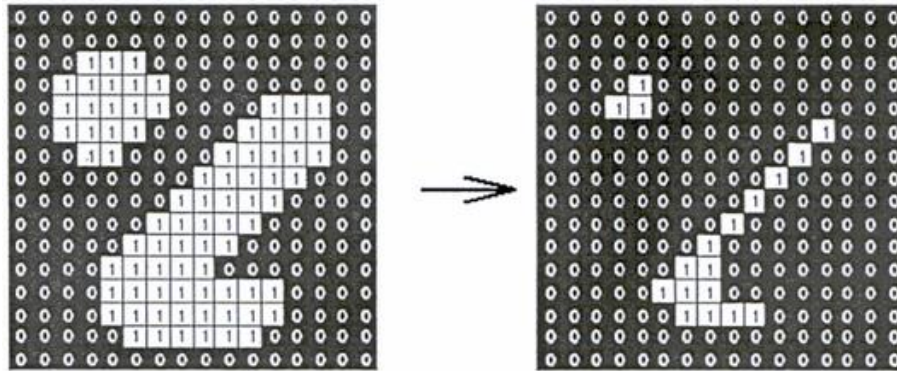
Semakin besar ukuran SE maka semakin besar perubahan yang terjadi. SE berukuran kecil juga dapat memberikan hasil yang sama dengan SE berukuran besar dengan cara melakukan dilasi berulang kali. Efek dilasi terhadap citra biner adalah memperbesar batas dari objek yang ada sehingga objek terlihat semakin besar dan lubang-lubang yang terdapat di tengah objek akan tampak mengecil.

2.4.4 Erosi

Proses erosi merupakan kebalikan dari proses dilasi. Jika dalam proses dilasi menghasilkan objek yang lebih luas maka dalam proses erosi akan menghasilkan objek yang menyempit (mengecil). Lubang pada objek juga akan tampak membesar seiring menyempitnya batas objek tersebut.

Proses erosi dilakukan dengan membandingkan setiap piksel citra *input* dengan nilai pusat SE dengan cara melapiskan SE dengan citra sehingga pusat SE tepat dengan posisi piksel citra yang diproses. Jika semua piksel pada SE tepat sama dengan semua nilai piksel objek (*foreground*) citra maka piksel *input* di-set nilainya dengan nilai piksel *foreground*, bila tidak maka *input* piksel diberi nilai piksel

background. Proses serupa dilanjutkan dengan menggerakkan SE piksel demi piksel pada citra *input*. Gambar 2.4 berikut ini mengilustrasikan citra sebelum dan sesudah proses erosi dengan SE berukuran 3 x 3 dengan semua elemen SE bernilai 1.



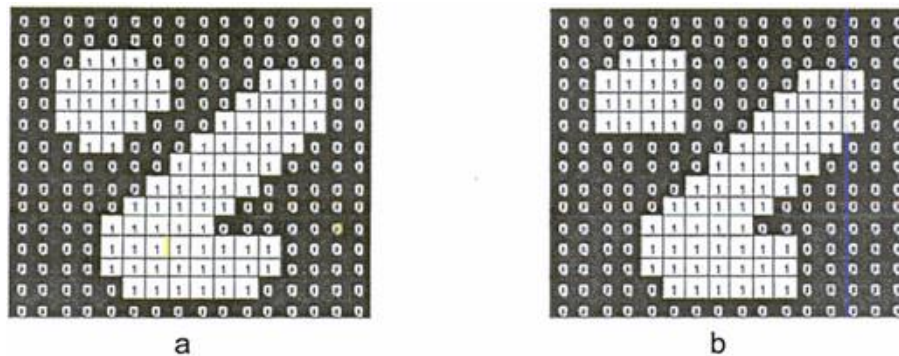
Gambar 2.4 Proses Erosi

Sumber: [13]

Dari gambar di atas terlihat hasil proses erosi menyebabkan objek mengecil. Semakin besar *kernel* yang digunakan maka hasil yang akan didapatkan akan semakin kecil. Begitu juga apabila proses erosi dilakukan berulang-ulang akan terus mengecilkan objek walaupun hanya menggunakan SE berukuran kecil.

2.4.5 Opening

Operasi *opening* merupakan kombinasi dari proses dilasi dan erosi dimana proses dilasi dilakukan setelah proses erosi dengan menggunakan SE yang sama. Operasi *opening* akan mencegah penurunan ukuran objek secara keseluruhan. Gambar 2.5 menunjukkan contoh operasi *opening* menggunakan SE berukuran 3 x 3 dengan semua elemen SE bernilai 1.



Gambar 2.5 Operasi *Opening* (a) Citra Asli (b) Citra Hasil

Sumber: [13]

2.5 Ekstraksi Fitur

Ekstraksi fitur (*Feature Extraction*) merupakan bagian fundamental dari analisis citra. Fitur adalah karakteristik unik dari suatu objek. Karakteristik fitur yang baik sebisa mungkin memenuhi persyaratan berikut.

1. Dapat membedakan suatu objek dengan yang lainnya (*discrimination*).
2. Memperhatikan kompleksitas komputasi dalam memperoleh fitur. Kompleksitas komputasi yang tinggi tentu akan menjadi beban tersendiri dalam menemukan suatu fitur.
3. Tidak terikat (*independence*) dalam arti bersifat invarian terhadap berbagai transformasi (rotasi, penskalaan, pergeseran dan lain sebagainya).
4. Jumlahnya sedikit, karena fitur yang jumlahnya sedikit akan dapat menghemat waktu komputasi dan ruang penyimpanan untuk proses selanjutnya (proses pemanfaatan fitur) [13].

2.6 Jaringan Saraf Tiruan

Jaringan saraf tiruan atau *Artificial Neural Network* yang sering disingkat ANN merupakan model jaringan neural yang meniru prinsip kerja dari neuron otak manusia (neuron biologis). ANN pertama kali muncul setelah model sederhana dari neuron buatan diperkenalkan oleh *McCulloch* dan *Pitts* pada tahun 1943. Model sederhana tersebut dibuat berdasarkan fungsi neuron biologis yang merupakan dasar unit pensinyalan dari sistem saraf.

Jaringan saraf tiruan memiliki beberapa kemampuan seperti yang dimiliki otak manusia, yaitu:

1. Kemampuan untuk belajar dari pengalaman.
2. Kemampuan melakukan perumpamaan (*generalization*) terhadap *input* baru dari pengalaman yang dimilikinya.
3. Kemampuan memisahkan (*abstraction*) karakteristik penting dari *input* yang mengandung data yang tidak penting [13].

Jaringan saraf yang memiliki beberapa neuron dan adanya hubungan antara neuron-neuron tersebut. Neuron akan mengirimkan informasi yang diterima kepada neuron-neuron yang lain melalui hubungan koneksi. Pada jaringan saraf tiruan,

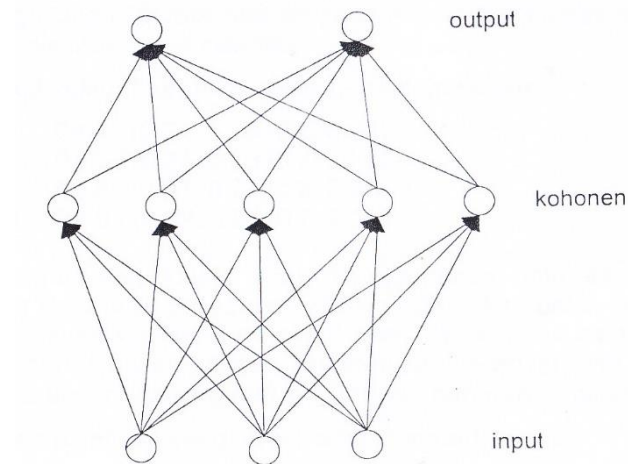
hubungan koneksi ini dikenal sebagai bobot. Informasi yang akan dikirimkan disimpan dalam suatu bobot dengan nilai tertentu. Informasi yang datang akan diproses dengan fungsi perambatan untuk menjumlahkan nilai dari semua bobot yang masuk. Kemudian hasil penjumlahan akan dibandingkan dengan nilai ambang (*threshold*) tertentu melalui fungsi aktivasi. Jika nilai *input* melewati nilai ambang, maka neuron akan diaktifkan, jika tidak maka sebaliknya. Apabila neuron diaktifkan, maka neuron tersebut akan mengirimkan sinyal melalui bobot-bobot kepada *output* ke semua neuron yang berhubungan [17].

2.7 *Learning Vector Quantization*

Learning Vector Quantization merupakan salah satu jaringan saraf tiruan yang melakukan pembelajaran secara terawasi. LVQ mengklasifikasikan *input* secara berkelompok ke dalam kelas yang sudah didefinisikan melalui jaringan yang telah dilatih. Dengan kata lain LVQ mendapatkan *input* dan mengelompokkan ke dalam *m output* [13]. LVQ berada pada lapisan kompetitif. Lapisan kompetitif akan melakukan pembelajaran secara otomatis untuk mengklasifikasikan vektor-vektor *input*. Hasil kelas yang didapatkan dari lapisan kompetitif ini tergantung pada jarak dari masing-masing vektor *input*. Suatu vektor *input* akan diletakkan pada kelas yang sama dengan unit *output*, apabila memiliki vektor bobot yang terdekat dengan vektor *input*-nya [17].

2.7.1 Arsitektur LVQ

Arsitektur LVQ pada dasarnya memiliki bentuk aksitektur yang sama dengan kohonen *Self Organizing Map* (SOM). Namun pada LVQ tidak ada struktur topologi ketetanggaan pada unit *output*, dan setiap unit *output*-nya merepresentasikan suatu kelas tertentu. Arsitektur LVQ terdiri dari *input*, lapisan kohonen dan lapisan *output* [13]. Arsitektur LVQ ini ditunjukkan pada gambar 2.6 berikut.



Gambar 2.6 Arsitektur Jaringan *Learning Vector Quantization*

Sumber: [13]

2.7.2 Algoritma LVQ

Tujuan dari algoritma LVQ adalah untuk menemukan unit *output* yang terdekat dengan vektor *input*-nya. Untuk mencapai tujuan tersebut, maka akan dilakukan pembaruan nilai bobot. Apabila target yang dihasilkan sama dengan kelasnya, maka nilai bobot akan diperbarui agar lebih mendekati x (vektor *input*). Sedangkan apabila target tidak sama dengan kelasnya, maka bobot baru dijauhkan dari x . Secara garis besar, algoritma LVQ adalah sebagai berikut.

1. Tetapkan: bobot (w), maksimum *epoch* (*MaxEpoch*), *error* minimum yang diharapkan (ϵ), *learning rate* (α).
2. Masukkan:
 - *Input* : $x(m,n)$;
 - *Target* : $T(1,n)$;
3. Tetapkan kondisi awal: $epoch = 0$;
4. Kerjakan jika: ($epoch < MaxEpoch$) atau ($\alpha > \epsilon$)
 - a. $epoch = epoch + 1$;
 - b. kerjakan untuk $i = 1$ sampai n
 - i. Temukan J sedemikian hingga $\|x - w_j\|$ minimum (disebut sebagai C_j)
 - ii. Perbaharui nilai w_j dengan ketentuan:
Jika $T = C_j$ maka:

$$w_j(\text{baru}) = w_j(\text{lama}) + \alpha (x - w_j(\text{lama})) \quad (2.5)$$

Jika $T \neq C_j$ maka:

$$w_j(\text{baru}) = w_j(\text{lama}) - \alpha (x - w_j(\text{lama})) \quad (2.6)$$

- c. Kurangi nilai α .
- d. Cek kondisi berhenti.

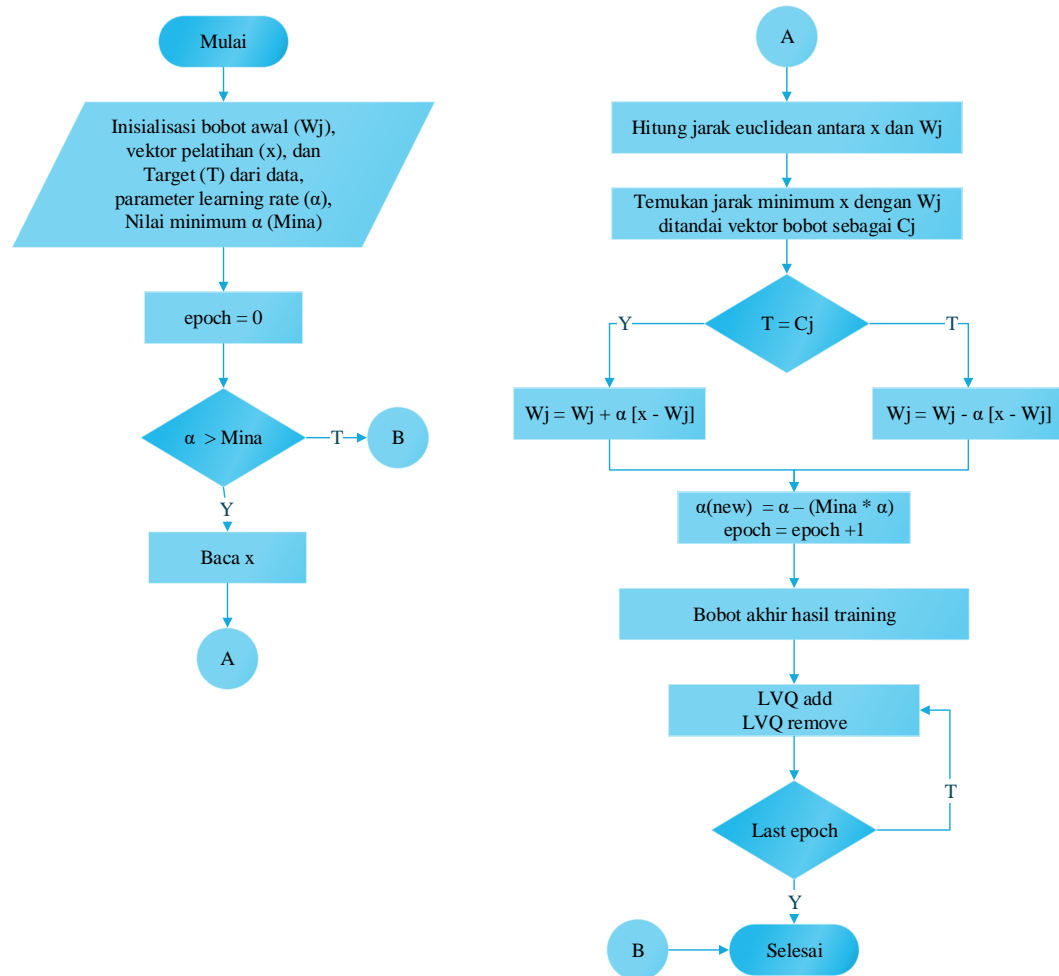
Kondisi iterasi/*epoch* berhenti apabila kondisi pada langkah 4 telah terpenuhi yaitu jumlah iterasi mencapai jumlah maksimal iterasi yang dimasukkan atau nilai *learning rate* (α) lebih besar dari *error* minimum yang diharapkan (*eps*).

Keterangan notasi:

- w : Vektor bobot.
- epoch* : Perulangan.
- MaxEpoch* : Maksimal perulangan.
- eps* : *Error* minimum yang diharapkan.
- α : *Learning rate*.
- x : Vektor *input* / vektor *training*.
- T : Kategori yang benar untuk vektor *training*.
- C_j : Kategori yang diwakilkan oleh unit *output* ke- j .
- $\|x - w_j\|$: Jarak *Euclidean* antara vektor *input* dan vektor bobot untuk unit *output* ke- j .
- $w_j(\text{baru})$: Vektor bobot baru.
- $w_j(\text{lama})$: Vektor bobot lama [17].

2.7.3 Adaptive LVQ

Adaptive LVQ merupakan modifikasi yang memperbaiki kinerja metode klasifikasi LVQ. *Adaptive LVQ* merupakan cara mengolah data *training* dengan cara mengurangi data yang kurang baik atau kurang bermanfaat dan menambahkan data yang berpotensi menguntungkan dimana *learning rate* berubah berdasarkan iterasi. Berikut gambar 2.7 adalah *flowchart* algoritma *Adaptive LVQ*.



Gambar 2.7 Flowchart Algoritma Adaptive LVQ

Sumber: [6]

Proses pertama yang dilakukan merupakan proses LVQ yang melakukan perbaikan bobot seperti pada persamaan 2.5 dan 2.6. Sedangkan proses *Adaptive LVQ* terdiri dari dua metode yaitu *LVQadd* untuk menambahkan data *training* saat data yang ada membuat banyak kesalahan untuk dibelajarkan kembali, sedangkan *LVQremove* menghapus data *training* jika data yang ada dapat mengakibatkan penurunan pada kesalahan klasifikasi. *LVQadd* dan *LVQremove* ini bekerja jauh lebih efisien daripada menggunakan dan menyebarkan data *training* untuk semua kelas secara merata [6].

2.7.4 Contoh Perhitungan

Berikut ini adalah contoh tahapan-tahapan proses *Adaptive LVQ* yang dijalankan:

1. Data set *training*.

Terdapat 4 data *training* berupa matriks 2x2 sebagai berikut.

Data ke 1		Data ke 2	
1	1	1	0
1	0	1	1
Data ke 3		Data ke 4	
0	1	0	0
1	0	1	1

2. Konversi matriks ke dalam *input* vektor.

Matriks-matriks yang ada di konversi menjadi *input* vektor dengan 2 kelas sebagai berikut:

Tabel 2.1 *Input Vektor Data Training*

No.	Vektor	Target Kelas
1.	(1,1,1,0)	1
2.	(1,0,1,1)	2
3.	(0,1,1,0)	1
4.	(0,0,1,1)	2

Dua *input* vektor pertama digunakan sebagai inisialisasi bobot yang mewakili masing-masing kelas 1 dan 2 (secara simbolis $C_1 = 1$ dan $C_2 = 2$). Vektor yang digunakan dalam pelatihan antara lain (0,1,1,0) dan (0,0,1,1).

3. Proses *Training*, yang terdiri dari:

a. Inisialisasi *parameters*:

Misalkan dilakukan perhitungan 1 kali iterasi (1 *epoch*) dan inisialisasi *learning rate* (α) = 0.1 dengan minimum *learning rate* (min α) sebesar 0.001 dan pengurangan *learning rate* sebesar $\alpha - (\min \alpha * \alpha)$.

b. Inisialisasi bobot.

$$w_1 = (1,1,1,0)$$

$$w_2 = (1,0,1,1)$$

c. Untuk vektor $x = (0,1,1,0)$ dan $T = 1$, lakukan:

1. Perhitungan jarak terdekat dengan menggunakan jarak *Euclidean*.

$$d_1 = \sqrt{(0-1)^2 + (1-1)^2 + (1-1)^2 + (0-0)^2} = 1$$

$$d_2 = \sqrt{(0-1)^2 + (1-0)^2 + (1-1)^2 + (0-1)^2} = 1.73$$

2. Jarak terdekat pada w_1 , $J = 1$ dan $C_1 = 1$. Target sama dengan kelas.

3. Perbarui w_1 :

$$w_1 = (1,1,1,0) + 0.1[(0,1,1,0) - (1,1,1,0)] = (0.9,1,1,0)$$

d. Untuk vektor $x = (0,0,1,1)$ dan $T = 2$, lakukan:

1. Perhitungan jarak terdekat dengan menggunakan jarak *Euclidean*.

$$d_1 = \sqrt{(0-1)^2 + (0-1)^2 + (1-1)^2 + (1-0)^2} = 1.73$$

$$d_2 = \sqrt{(0-1)^2 + (0-0)^2 + (1-1)^2 + (1-1)^2} = 1$$

2. Jarak terdekat pada w_2 , $J = 2$ dan $C_2 = 2$. Target sama dengan kelas.

3. Perbarui w_2 :

$$w_2 = (1,0,1,1) + 0.1[(0,0,1,1) - (1,0,1,1)] = (0.9,0,1,1)$$

e. Kurangi nilai α .

$$\alpha = \alpha - (\min \alpha * \alpha) = 0.1 - (0.001 * 0.1) = 0.09$$

Bobot yang didapatkan pada proses *training* yaitu:

$$w_1 = (0.9,1,1,0)$$

$$w_2 = (0.9,0,1,1)$$

4. Proses *Testing*.

Input vektor yang digunakan dalam proses *testing* misalnya $x = (0,0,1,1)$. Maka dilakukan proses *testing* sebagai berikut:

a. Perhitungan jarak terdekat dengan menggunakan jarak *Euclidean* dan bobot (w) baru yang didapatkan dari proses *training*.

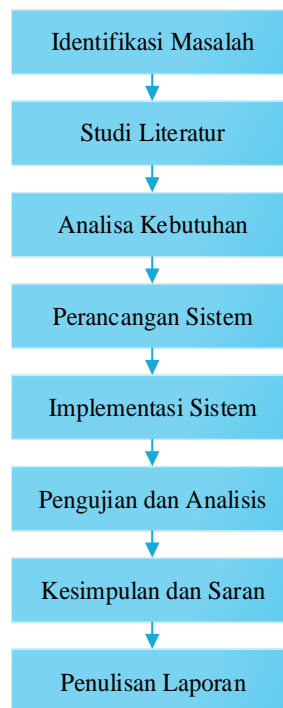
$$d_1 = \sqrt{(0-0.9)^2 + (0-1)^2 + (1-1)^2 + (1-0)^2} = 1.58$$

$$d_2 = \sqrt{(0-0.9)^2 + (0-0)^2 + (1-1)^2 + (1-1)^2} = 0.7$$

b. Jarak terdekat pada w_2 , $j = 2$ dan $C_2 = 2$. Maka *input* vektor tersebut diklasifikasikan atau masuk ke dalam kelas 2.

BAB III. METODOLOGI PENELITIAN

Pada bab ini membahas mengenai metode penelitian yang digunakan dalam penelitian penerapan algoritma *Adaptive Learning Vector Quantization* pada pengenalan huruf bahasa isyarat. Metode penelitian yang dilakukan terdiri dari identifikasi masalah, studi literatur, analisa kebutuhan, pengumpulan data, perancangan sistem, implementasi sistem, pengujian dan analisis, pengambilan kesimpulan dan saran serta penulisan laporan. Berikut gambar 3.1 diagram alir metode penelitian yang dilakukan.



Gambar 3.1 Diagram Alir Metode Penelitian

3.1 Identifikasi Masalah

Identifikasi masalah merupakan tahap awal yang dilakukan dalam penelitian, yang digunakan untuk menentukan objek yang akan dimuat dalam sebuah penelitian. Objek yang akan diidentifikasi sebagai suatu permasalahan akan dimuat berdasarkan latar belakangnya, dan juga mencari solusi apa saja yang telah dilakukan selama ini serta penelitian-penelitian yang pernah dilakukan terkait dengan objek tersebut. Permasalahan yang diambil dalam penelitian ini adalah bagaimana cara mengenalkan huruf bahasa isyarat kepada warga SLB Islam

Yasindo Malang yang belum mengenal huruf bahasa isyarat sehingga memudahkan komunikasi antara tuna rungu dan tuna wicara dengan orang normal.

3.2 Studi Literatur

Studi literatur bertujuan untuk mengumpulkan dan mempelajari teori-teori dasar yang diterapkan dalam penelitian ini. Literatur yang digunakan bersumber dari buku, jurnal ilmiah, dan situs internet serta literatur lain yang berkaitan. Teori utama yang berkaitan dengan penulisan laporan skripsi ini meliputi:

1. Bahasa Isyarat.
2. Pengolahan Citra Digital.
3. Jaringan Saraf Tiruan.
4. Algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal*.

3.3 Analisa Kebutuhan

Analisa kebutuhan bertujuan untuk menganalisis dan mendapatkan semua kebutuhan yang diperlukan dalam membangun sistem pengenalan huruf Bahasa isyarat dengan algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal*. Analisa kebutuhan sistem yang akan dirancang meliputi masukan (*input*), proses, dan keluaran (*output*) serta kebutuhan *hardware* dan *software* untuk dapat melakukan pengenalan huruf bahasa isyarat sesuai dengan tujuan yang diharapkan.

1. Masukan (*Input*).

Masukan dari sistem ini berupa data yang diberikan oleh pengguna kepada sistem yang terdiri dari: Citra bentuk tangan untuk Bahasa isyarat alfabet statis yang meliputi 24 huruf yaitu A, B, C, D, E, F, G, H, I, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y.

2. Proses

Proses yang dilakukan setelah masukan dari sistem diterima yaitu berupa *load* citra tangan dari direktori. Setelah itu, dilakukan *preprocessing* dan ekstraksi ciri serta pembelajaran. Pada tahap pembelajaran ini, perhitungan algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal* dilakukan.

3. Keluaran (*output*)

Hasil keluaran dari sistem yang akan dibangun berupa tulisan alfabet yang sesuai dengan hasil pengenalan citra huruf Bahasa isyarat.

4. *Hardware* dan *Software*

Kebutuhan-kebutuhan yang digunakan dalam pembuatan sistem diantaranya adalah:

a. Kebutuhan *hardware*, meliputi:

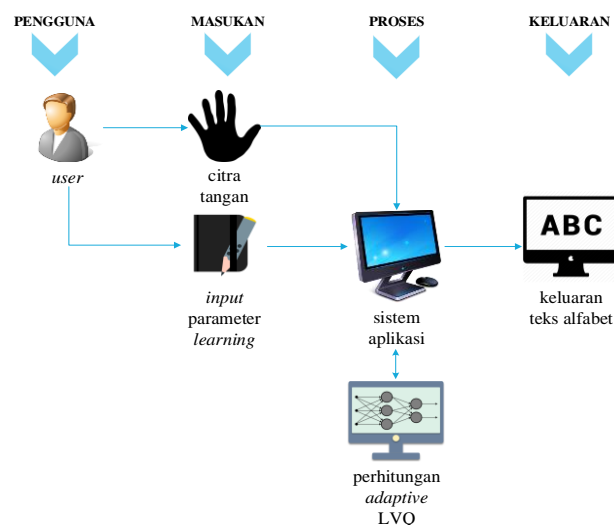
- Laptop atau PC.
- Kamera.

b. Kebutuhan *software*, meliputi:

- Visual Studio 2015 sebagai editor untuk pembuatan sistem menggunakan bahasa pemrograman Visual Basic.NET.
- SQL Server 2014 sebagai penyimpanan *database*.
- *Library* RedMatter dan Math.NET Numerics sebagai tools yang dapat membantu dalam proses tahapan dari kinerja sistem.

3.4 Perancangan Sistem

Perancangan sistem dilakukan setelah semua kebutuhan dari sistem didapatkan melalui analisa kebutuhan sebelumnya. Dalam rangka memenuhi semua kebutuhan fungsional dari sistem, maka diperlukan komponen-komponen yang saling berhubungan untuk menjalankan kinerja sistem seperti pada gambar 3.2 berikut.



Gambar 3.2 Arsitektur Sistem Pengenalan Huruf Bahasa Isyarat

3.5 Implementasi Sistem

Pada tahap implementasi sistem akan dilakukan pembuatan sistem yang mengacu pada tahap perancangan sistem yang dilakukan sebelumnya. Implementasi perangkat lunak dilakukan dengan menggunakan model pemrograman berorientasi objek yaitu menggunakan bahasa pemrograman Visual Basic.NET dengan *software* Visual Studio 2015.

3.6 Pengujian dan Analisis

Pada tahap ini dilakukan pengujian berdasarkan implementasi sistem yang dibuat. Pengujian yang dilakukan untuk mengetahui bahwa sistem dapat bekerja dengan baik sesuai tujuan yang diharapkan yaitu sistem dapat melakukan menerjemahkan citra tangan menjadi keluaran tulisan alfabet sesuai dengan perhitungan algoritma *Learning Vector Quantization with Adaptive Prototype Addition and Removal*.

Analisis dari hasil uji coba yang telah dilakukan digunakan untuk menarik kesimpulan dari hasil kinerja sistem pengenalan huruf bahasa isyarat. Analisis dari sistem ini bertujuan untuk menguji kinerja penggunaan dari alur proses dan ekstraksi ciri serta algoritma *Adaptive LVQ* pada pengenalan huruf bahasa isyarat.

3.7 Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan berdasarkan hasil pengujian dan analisis yang merupakan hasil dari keseluruhan proses penelitian. Selain itu juga dapat memberikan saran yang bertujuan untuk memberikan pertimbangan untuk penelitian sistem selanjutnya.

3.8 Penulisan Laporan

Tahap terakhir dari penelitian yaitu penulisan laporan yang berisi dokumentasi dari keseluruhan proses penelitian yang akan berguna untuk membantu dalam penelitian sistem selanjutnya.

BAB IV. ANALISIS DAN PERANCANGAN

Sub bab ini membahas tentang tahap awal dari desain sistem yang akan diimplementasikan, perancangan antarmuka sistem dan pengujian yang akan dilakukan. Perancangan sistem dibangun berdasarkan hasil studi literatur dan analisa kebutuhan. Pada tahap analisa dan perancangan meliputi analisa perangkat lunak, perancangan perangkat lunak dan perancangan pengujian.

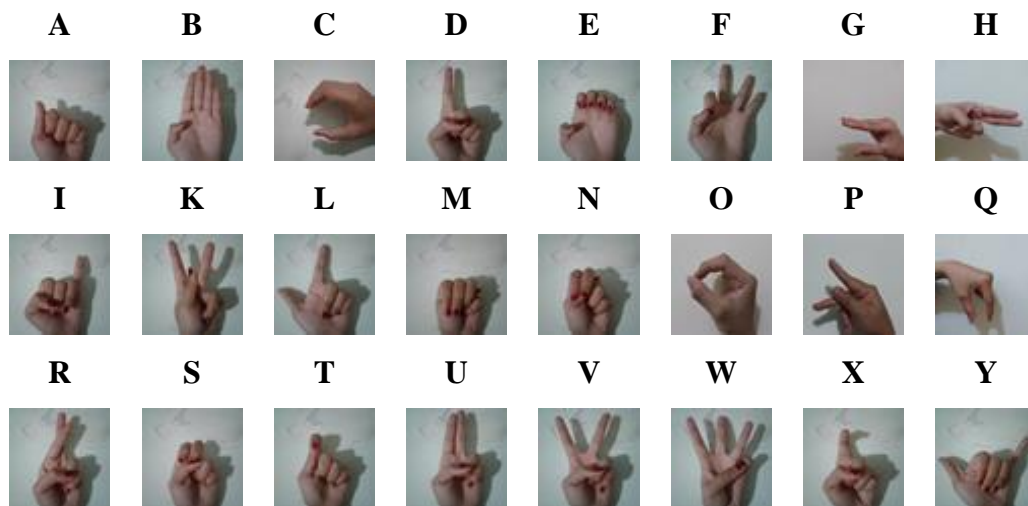
4.1 Analisa Perangkat Lunak

Analisa perangkat lunak sistem pengenalan huruf bahasa isyarat terdiri dari analisa kebutuhan data dan analisa kebutuhan perangkat lunak secara fungsional.

4.1.1 Analisa Kebutuhan Data

Analisa kebutuhan data bertujuan untuk mengetahui kebutuhan data yang diperlukan oleh sistem baik masukan maupun keluaran agar sesuai dengan tujuan sistem. Data yang diperlukan dalam sistem antara lain sebagai berikut:

1. Data masukan citra bentuk tangan huruf bahasa isyarat (A-Z) statis, yaitu tanpa huruf J dan Z, yang menggunakan gerakan. Citra bentuk pola tangan yang digunakan dalam sistem antara lain seperti pada gambar 4.1.



Gambar 4.1 Data Masukan Bentuk Pola Tangan Huruf Bahasa Isyarat

2. Data keluaran dari sistem berupa tulisan huruf sesuai dengan kelas yang digunakan dalam penelitian, yaitu huruf alfabet tanpa huruf J dan Z.

4.1.2 Analisa Kebutuhan Perangkat Lunak

Analisa kebutuhan perangkat lunak bertujuan untuk merancang kebutuhan-kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan sesuai dengan tujuan yang diharapkan. Analisa kebutuhan perangkat lunak akan dimodelkan dalam diagram *use case* dan diagram kelas.

4.1.2.1 Diagram *use case*

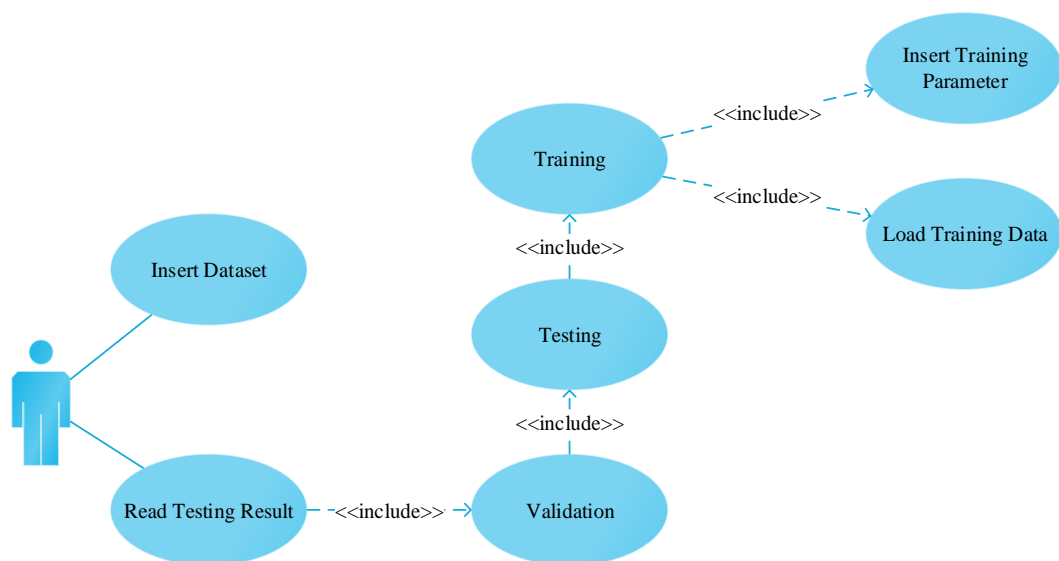
Diagram *use case* digunakan untuk mendeskripsikan segala kegiatan aktor pada sistem. Kebutuhan fungsional sistem digambarkan dengan tabel yang berisi nama *use case* yang digunakan, aktor yang berperan dan kebutuhan yang disediakan oleh sistem. Daftar kebutuhan sistem ditunjukkan pada tabel 4.1.

Tabel 4.1 Daftar Kebutuhan Sistem

Nama Use Case	Aktor	Kebutuhan
<i>Insert Dataset</i>	<i>User</i>	Sistem menyediakan fitur untuk menambahkan <i>dataset</i> yang digunakan untuk proses <i>training</i> , <i>validation</i> , dan <i>testing</i> . Data yang digunakan berupa citra tangan yang dimuat melalui direktori.
<i>Load Training Data</i>	<i>User</i>	Sistem menyediakan fungsi untuk memuat data <i>training</i> yang digunakan untuk menganalisis citra untuk diambil cirinya dalam proses <i>training</i> .
<i>Insert Training Parameter</i>	<i>User</i>	Sistem menyediakan tempat untuk menerima masukan parameter yang digunakan saat proses <i>training</i> . Parameter masukan berupa nilai <i>learning rate</i> , minimum <i>learning rate</i> , reduksi <i>learning rate</i> , dan maksimum <i>epoch</i> .
<i>Training</i>	<i>User</i>	Sistem menyediakan fungsi pelatihan yang digunakan untuk melakukan perhitungan algoritma pembelajaran <i>Learning Vector Quantization</i> .

<i>Testing</i>	<i>User</i>	Sistem menyediakan fungsi untuk melakukan pengenalan menggunakan algoritma yang telah ditentukan setelah proses pelatihan dilakukan.
<i>Validation</i>	<i>User</i>	Sistem menyediakan fungsi validasi yang digunakan untuk melakukan validasi data setelah proses pengenalan dilakukan berdasarkan metode <i>Adaptive Prototype Addition and Removal</i> .
<i>Read Testing Result</i>	<i>User</i>	Sistem menyediakan halaman untuk menampilkan hasil dari pengenalan berupa tulisan huruf bahasa isyarat.

Berdasarkan daftar kebutuhan sistem di atas, maka rancangan *use case* sistem pengenalan huruf bahasa isyarat ditunjukkan pada gambar 4.2.



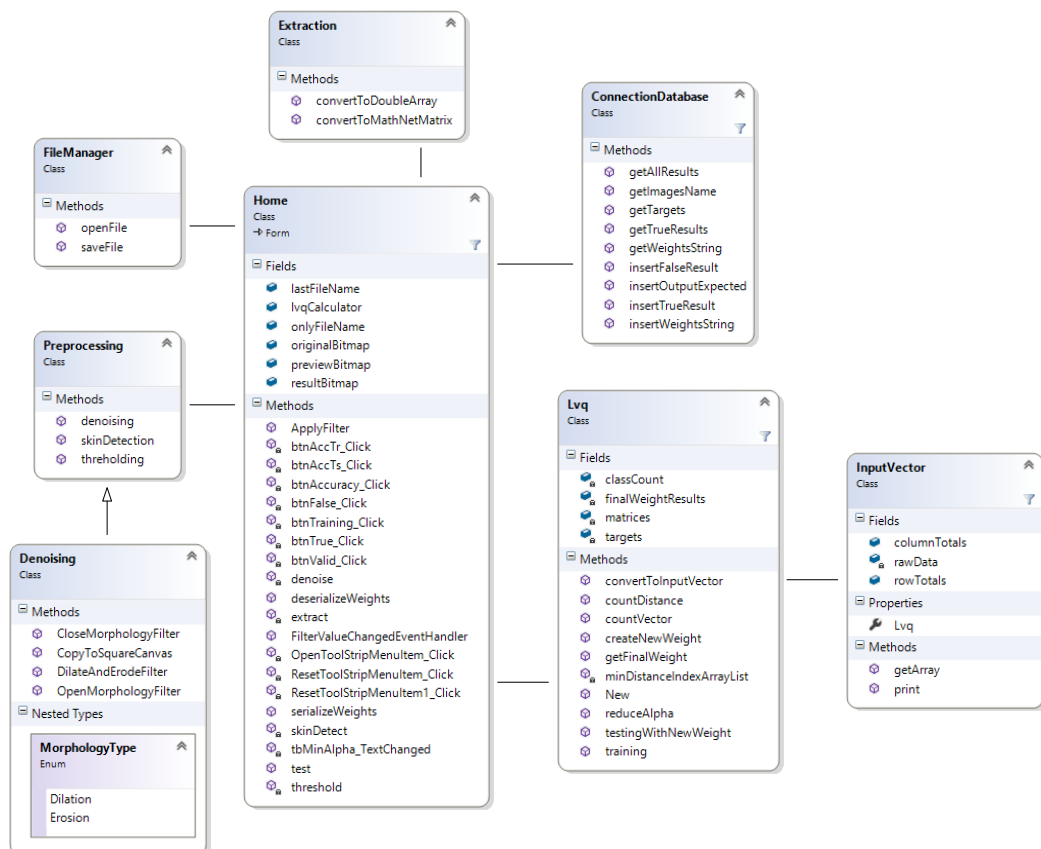
Gambar 4.2 Diagram *Use Case* Sistem

Pada diagram *use case*, *user* dapat melakukan penambahan data yang digunakan untuk proses *testing* dan melihat hasil proses *testing* dari masukan yang diberikan oleh *user* pada sistem. Untuk melihat hasil proses *testing* maka dilakukan proses *training* terlebih dahulu. *Training* yang dilakukan oleh sistem dengan cara memuat data citra yang tersimpan dalam direktori dan membutuhkan masukan parameter yang diberikan oleh *user* antara lain nilai *learning rate*, minimum

learning rate, reduksi *learning rate*, dan maksimum *epoch* pada algoritma yang digunakan.

4.1.2.2 Diagram kelas

Sistem pengenalan huruf bahasa isyarat dibangun dengan menggunakan model *Object Oriented Programming* (OOP) yang berorientasi pada interaksi antar objek. Interaksi antar objek ini digambarkan dalam sebuah diagram kelas. Pada diagram kelas terdapat kelas *FileManager* untuk mengatur *image file*, kelas *Preprocessing* yang terdiri dari fungsi *skinDetection* untuk deteksi warna kulit, *thresholding* untuk memisahkan objek (*foreground*) dan latar belakang (*background*), dan *denoising* untuk mengurangi *noise* pada citra. Selain itu terdapat kelas *InputVector* dan *Extraction* untuk melakukan proses ekstraksi fitur citra dan kelas *Lvq* untuk proses perhitungan algoritma pembelajaran serta kelas *Home* yang merupakan halaman utama sistem. Rancangan kelas diagram yang digunakan dalam pembuatan sistem ditunjukkan pada gambar 4.3.



Gambar 4.3 Diagram Kelas Sistem

4.1.3 Teknik Pengambilan Data

Tahapan awal yang dilakukan oleh sistem adalah tahapan penambahan data untuk pengenalan huruf bahasa isyarat. Teknik pengambilan data dilakukan agar *sample* yang diuji dalam sistem memiliki perlakuan yang sama. Dalam pengambilan data dibutuhkan:

1. Merk dan Tipe Kamera : Samsung J1 (2016)
2. Resolusi Kamera : CMOS 5.0 *Megapixel*
3. Resolusi Citra : 50 x 50 *Pixel*
4. Jarak Pengambilan Citra : 20 cm
5. Pencahayaan : Di dalam ruang tertutup dengan cahaya lampu pada malam hari.

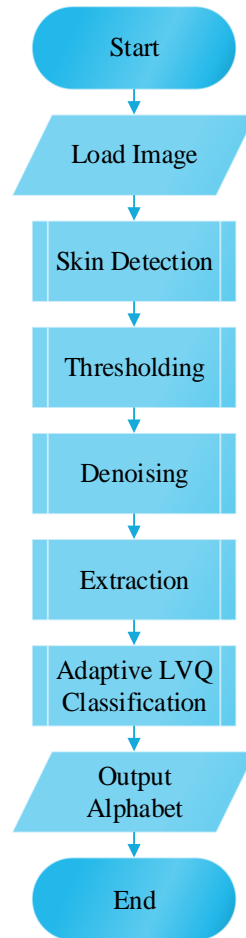
4.2 Perancangan Diagram Alur Sistem

Pada perancangan diagram alur sistem menggambarkan bagaimana sistem bekerja untuk melakukan pengenalan huruf bahasa isyarat. Rancangan alur sistem yang diimplementasikan adalah sebagai berikut:

1. *User* memuat citra dari direktori ke dalam sistem.
2. Melakukan *preprocessing* terhadap citra yang telah dimuat antara lain:
 - a. *Skin Detection* untuk memisahkan warna kulit dengan latar belakang.
 - b. *Thresholding* untuk mengubah nilai piksel bagian tangan dan latar belakang.
 - c. *Denoising* untuk mengurangi *noise* pada citra dengan menggunakan metode *Opening* yang terdiri dari erosi dan dilasi.
3. Melakukan ekstraksi ciri citra menggunakan fitur biner.
4. Ciri yang didapatkan dijadikan parameter untuk proses klasifikasi pada algoritma *Adaptive LVQ* untuk mengenali citra huruf bahasa isyarat.
5. Menampilkan hasil pengenalan berupa tulisan huruf bahasa isyarat.

Flowchart perancangan diagram alur sistem ini digambarkan pada gambar

4.4.



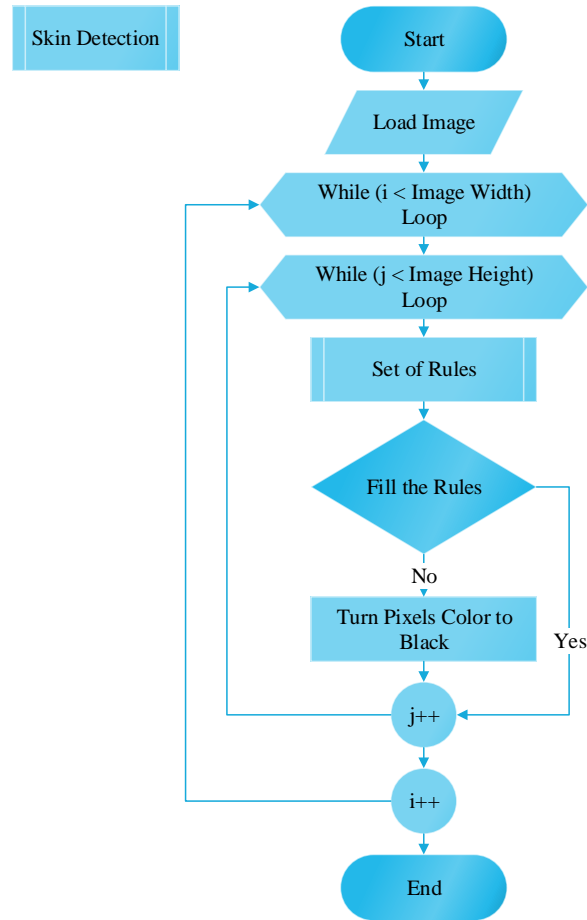
Gambar 4.4 Flowchart Sistem Pengenalan Huruf Bahasa Isyarat

4.2.1 Load Image

Masukan citra diperoleh dengan cara memuat *file* citra dari direktori ke dalam sistem. Citra yang didapat berasal dari tangkapan kamera dengan dilakukan *resize* citra menjadi 50x50 piksel.

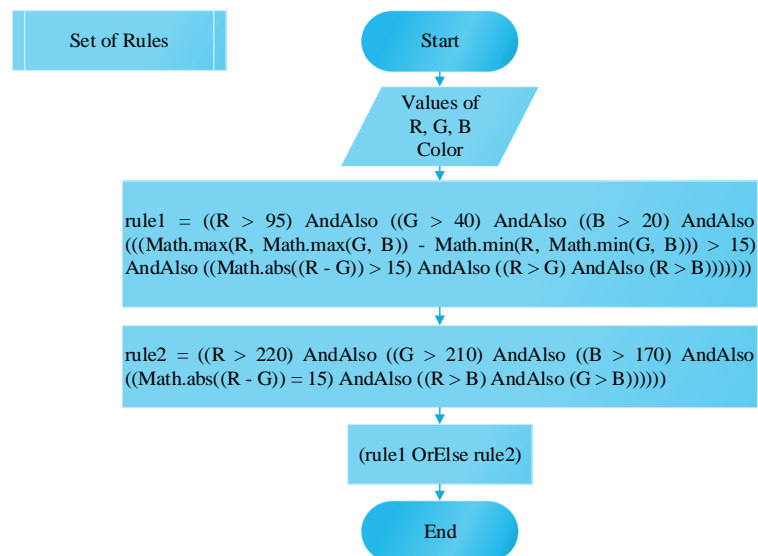
4.2.2 Skin Detection

Proses *skin detection* bertujuan untuk memisahkan warna kulit (*foreground*) dengan latar belakang (*background*) citra. Pada proses ini menggunakan aturan ruang warna RGB seperti pada persamaan 2.3 yang digabungkan dengan operator logika OR dengan persamaan 2.4. Jika memenuhi kedua aturan tersebut, maka warna kulit pada citra akan terdeteksi dan nilai piksel tetap (RGB kulit). Jika bukan merupakan warna kulit, maka nilai piksel diubah menjadi hitam yaitu sebagai latar belakang. Proses *skin detection* ini menggunakan kelas BaseDetector dan SimpleSkinDetector4 pada *library* RedMatter [18].



Gambar 4.5 Flowchart Skin Detection

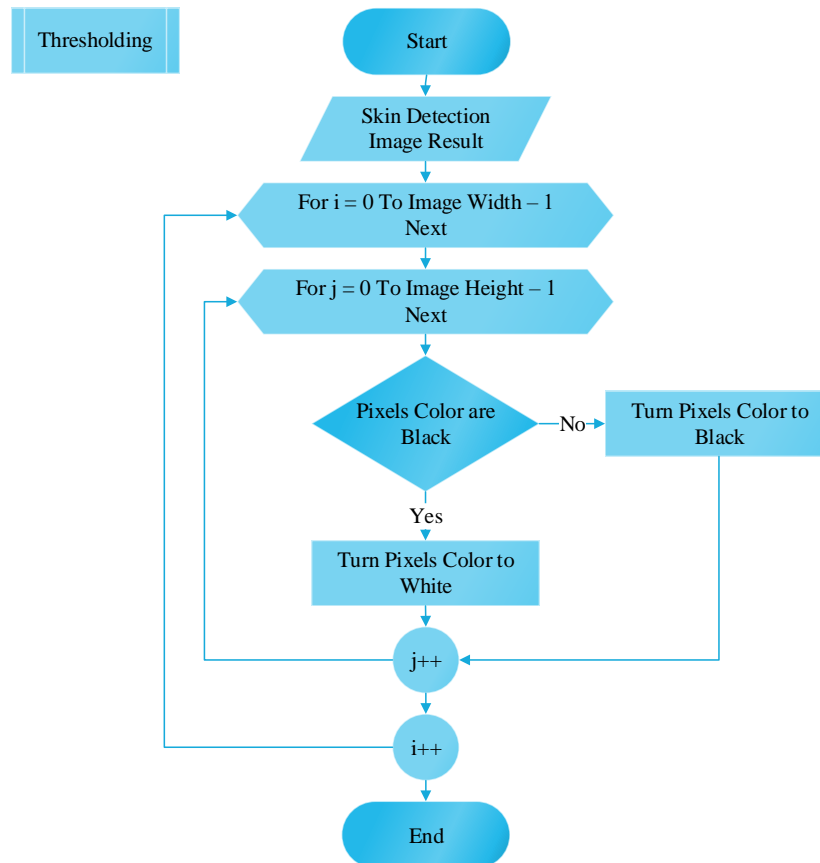
Aturan yang digunakan pada proses *skin detection* secara rinci digambarkan pada gambar 4.6.



Gambar 4.6 Flowchart Set of Rules (RGB Color Space)

4.2.3 Thresholding

Proses *thresholding* bertujuan untuk mengubah citra menjadi 2 warna yaitu putih sebagai latar belakang (*background*) dan hitam sebagai objek (*foreground*) dimana warna kulit menjadi hitam dan latar belakang yang sebelumnya hitam akan menjadi putih. Proses *thresholding* digambarkan pada gambar 4.7.

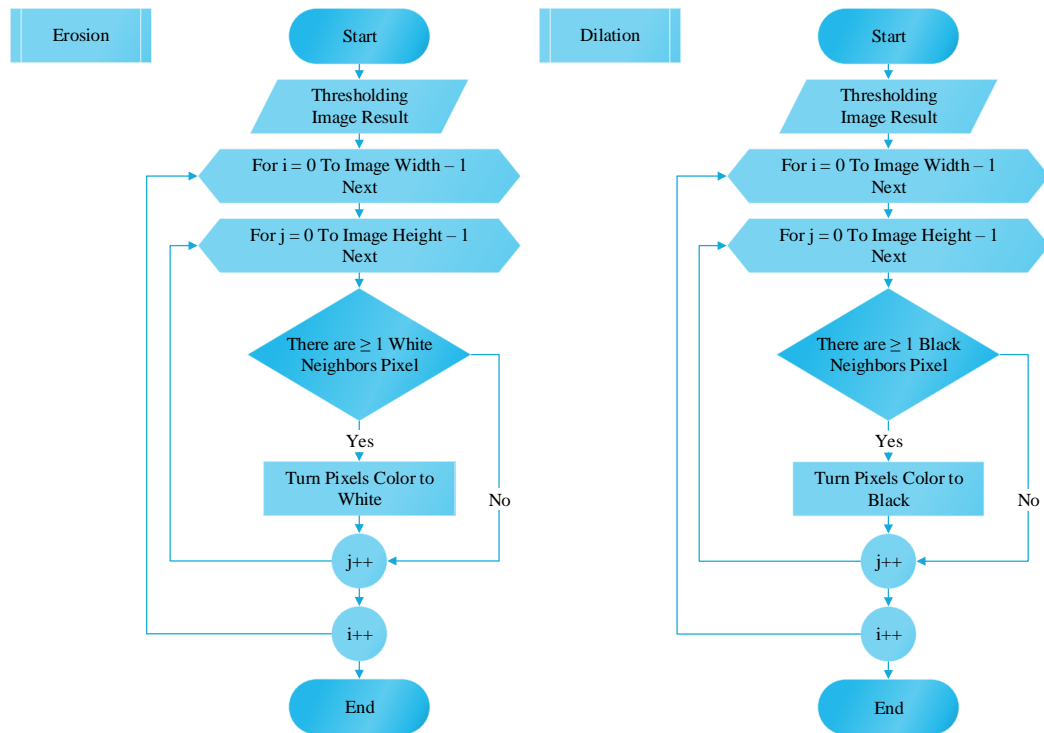


Gambar 4.7 Flowchart Thresholding

4.2.4 Denoising

Proses *denoising* digunakan untuk mereduksi atau mengurangi *noise* yang ada pada citra. Metode yang digunakan adalah proses *opening*. Proses *opening* merupakan kombinasi dari 2 proses morfologi, yaitu erosi dan dilasi. Citra hasil proses *thresholding* dilakukan proses erosi terlebih dahulu yang bertujuan untuk membuat ukuran sebuah citra menjadi lebih kecil sehingga dapat menghapus *noise-noise* yang tersebar acak. Setelah itu dilakukan proses dilasi untuk menambahkan piksel pada batasan dari objek citra sehingga dapat mengembalikan daerah objek yang sebelumnya hilang karena proses erosi. Proses *denoising* ini menggunakan *code snippet* dari situs [19].

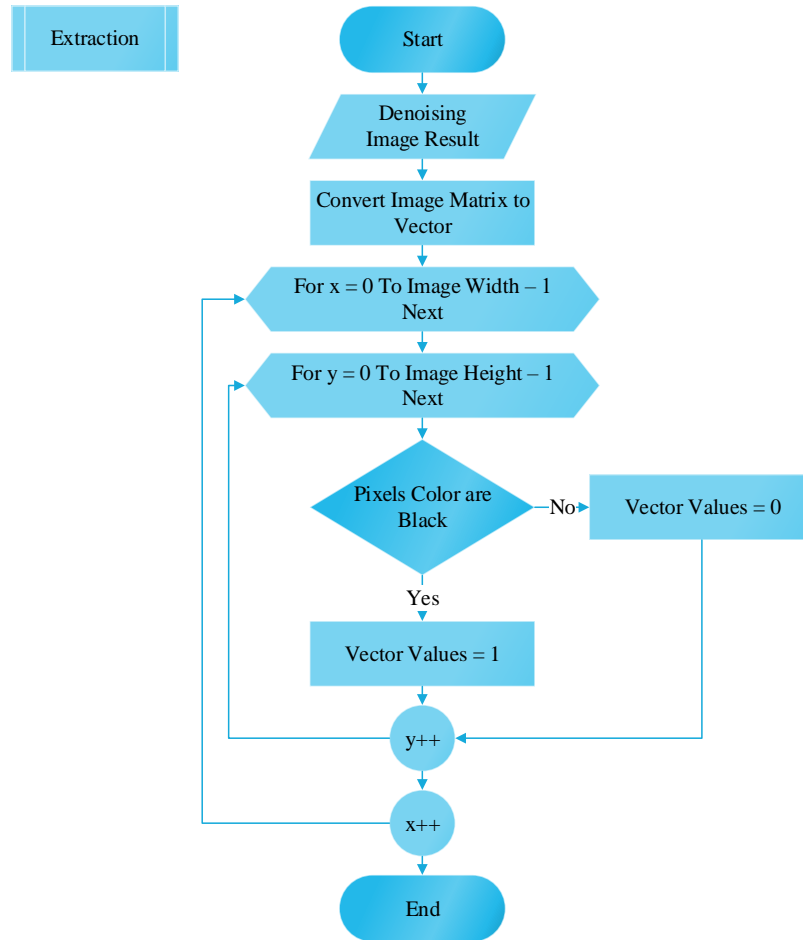
Pada proses *opening* ini menggunakan layer berukuran 3 x 3. Untuk proses erosi, jika pada daerah layer terdapat minimal satu piksel berwarna putih, maka piksel tersebut diberi warna putih. Sedangkan proses dilasi sebaliknya, jika pada daerah layer terdapat minimal satu piksel berwarna hitam, maka piksel tersebut diberi warna hitam. Berikut gambar 4.8 merupakan *flowchart* proses erosi dan dilasi.



Gambar 4.8 *Flowchart* Proses Erosi dan Dilasi

4.2.5 Extraction

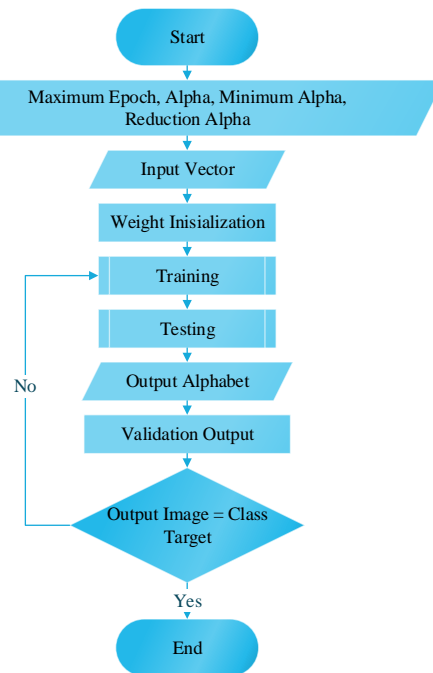
Proses ekstraksi ciri citra merupakan proses yang digunakan untuk tahap pengenalan. Metode yang digunakan adalah fitur biner. Piksel dari citra yang digunakan sampai tahap pengenalan berukuran 50 x 50 piksel. Fitur biner akan mengambil nilai dari tiap piksel citra dan menyimpannya ke dalam bentuk vektor *array* 1 dimensi, dengan panjang vektor *array* adalah 2500. Berikut gambar 4.9 merupakan *flowchart* ekstraksi ciri menggunakan fitur biner.



Gambar 4.9 Flowchart Extraction dengan Fitur Biner

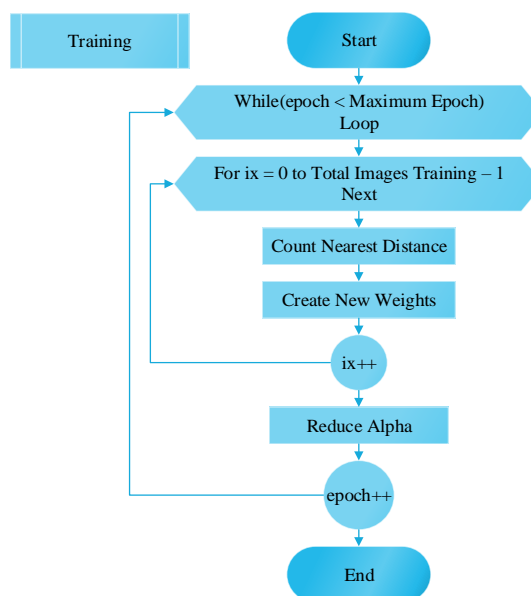
4.2.6 Adaptive LVQ Classification

Setelah didapatkan *input vector* dari proses ekstraksi ciri, selanjutnya *input vector* tersebut digunakan untuk proses klasifikasi dengan algoritma *Adaptive LVQ*. Pada proses klasifikasi dibutuhkan juga parameter nilai *learning rate* (α), minimal *learning rate* (α), reduksi *learning rate* (α) dan maksimal *epoch* yaitu iterasi atau perulangan. Setelah semua parameter *input* terpenuhi, dilakukan proses inisialisasi bobot, proses *training*, *testing* dan *validation*. Pada saat proses *testing* sebuah citra huruf bahasa isyarat dikenali.



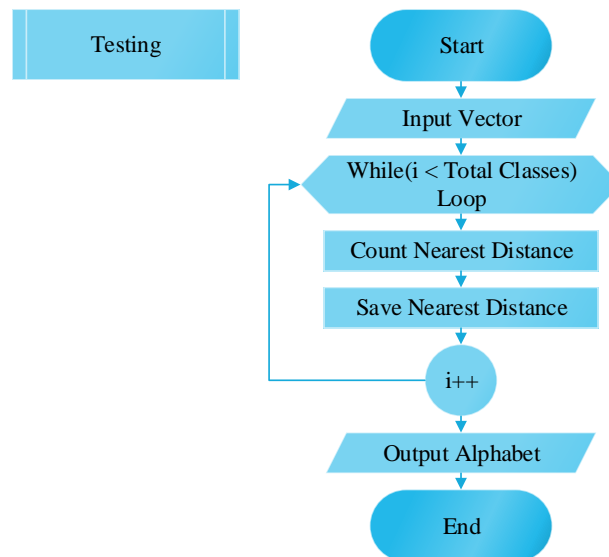
Gambar 4.10 Flowchart Adaptive LVQ Classification

Proses *training* digunakan untuk mengolah data *training* dan menetapkan nilai bobot untuk digunakan pada proses *testing*. Proses *training* dibatasi dengan jumlah maksimal *epoch* yang telah ditentukan dimana setiap *epoch* akan melakukan perubahan bobot untuk membedakan antara kelas satu dengan lainnya. Perubahan bobot dipengaruhi dari perhitungan jarak terdekat pada masing-masing *output*. Jika jarak terdekat yang diperoleh sama dengan targetnya, maka nilai bobot akan ditambah, tetapi jika tidak maka sebaliknya nilai bobot akan dikurangi.



Gambar 4.11 Flowchart Proses Training Adaptive LVQ

Setelah dilakukan proses *training*, maka didapatkan nilai-nilai bobot yang digunakan pada proses *testing*. Pada proses *testing* inilah suatu citra diklasifikasikan termasuk ke dalam kelas mana. Hasil klasifikasi merupakan keluaran kelas yang memiliki nilai perhitungan jarak terdekat antara vektor masukan dengan vektor bobot yang didapatkan dalam proses *training*. *Flowchart* dari proses *testing* ditunjukkan pada gambar 4.12.



Gambar 4.12 *Flowchart Proses Testing Adaptive LVQ*

4.2.7 Output Alphabet

Setelah hasil dari proses klasifikasi didapatkan, maka sistem memberikan keluaran berupa tulisan huruf bahasa isyarat yang berhasil menerjemahkan citra menjadi tulisan yang dapat dibaca.

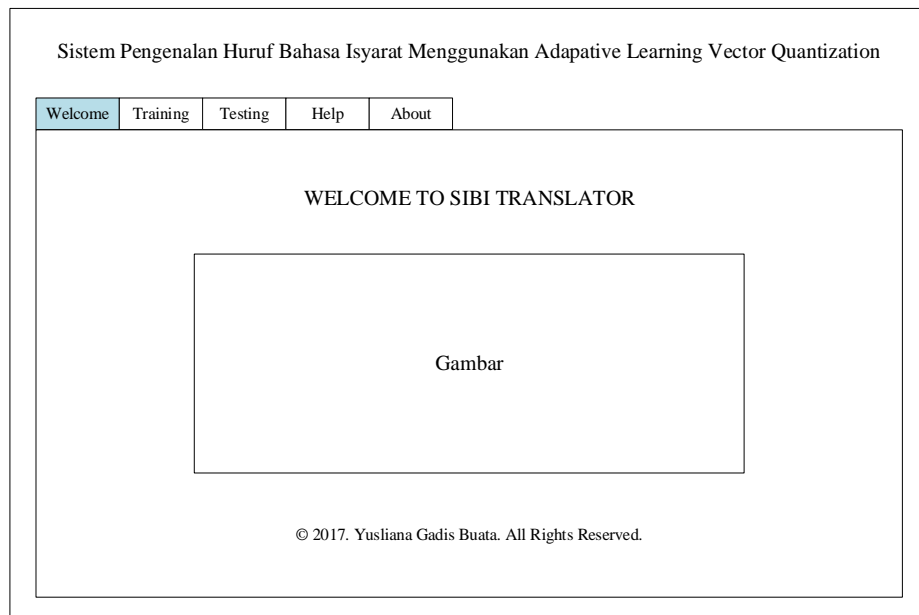
4.3 Perancangan Antarmuka Sistem

Antarmuka sistem merupakan cara sistem dan *user* dapat berkomunikasi. Perancangan antarmuka sistem berguna untuk memberikan gambaran desain tampilan sistem yang akan dibangun pada tahap implementasi selanjutnya. Perancangan pada sistem pengenalan huruf bahasa isyarat berbasis *desktop* terdiri dari beberapa halaman sebagai berikut:

1. Home Page

Halaman ini merupakan halaman yang ditampilkan pertama kali ketika sistem dijalankan. Halaman ini berisi gambar tangan huruf bahasa isyarat startis yang mengacu pada SIBI (Sistem Isyarat Bahasa Indonesia) yang

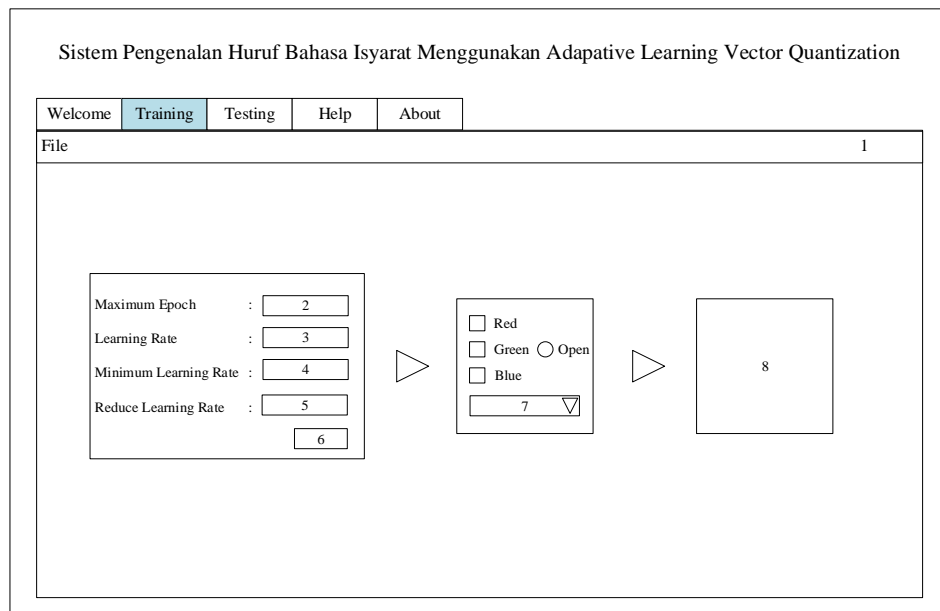
digunakan dalam sistem. Perancangan *home page* ditunjukkan pada gambar 4.13.



Gambar 4.13 Rancangan *Home Page*

2. *Training Page*

Halaman ini merupakan halaman yang menyediakan kebutuhan proses *training* seperti parameter-parameter yang ditunjukkan seperti pada gambar 4.14.



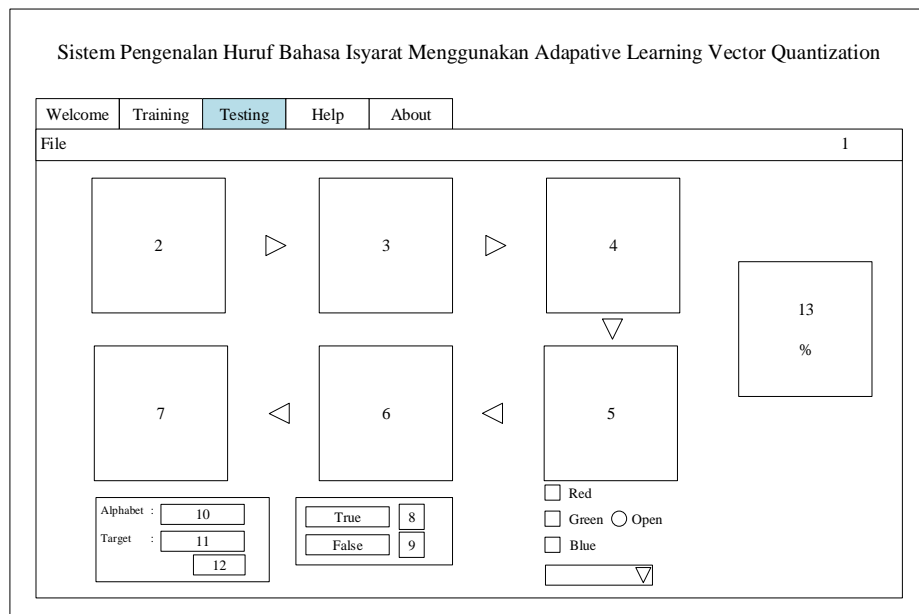
Gambar 4.14 Rancangan *Training Page*

Keterangan gambar 4.14:

- 1) MenuStrip yang memiliki sub menu 'Reset' yang digunakan untuk membersihkan semua *field* yang terisi.
- 2) Textbox untuk menerima masukan nilai maksimal *epoch*.
- 3) Textbox untuk menerima masukan nilai *learning rate*.
- 4) Textbox untuk menerima masukan nilai minimal *learning rate*.
- 5) Textbox untuk menerima masukan nilai reduksi *learning rate*.
- 6) Button untuk memulai proses *training*.
- 7) Groupbox untuk proses *denoising* pada *training*.
- 8) Groupbox untuk menampilkan hasil akhir gambar yang di-*training*.

3. Testing Page

Halaman ini merupakan halaman yang digunakan untuk proses *testing* dan *validation* setelah melakukan proses *training*. Halaman ini menyediakan tempat untuk melakukan proses pengenalan gambar tangan hingga proses pengenalan huruf bahasa isyarat. Perancangan *testing page* ditunjukkan pada gambar 4.15.

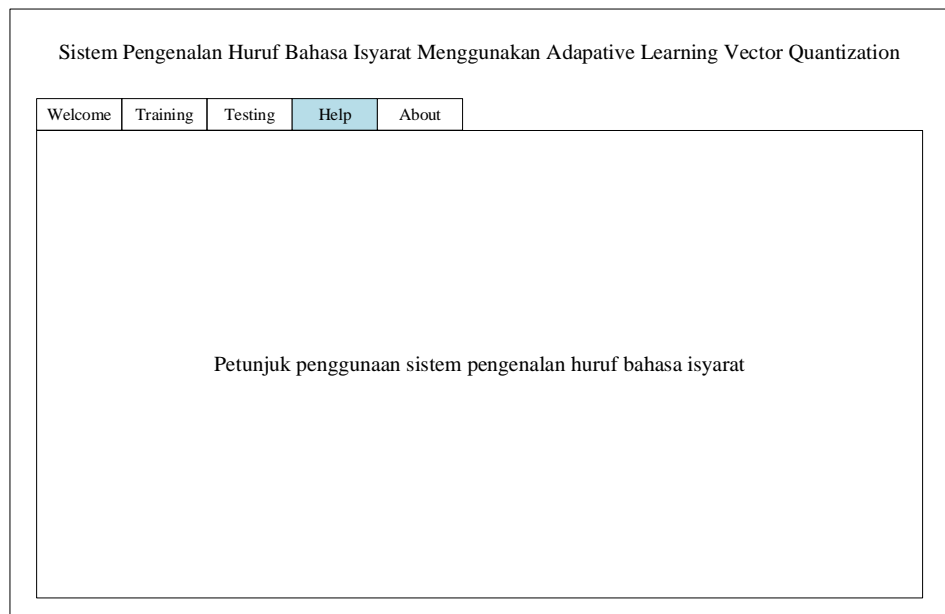


Gambar 4.15 Rancangan *Testing Page*

Keterangan gambar 4.15:

- 1) MenuStrip yang memiliki sub menu 'Open' yang digunakan untuk melakukan pemuatan gambar dari direktori untuk proses *testing*.
- 2) Groupbox untuk menampilkan hasil gambar yang dimuat dari direktori.

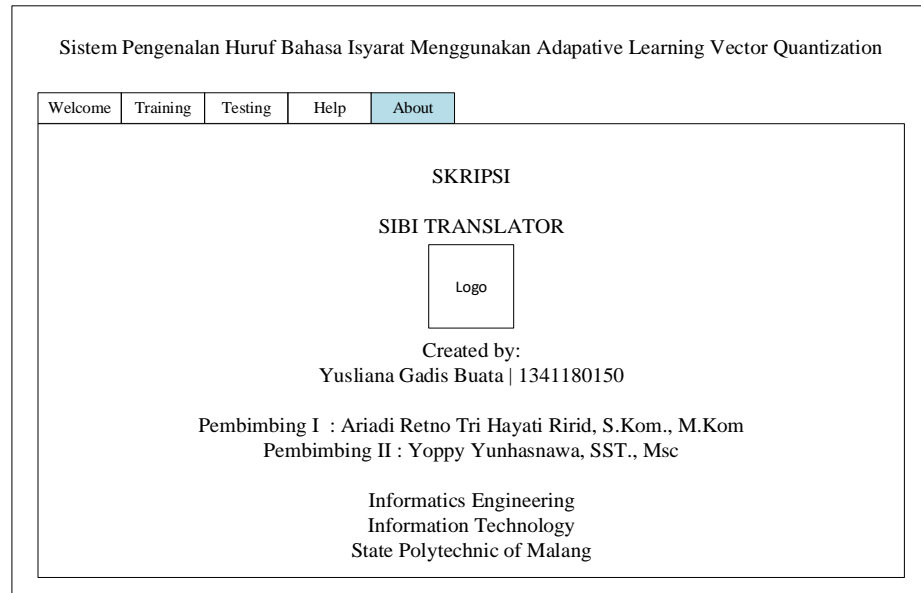
- 3) Groupbox untuk menampilkan hasil proses *skin detection*.
 - 4) Groupbox untuk menampilkan hasil proses *thresholding*.
 - 5) Groupbox untuk menampilkan hasil proses *denoising*.
 - 6) Groupbox untuk menampilkan hasil proses *extraction*.
 - 7) Groupbox untuk menampilkan hasil pengenalan huruf.
 - 8) Button untuk memasukkan status keluaran citra yang benar.
 - 9) Button untuk memasukkan status keluaran citra yang salah.
 - 10) Textbox untuk menerima masukan huruf yang seharusnya.
 - 11) Textbox untuk menerima masukan target yang seharusnya.
 - 12) Button untuk melakukan proses validasi.
 - 13) Groupbox untuk menampilkan hasil akurasi *training* dan *testing*.
4. *Help Page*
- Halaman ini merupakan halaman yang berisi petunjuk penggunaan sistem pengenalan huruf bahasa isyarat yang meliputi petunjuk proses *training* dan *testing*. Perancangan *help page* ditunjukkan pada gambar 4.16.



Gambar 4.16 Rancangan *Help Page*

5. About Page

Halaman ini merupakan halaman yang berisi tentang informasi pengembang sistem. Perancangan *about page* ditunjukkan pada gambar 4.17.



Gambar 4.17 Rancangan *About Page*

4.4 Perancangan Pengujian

Pengujian sistem dirancang untuk mengetahui pengaruh perhitungan akurasi dari sistem pengenalan huruf bahasa isyarat yang terdiri dari:

1. Pengaruh penggunaan ekstraksi ciri Fitur Biner.
2. Pengaruh penggunaan perhitungan jarak *Euclidean* pada algoritma *Adaptive LVQ*.
3. Pengaruh proses validasi (*Adaptive Prototype Addition and Removal*) dan banyaknya jumlah set data *testing* yang digunakan (n set x 24 kelas huruf).

Pada proses pengujian menggunakan 24 *dataset* tiap huruf yang didapatkan dari 8 orang. Jumlah data *training* yang digunakan sebanyak 4 *dataset* tiap huruf dan data *testing* sebanyak 20 *dataset* tiap huruf. Penambahan jumlah *set* data *testing* dilakukan sesuai dengan kelipatan dari jumlah data awal, yaitu 4, 8, 12, 16, dan 20. Nilai parameter yang digunakan yaitu nilai *learning rate* 0.1, minimum *learning rate* 0.001, dan maksimum *epoch* 50. Nilai *learning rate* dan minimum *learning rate* dipilih karena tidak memberikan perubahan nilai bobot yang terlalu signifikan pada algoritma *Adaptive LVQ*. Nilai maksimum *epoch* dipilih karena merupakan

epoch maksimum yang dicapai ketika nilai bobot LVQ telah mencapai kondisi konvergen adalah 43. Perancangan pengujian pada data *testing* dilakukan seperti pada tabel 4.2.

Tabel 4.2 Perancangan Pengujian Data *Testing*

Jumlah Data <i>Training</i> (n set)	Akurasi
96 (4 set)	
192 (8 set)	
288 (12 set)	
384 (16 set)	
480 (20 set)	

BAB V. IMPLEMENTASI

Pada bab ini akan membahas mengenai implementasi sistem pengenalan huruf bahasa isyarat berdasarkan pembahasan pada bab analisis dan perancangan sebelumnya. Pembahasan terdiri dari penjelasan tentang spesifikasi sistem, implementasi algoritma pada program dan implementasi *user interface*.

5.1 Spesifikasi Sistem

Spesifikasi sistem meliputi spesifikasi *hardware* dan *software* yang dibutuhkan dalam implementasi sistem pengenalan huruf bahasa isyarat menggunakan *Adaptive Learning Vector Quantization*.

5.1.1 Spesifikasi Hardware

Hardware yang digunakan dalam implementasi sistem ditunjukkan pada tabel 5.1.

Tabel 5.1 Spesifikasi *Hardware*

Nama Komponen	Spesifikasi
CPU	Intel Celeron N2830
Memori / RAM	2 GB DDR3
Harddisk	500 GB
Tipe Grafis	Intel HD
Monitor	14" WXGA, Resolusi 1366 x 768
Speaker	Dolby Advanced Audio v2
Kamera	Native HD 1.0 MP

5.1.2 Spesifikasi Software

Software yang digunakan dalam implementasi sistem ditunjukkan pada tabel 5.2.

Tabel 5.2 Spesifikasi *Software*

Tools	Nama Software
Sistem Operasi	Windows 10 (64-bit)
Bahasa Pemrograman	Visual Basic.NET
Tools Pemrograman	Visual Studio 2015 dan SQL Server 2014
Library	RedMatter dan Math.NET Numerics

5.2 Implementasi Kode Program

Implementasi kode program menjelaskan implementasi kode dari sistem pengenalan huruf bahasa isyarat yang didasarkan pada alur sistem yang telah

dijelaskan pada bab analisis dan perancangan. Tahapan ini meliputi proses *load image*, *skin detection*, *thresholding*, *denoising*, *extraction* dan *Adaptive LVQ Classification*.

5.2.1 Kode Program *Load Image*

Implementasi *load image* digunakan untuk memuat citra tangan dari direktori ke sistem. Fungsi untuk melakukan *load image* terdapat pada kelas FileManager seperti pada rancangan diagram kelas pada bab analisis dan perancangan. Fungsi untuk melakukan *load data* dapat dilihat pada kode program 5.1.

Kode Program 5.1 Implementasi Fungsi *Load Image*

1	Public Shared Function openFile()
2	Dim imageFile As Image = Nothing
3	Dim openFileDialog As OpenFileDialog = New
4	OpenFileDialog()
5	openDialog.RestoreDirectory = True
6	
7	If openFileDialog.ShowDialog =
8	Windows.Forms.DialogResult.OK Then
9	imageFile =
10	Image.FromFile(openDialog.FileName)
11	End If
12	
13	Return imageFile
14	End Function

Pada kode program di atas, parameter yang digunakan adalah variabel *imageFile* bertipe data *Image* dan *openDialog* bertipe data *OpenFileDialog* seperti pada baris 2-4. Kemudian pada baris 7-11 dilakukan pemuatan citra dari direktori.

5.2.2 Kode Program *Skin Detection*

Proses *Skin Detection* digunakan untuk membedakan *foreground* dan *background* dari citra, dimana nilai warna piksel yang bukan merupakan kulit diubah menjadi hitam. Pada penelitian ini, untuk melakukan proses *skin detection* menggunakan *library* *RedMatter* yang menyediakan fungsi *skin detection* seperti pada kode program 5.2.

Kode Program 5.2 Implementasi Fungsi *Skin Detection*

1	Public Shared Function skinDetection(img As Image)
2	Dim detector As BaseDetector = New
3	BaseDetector(img, New SimpleSkinDetector4)
4	Dim outputImg As Image =
5	detector.SkinDetectionImage
6	
7	Return outputImg
8	End Function

Pada kode program di atas, menggunakan kelas *BaseDetector* yang disediakan library *RedMatter* yang memiliki parameter masukan *image* dan jenis *skin detection* yang digunakan (baris 2-3) sehingga menghasilkan warna kulit dan bukan kulit pada citra.

5.2.3 Kode Program *Thresholding*

Proses *thresholding* dilakukan untuk mnegubah citra menjadi dua warna saja yaitu hitam dan putih. Setelah dilakukan *skin detection*, kemudian citra daerah warna kulit diubah warnanya menjadi hitam (nilai piksel = 0) sebagai *foreground* dan *background*-nya diubah menjadi warna putih (nilai piksel = 255). Fungsi *thresholding* dapat dilihat pada kode program 5.3.

Kode Program 5.3 Implementasi Fungsi *Thresholding*

1	Public Shared Function threholding(bmp As Bitmap)
2	
3	Dim i, j As Integer
4	Dim red, green, blue, sum As Integer
5	Dim binaryValue As Integer
6	Dim getWidth, getHeight As Integer
7	
8	getWidth = bmp.Width
9	getHeight = bmp.Height
10	
11	For i = 0 To getWidth - 1
12	For j = 0 To getHeight - 1
13	red = bmp.GetPixel(i, j).R
14	green = bmp.GetPixel(i, j).G
15	blue = bmp.GetPixel(i, j).B
16	
17	sum = red + green + blue
18	
19	If (sum = 0) Then
20	binaryValue = 255
21	Else
22	binaryValue = 0
23	End If
24	
25	bmp.SetPixel(i, j,
26	Color.FromArgb(binaryValue, binaryValue, binaryValue))
27	Next
28	Next
29	Return bmp
30	End Function

Pada kode program di atas, untuk mendapatkan nilai RGB tiap piksel citra terdapat pada baris 13-15. Kemudian dilakukan pengecekan apakah piksel tersebut merupakan *background* yang bernilai 0. Jika iya, maka *background* diubah menjadi putih (nilai piksel = 255). Tetapi jika tidak, maka piksel tersebut merupakan

foreground yang berwarna putih (nilai piksel = 255) dan diubah menjadi hitam (nilai piksel = 0) yang dijelaskan pada baris 19-23.

5.2.4 Kode Program *Denoising*

Proses *denoising* digunakan untuk mengurangi atau mereduksi *noise* pada citra. *Noise* di sini yaitu warna yang bukan kulit menjadi objek dan warna kulit menjadi objek. Fungsi *denoising* dapat dilihat pada kode program 5.4.

Kode Program 5.4 Implementasi Fungsi *Denoising*

1	Public Shared Function denoising(imgThreshold As Bitmap)
2	Dim imgResult As Bitmap =
3	ImageErosionDilation.Denoising.CopyToSquareCanvas(imgThres
4	hold, imgThreshold.Width)
5	Return imgResult
6	End Function

Pada kode program di atas, proses *denoising* menggunakan metode erosi dan dilasi seperti pada baris 3-4. Fungsi *denoising()* menggunakan masukan parameter bertipe *Bitmap*. Proses erosi dan dilasi menggunakan *structuring element* (SE) berukuran 3 x 3.

5.2.5 Kode Program *Extraction*

Proses *extraction* digunakan untuk melakukan ekstraksi fitur citra dimana dilakukan saat semua tahapan *preprocessing* telah dijalankan. Ekstraksi fitur yang digunakan yaitu fitur biner yaitu mengambil nilai dari tiap piksel citra dan menyimpannya ke dalam bentuk vektor *array* 1 dimensi, dengan panjang vektor *array* adalah 2500. Sebelum diubah menjadi vektor, citra dikonversi terlebih dahulu menjadi matriks yang bertipe data *double* seperti pada fungsi konversi matriks 5.5.

Kode Program 5.5 Implementasi Fungsi *Extraction* (Konversi Matriks)

1	Public Shared Function convertToDoubleArray(img As Bitmap)
2	Dim doubleArray(img.Width - 1, img.Height - 1) As Double
3	Dim bmp As Bitmap = img
4	For y As Integer = 0 To bmp.Height - 1
5	For x As Integer = 0 To bmp.Width - 1
6	Dim clr As Color = bmp.GetPixel(x, y)
7	Dim d As Double = 0
8	If clr.R <= 0 Then
9	d = 1
10	Else
11	d = 0
12	End If
13	doubleArray(x, y) = d
14	Next
15	Next
16	Return doubleArray
17	End Function

Nilai vektor tiap elemen diisi dengan nilai antara 0 atau 1. Jika piksel berwarna hitam atau sebagai *foreground* dengan nilai piksel = 0, maka nilai biner untuk elemen vektor diubah menjadi 1. Sedangkan untuk *background* nilai binernya adalah 0 seperti pada baris 9-13. Kemudian untuk fungsi ekstraksi ciri fitur biner dapat dilihat pada kode program 5.6.

Kode Program 5.6 Implementasi Fungsi *Extraction* (Fitur Biner)

1	Public Shared Function
2	convertToInputVector(inputMatriks As Matrix(Of Double))
3	Dim rowTotals As Integer = inputMatriks.RowCount
4	Dim columnTotals As Integer =
5	inputMatriks.ColumnCount
6	Dim arrLength As Integer = rowTotals *
7	columnTotals
8	Dim data = New List(Of Integer)
9	Dim index As Integer = 0
10	
11	For i As Integer = 0 To rowTotals - 1
12	For j As Integer = 0 To columnTotals - 1
13	data.Add(inputMatriks(i, j))
14	index += 1
15	Next
16	Next
17	Dim arrRawData() As Integer = New
18	Integer(arrLength + 1) {}
19	arrRawData = data.ToArray()
20	Dim rawData As New
21	System.Collections.ArrayList(arrRawData)
22	Dim iv As InputVector = New InputVector(rawData)
23	Return iv
24	End Function

Keluaran dari proses ini berupa vektor *array* 1 dimensi dengan panjang vektor ukuran lebar dikali tinggi citra, yaitu 2500 (50 x 50 piksel) yang didefinisikan pada baris 6-7.

5.2.6 Kode Program *Adaptive LVQ Classification*

Setelah ciri didapatkan, kemudian dilakukan proses klasifikasi yang menggunakan algoritma *Adaptive Learning Vector Quantization* dimana merupakan algoritma pembelajaran, dengan mencari jarak terdekat untuk menentukan suatu kelas. Proses klasifikasi ini terdiri dari proses *training*, *testing* dan *validation*. Implementasi fungsi training algoritma *Adaptive LVQ* dapat dilihat pada kode program 5.7.

Kode Program 5.7 Implementasi Fungsi *Training Adaptive LVQ*

1	Public Function training(maxEpoch As Integer, alpha As
2	Double, minAlpha As Double, redAlpha As Double)
3	

4	Dim arrMatrixW As ArrayList = New ArrayList
5	Dim arrMatrixX As ArrayList = New ArrayList
6	
7	For i As Integer = 0 To classCount - 1
8	arrMatrixW.Add(matrices(i))
9	Next
10	For j As Integer = classCount To matrices.Count -
11	1
12	arrMatrixX.Add(matrices(j))
13	Next
14	
15	Dim arrInputVectorW As ArrayList = New
16	ArrayList(arrMatrixW.Count)
17	For i As Integer = 0 To classCount - 1
18	Dim matrixW As Matrix(Of Double) =
19	arrMatrixW(i)
20	Dim ivW As InputVector =
21	convertToInputVector(matrixW)
22	arrInputVectorW.Add(ivW)
23	Next
24	
25	Dim arrInputVectorX As ArrayList = New
26	ArrayList(arrMatrixX.Count)
27	For i As Integer = 0 To arrMatrixX.Count - 1
28	Dim matrixX As Matrix(Of Double) =
29	arrMatrixX(i)
30	Dim ivX As InputVector =
31	convertToInputVector(matrixX)
32	arrInputVectorX.Add(ivX)
33	Next
34	
35	Dim arrIvX As ArrayList = arrInputVectorX
35	Dim arrIvW As ArrayList = arrInputVectorW
37	Dim epoch As Integer = 0
38	
39	For i As Integer = 0 To classCount - 1
40	Me.finalWeightResults.Add(arrInputVectorW(i))
41	Next
42	
43	Do While (epoch < maxEpoch)
44	If (alpha < minAlpha) Then
45	Exit Do
46	End If
47	
48	For ix As Integer = 0 To arrIvX.Count - 1
49	Dim ivx As InputVector = arrIvX(ix)
50	Dim distances As ArrayList = New
51	ArrayList(arrIvW.Count)
52	For w As Integer = 0 To arrIvW.Count - 1
53	Dim ivw As InputVector = arrIvW(w)
54	Dim d As Double = countDistance(ivx,
55	ivw)
56	distances.Insert(w, d)
57	Next
58	Dim minDistanceClass =
59	minDistanceIndexArrayList(distances)
60	Dim expectedTarget = Me.targets(ix +
61	arrIvW.Count)
62	Dim updatedWeight As InputVector

63	Dim oldW = arrIvW(minDistanceClass)
64	
65	If (minDistanceClass = expectedTarget)
66	Then
67	updatedWeight = createNewWeight(oldW,
68	ivx, alpha, True)
69	Else
70	updatedWeight = createNewWeight(oldW,
71	ivx, alpha, False)
72	End If
73	Me.finalWeightResults(minDistanceClass) =
74	updatedWeight
75	Next
76	
77	alpha = reduceAlpha(alpha, redAlpha)
78	epoch += 1
79	Loop
80	
81	Return finalWeightResults
82	End Function

Kode program di atas merupakan fungsi *training* yang dijalankan untuk proses *training* klasifikasi. Pada proses ini membutuhkan masukan parameter berupa maksimal *epoch* untuk menentukan batas perulangan yang dilakukan. Selain itu, perulangan akan berhenti apabila nilai *learning rate* telah mencapai nilai yang sangat kecil, dalam implementasi menggunakan variabel *minAlpha* yang didapat dari masukan sebagai batas minimum *learning rate* yang ditunjukkan pada baris 43-44. Setiap perulangan akan melakukan perhitungan jarak untuk mendapatkan jarak kelas terdekat dari vektor masukan terhadap masing-masing keluaran (pada baris 54-59). Jika jarak kelas terdekat yang didapatkan sama dengan nilai targetnya, maka akan dilakukan penambahan bobot dari nilai sebelumnya. Tetapi jika tidak, maka nilai bobot akan dikurangi dari nilai bobot sebelumnya (pada baris 65-75). Langkah terakhir dari proses *training* adalah melakukan pengurangan nilai *learning rate* berdasarkan masukan parameter reduksi *learning rate* (pada baris 77). Maka hasil yang didapatkan dari proses *training* ini berupa vektor bobot yang telah diperbaharui pada setiap perulangan. Sedangkan untuk implementasi fungsi *testing* algoritma *Adaptive LVQ* dapat dilihat pada kode program 5.8.

Kode Program 5.8 Implementasi Fungsi *Testing Adaptive LVQ*

1	Public Shared Function testingWithNewWeight(inputMatrix As
2	Matrix(Of Double), newWeightFromDb As String, classCount
3	As Integer)
4	Dim newWeights As ArrayList =
5	Home.deserializeWeights(newWeightFromDb)
6	Dim ivX As InputVector =
7	convertToInputVector(inputMatrix)
8	

9	Dim distances As ArrayList = New
10	ArrayList(newWeights.Count)
11	For w As Integer = 0 To newWeights.Count - 1
12	Dim ivw As InputVector = newWeights(w)
13	Dim d As Double = countDistance(ivX, ivw)
14	distances.Insert(w, d)
15	Dim vWi As Double = countVector(ivw)
16	Next
17	Dim minDistanceClass =
18	minDistanceIndexArrayList(distances)
19	
20	Dim vectorX As Double = countVector(ivX)
21	
22	Console.WriteLine("Included Class: " +
23	minDistanceClass.ToString())
24	Console.ReadLine()
25	Return minDistanceClass
26	End Function

Proses yang ada pada *testing* dilakukan pula pencarian jarak kelas terdekat dari vektor masukan terhadap kelas keluaran (pada baris 13-18). Perbedaannya pada proses *testing* hanya dilakukan 1 kali perhitungan jarak tanpa adanya perulangan. Jika jarak dari kelas terdekat telah didapatkan, maka nilai kelas inilah yang akan dikembalikan sebagai hasil dari proses *testing* (pada baris 25).

Proses validasi dilakukan setelah proses *testing* dijalankan. Jika hasil keluaran huruf bahasa isyarat belum sesuai dengan target atau keluaran yang diharapkan maka dilakukan proses *insert* data tersebut ke *database* yang akan dilatih pada proses *training* selanjutnya. Fungsi *validation* algoritma *Adaptive LVQ* dapat dilihat pada kode program 5.9.

Kode Program 5.9 Implementasi Fungsi *Validation Adaptive LVQ*

1	Private Sub btnValid_Click(sender As Object, e As
2	EventArgs) Handles btnValid.Click
3	Dim FileInfo As New FileInfo(Me.lastFileName)
4	Dim onlyFileName As String = FileInfo.Name
5	Dim outputExpected As String =
6	ConnectionDatabase.insertOutputExpected(onlyFileName,
7	tbAlphabet, tbTarget)
8	MessageBox.Show("Output expected inserted!")
9	End Sub

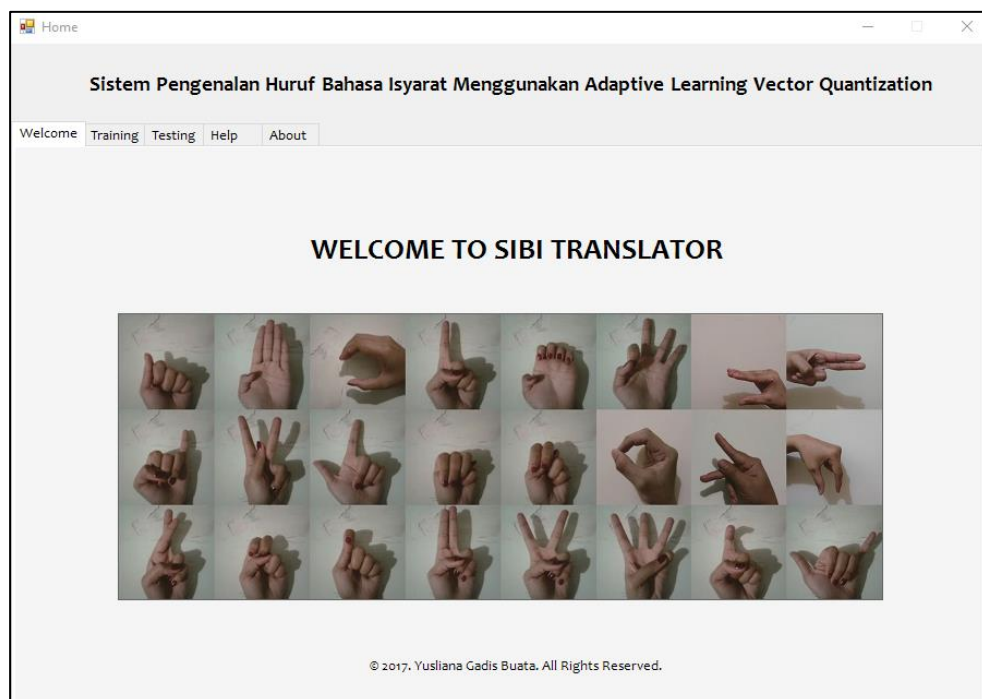
Elemen-elemen yang dimasukkan ke dalam *database* saat keluaran tidak sesuai dengan target antara lain nama *image file*, huruf, dan target yang nantinya akan dilakukan *training* selanjutnya sehingga menghasilkan akurasi yang lebih tinggi dari sebelumnya (pada baris 3-8).

5.3 Implementasi *User Interface*

Implementasi *user interface* yang diterapkan berdasarkan perancangan antarmuka sistem pada bab analisis dan perancangan. Berikut ini merupakan *user interface* yang diimplementasikan dalam sistem pengenalan huruf bahasa isyarat.

5.3.1 UI *Home Page*

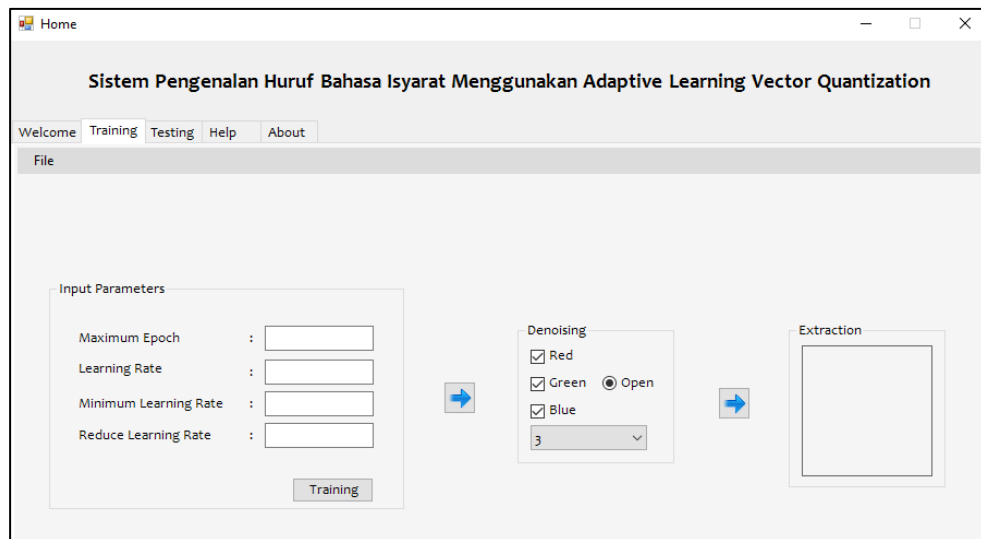
Home page merupakan halaman utama yang pertama kali tampil pada saat sistem. Pada halaman ini, berisi mengenai gambar petunjuk huruf bahasa isyarat statis yang mengacu pada SIBI (Sistem Isyarat Bahasa Indonesia) yaitu A-Z kecuali J dan Z. Berikut gambar 5.1 merupakan tampilan *home page*.



Gambar 5.1 Implementasi *Home Page*

5.3.2 UI *Training Page*

Training page merupakan halaman sistem yang berisi proses pelatihan data menggunakan algoritma *Adaptive LVQ*. Di dalam halaman ini membutuhkan masukan parameter dan masukan vektor yang didapatkan dari proses *load* citra *training* sehingga menghasilkan vektor bobot yang akan digunakan pada proses *testing*. Tampilan *training page* ditunjukkan pada gambar 5.2 berikut.

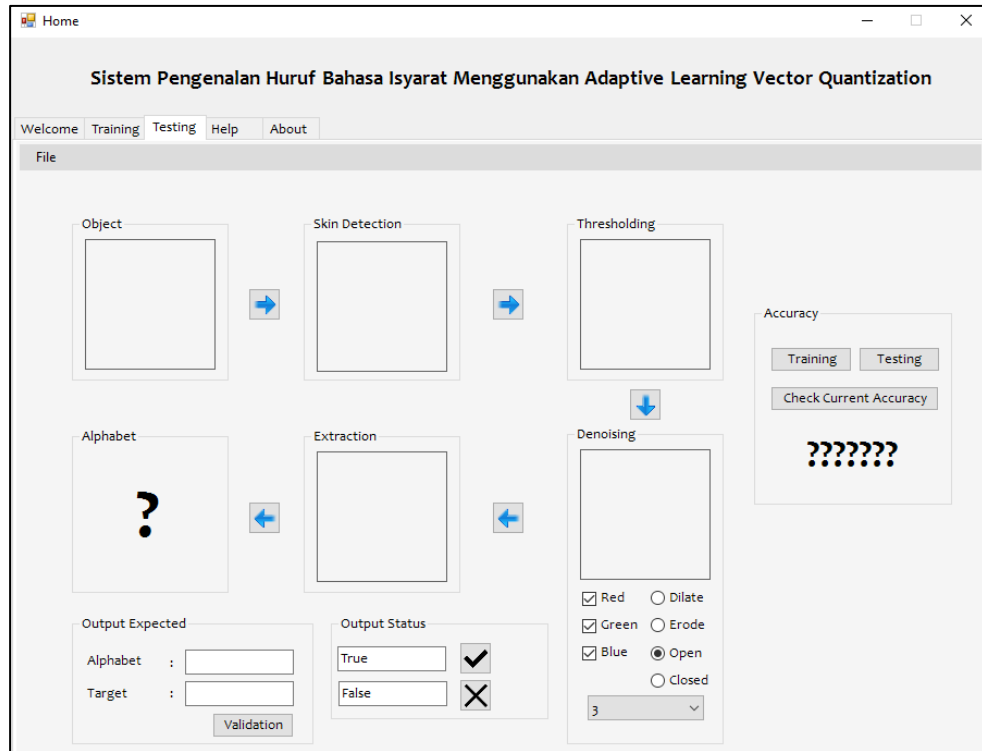


Gambar 5.2 Implementasi *Training Page*

5.3.3 UI *Testing Page*

Testing page merupakan halaman sistem yang digunakan untuk proses pengenalan atau klasifikasi citra. Pada halaman ini terdapat beberapa komponen *groupbox* yang diantaranya *groupbox* objek, deteksi kulit, binerisasi, reduksi *noise*, ekstraksi fitur, keluaran huruf, dan hasil akurasi proses *training* dan *testing*. Pada halaman ini juga digunakan untuk proses *validation* yaitu dimana *adaptive prototype addition and removal* berperan. Proses ini dilakukan dengan cara *user* memasukkan keluaran yang seharusnya (huruf dan targetnya) apabila keluaran huruf tidak sesuai dengan yang semestinya pada *groupbox* keluaran seharusnya.

Hasil akurasi dari proses *training* dan *testing* juga ditampilkan pada halaman ini. Pengecekan akurasi *training* dilakukan dengan melakukan klik terlebih dahulu pada *button Training*. Sedangkan pengecekan akurasi *testing* dilakukan dengan melakukan klik terlebih dahulu pada *button Testing*. Berikut *testing page* dapat dilihat pada gambar 5.3.



Gambar 5.3 Implementasi *Testing Page*

5.3.4 UI *Help Page*

Help page merupakan halaman sistem yang berisi mengenai petunjuk penggunaan sistem mulai dari proses *load* citra untuk *training* hingga proses *load* citra untuk proses *testing* dan *validation*. *Help page* dapat dilihat pada gambar 5.4 berikut.



Gambar 5.4 Implementasi *Help Page*

5.3.5 UI *About Page*

Pada *about page* berisi informasi mengenai identitas pengembang sistem seperti pada gambar 5.5 berikut.



Gambar 5.5 Implementasi *About Page*

BAB VI. PENGUJIAN DAN PEMBAHASAN

Pada bab pengujian dan pembahasan membahas mengenai pengujian akurasi dan hasil pembahasan analisis penggunaan sistem pengenalan huruf bahasa isyarat menggunakan algoritma *Adaptive LVQ*.

6.1 Pengujian

Proses pengujian sistem dilakukan dengan cara pengujian akurasi sistem saat proses *training* dan *testing* dilakukan. Pengujian akurasi ini bertujuan untuk mengetahui seberapa besar persentase ketepatan penggunaan algoritma *Adaptive LVQ* untuk pengenalan huruf bahasa isyarat.

Pada proses klasifikasi menggunakan *dataset* sebanyak 24 data citra tiap huruf yang berasal dari pengambilan citra tangan pada 8 orang yang diambil di dalam ruang tertutup dengan cahaya lampu pada malam hari dan pada jarak 20 cm antara objek tangan dengan kamera. Pembagian komposisi data yang digunakan sebanyak 4 data citra tiap huruf sebagai data *training* dan 20 data citra lainnya sebagai data *testing*. Proses *testing* dilakukan secara bertahap dengan mencari hasil akurasi terbaik pada jumlah data yang digunakan. Data yang digunakan untuk proses *testing* sebanyak 20 citra tiap hurufnya yaitu 480 citra tangan (20x24 kelas huruf) yang diambil dari semua data *testing* yang disimpan di dalam direktori.

6.1.1 Pengujian pada Data *Training*

Pengujian yang pertama dilakukan adalah untuk mencari hasil akurasi dari proses *training* yang dilakukan. Proses ini menggunakan 4 citra tiap hurufnya yaitu 96 citra tangan (4x24 kelas huruf). Persentase akurasi yang dihasilkan dari proses *training* adalah 94,8 % yang secara rinci dapat dilihat pada bagian lampiran.

6.1.2 Pengujian pada Data *Testing*

Pengujian selanjutnya yang dilakukan adalah untuk mencari hasil akurasi terbaik pada pengaruh jumlah data *testing* yang berbeda-beda. Jumlah data *testing* yang digunakan sebanyak n set atau banyaknya data citra tiap hurufnya dalam 1 set, yang terdiri dari 4, 8, 12, 16, dan 20 set data citra untuk masing-masing huruf. Pengujian dilakukan dengan menggunakan *parameter* yang sama yaitu, nilai *learning rate* 0.1, minimum *learning rate* 0.001, dan maksimum *epoch* 50. Hasil

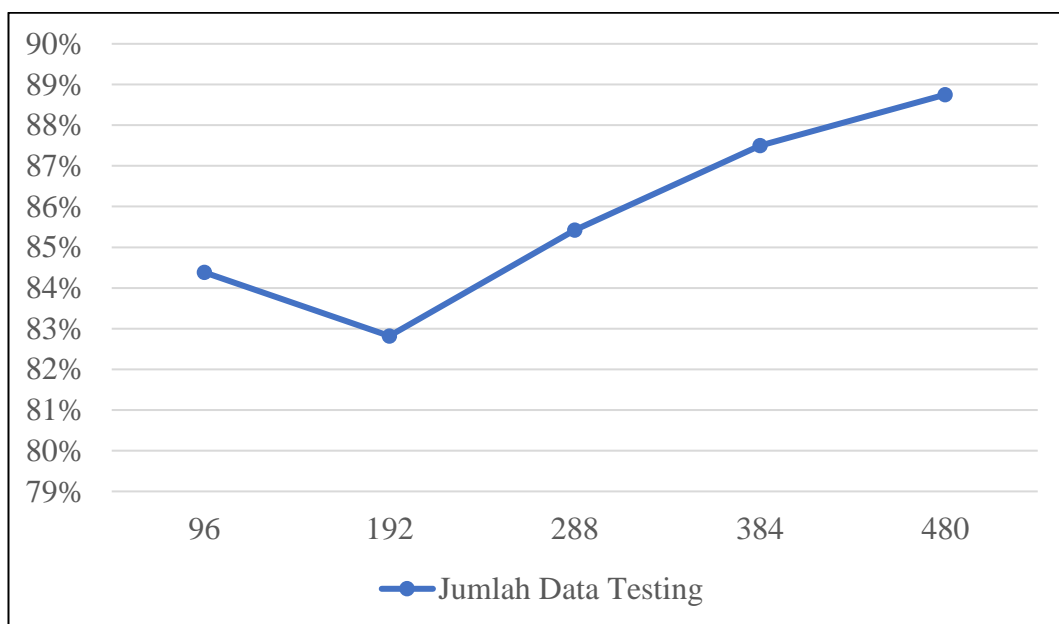
pengujian yang dilakukan, didapatkan untuk masing-masing nilai akurasi proses *testing* ditunjukkan pada tabel 6.1. Hasil secara rinci dari setiap pengujian pada tiap hurufnya dapat dilihat pada bagian lampiran.

Tabel 6.1 Hasil Pengujian terhadap Pengaruh Jumlah *Dataset Testing*

Jumlah Data <i>Testing</i> (n set)	Akurasi (%)
96 (4 set)	84,38 %
192 (8 set)	82,82 %
288 (12 set)	85,42 %
384 (16 set)	87,50 %
480 (20 set)	88,75 %

6.2 Analisis

Proses analisis dilakukan untuk mendapatkan kesimpulan dari hasil pengujian penerapan algoritma *Adaptive Learning Vector Quantization* dalam pengenalan huruf bahasa isyarat. Hasil analisis didapatkan dari setiap pengujian yang telah dilakukan. Berdasarkan hasil pengujian pada tabel 6.1 didapatkan hasil akurasi yang berbeda-beda. Hasil pengujian tersebut disajikan dalam bentuk grafik pada gambar 6.1.



Gambar 6.1 Grafik Hasil Akurasi Data *Testing*

Pada grafik 6.1 menunjukkan bahwa hasil akurasi terbaik ada pada saat menggunakan jumlah data *testing* sebanyak 480. Pengaruh dari proses validasi atau proses *Adaptive Prototype Addition and Removal* memberikan kenaikan hasil akurasi yang stabil. Pada penggunaan jumlah data 96 (4 set) menuju jumlah data

192 (8 set) mengalami penurunan disebabkan oleh adanya data *testing* yang memiliki bentuk variasi berbeda seperti bentuk tangan agak miring, cahaya yang kurang sehingga menyebabkan kualitas citra kurang baik. Sedangkan pada penggunaan data mulai dari 288 (12 set) sampai 480 (20 set) terus mengalami kenaikan hasil akurasi yang baik yang dikarenakan adanya proses validasi setelah proses *testing* setiap hurufnya.

Berdasarkan beberapa hasil pengujian didapatkan beberapa huruf yang sering salah dikenali oleh sistem. Huruf-huruf tersebut antara lain huruf H, K, O, dan R. Hal ini disebabkan karena faktor pencahayaan yang kurang maksimal sehingga mengakibatkan hasil proses *skin detection* yang kurang baik. Warna kulit yang seharusnya terdeteksi menjadi tidak terdeteksi sehingga menjadi warna latar belakang, begitu juga sebaliknya. Selain itu bentuk dan posisi citra tangan yang berbeda juga menyebabkan kesalahan dalam proses pengenalan.

BAB VII. PENUTUP

Pada bab ini akan dipaparkan kesimpulan dan saran yang dapat diambil berdasarkan pengujian yang telah dilakukan terhadap sistem pengenalan huruf bahasa isyarat dengan algoritma *Adaptive Learning Vector Quantization*.

7.1 Kesimpulan

Berdasarkan hasil dari pengujian yang dilakukan dapat diambil beberapa kesimpulan sebagai berikut.

1. Algoritma *Adaptive Learning Vector Quantization* dapat diterapkan untuk melakukan pengenalan huruf bahasa isyarat dengan melakukan beberapa tahapan proses mulai dari *preprocessing* yang terdiri dari *skin detection*, *thresholding*, dan *denoising* kemudian proses ekstraksi ciri dengan fitur biner serta proses klasifikasi citra tangan hingga mengeluarkan *output* huruf bahasa isyarat.
2. Hasil pengujian akurasi algoritma *Adaptive Learning Vector Quantization* pada proses *training* data citra tangan huruf bahasa isyarat terhadap pengaruh penggunaan ekstraksi ciri dengan fitur biner dan penggunaan perhitungan jarak *Euclidean* didapatkan hasil akurasi sebesar 94,8%.
3. Hasil pengujian akurasi algoritma *Adaptive Learning Vector Quantization* pada proses *testing* data citra tangan huruf bahasa isyarat terhadap pengaruh proses validasi (*Adaptive Prototype Addition and Removal*) dan jumlah data *testing* didapatkan hasil akurasi terbaik ketika jumlah data *testing* sebanyak 480 (20 set) dengan persentase sebesar 88,75%.
4. Pengenalan huruf bahasa isyarat pada sistem bergantung pada teknik pengambilan gambar antara lain merk dan tipe kamera, resolusi kamera, resolusi citra yang digunakan, bentuk dan posisi tangan atau jarak pengambilan citra serta faktor pencahayaan.

7.2 Saran

Saran yang ditujukan untuk penelitian selanjutnya antara lain sebagai berikut.

1. Teknik pengambilan citra tangan huruf bahasa isyarat harus bersifat konsisten. Bentuk dan posisi tangan harus stabil, tidak adanya perubahan pada rotasi dan translasi pada citra tangan yang diambil. Selain itu faktor pencahayaan juga harus stabil atau homogen sehingga menghindari perbedaan hasil atau kesalahan pada saat dilakukannya proses *skin detection* pada citra.
2. Pengembangan pada metode *thresholding* yang digunakan untuk mengantisipasi adanya faktor pencahayaan yang berbeda saat pengambilan citra.
3. Penambahan metode deteksi bentuk citra tangan atau proses segmentasi untuk meminimalisir kesalahan klasifikasi citra.
4. Penelitian selanjutnya dapat mengenali huruf isyarat dinamis, seperti huruf J dan Z.

























DAFTAR PUSTAKA





























- [1] Tula, Jerry J. 2015. “*Pelayanan Penyandang Disabilitas dalam Menggunakan Berbagai Sarana Akseibilitas*”. <http://rehsos.kemsos.go.id/modules.php?name=News&file=article&sid=1890>. Diakses pada 6 April 2017.
- [2] Kementrian Sosial RI. 2012. “*Kementrian Sosial dalam Angka Pembangunan Kesejahteraan Sosial*”. Badan Pendidikan dan Penelitian Kesejahteraan Sosial, Pusat Data dan Informasi Kesejahteraan Sosial. Jakarta.
- [3] Rakhman, Juniar P., et al. 2010. “*Transalasi Bahasa Isyarat*”. Jurnal Library ITS, 2, Vol. 7. Jurusan Teknik Informatika, PENS-ITS Surabaya. Surabaya.
- [4] Mardiyani, Atik, et al. 2012. “*Pengenalan Bahasa Isyarat Menggunakan Metode PCA dan Haar Like Feature*”. Paper and Presentation of Electrical Engineering, RSE 006.42 Mar p. Jurusan Teknik Elektro, ITS Surabaya. Surabaya.
- [5] Suderajad, Imam, et al. 2015. “*Aplikasi Pengenalan Citra Chord Gitar Menggunakan Learning Vector Quantization (LVQ)*”. SEMANTIK. Jurusan Teknik Elektro, Universitas Panca Marga Probolinggo. Probolinggo.
- [6] Grbovic, Mihajlo, et al. 2009. “*Learning Vector Quantization with Adaptive Prototype Addition and Removal*”. Department of Computer and Information Sciences, Temple University, Philadelphia.
- [7] Utono, Dany A.B. “*Desain Bahasa Gambar Untuk Anak Tuna Rungu*”. Jurnal library. Jurusan Desain Produk Industri, FTSP ITS.Surabaya. Surabaya.
- [8] Wulandari, Helga. 2012. “*Sistem Isyarat Bahasa Indonesia*”. <http://www.slideshare.net/arieplb/sistem-isyarat-bahasa-indonesia-11079733>. Diakses pada 7 Januari 2017.
- [9] Wicaksono, Budi. 2015. “*Eksistensi Bahasa Isyarat Indonesia (BISINDO)*”. <http://jurnal.selasar.com/budaya/eksistensi-bahasa-isyarat-indonesia-bisindo>. Diakses pada 27 Januari 2017.
- [10] Manurung, Kurniawan Kumala Maringan, et al. 2010. “*Berkenalan dengan Bahasa Isyarat Indonesia (BISINDO)*”. DPD GERKATIN DKI Jakarta, Jakarta.
- [11] Zhou, H., et al. 2010. “*Digital Image Processing: Part I*”. Ventus Publishing Aps.
- [12] Sutoyo, T., et al. 2009. “*Teori Pengolahan Citra Digital*”. Yogyakarta: ANDI.
- [13] Putra, Darma. 2010. “*Pengolahan Citra Digital*”. Yogyakarta: ANDI.
- [14] Rizanti, Hanifa V., et al. 2013. “*Pengenalan Citra Alphabet Berdasarkan Parameter Momen Invarian dengan Metode Case-Based Reasoning*”. Skripsi. Malang: Universitas Brawijaya.
- [15] Elgammal, Ahmed., et al. “*Skin Detection - A Short Tutorial*”. <https://www.cs.rutgers.edu/~elgammal/pub/skin.pdf>. Diakses pada 18 April 2017.





























- [16] Kovac, J., Peer, P., Solina, F. 2003. “*Human Skin Colour Clustering for Face Detection*”. EUROCON 2003. University of Ljubljana. Ljubljana, Slovenia.
- [17] Kusumadewi, Sri. 2003. “*Artificial Intelligence (Teknik dan Aplikasinya)*”. Yogyakarta: GRAHA ILMU.
- [18] Barsuk, Maxim. 2009. “*Image Processing: Skin Detection, Some Filters and EXIF Tags*”. <https://www.codeproject.com/Articles/38176/Image-Processing-Skin-Detection-Some-Filters-and-E>. Diakses pada 2 Maret 2017.
- [19] Esterhuizen, Dewald. 2013. “*Image Erosion and Dilation*”. <https://softwarebydefault.com/2013/05/19/image-erosion-dilation/>. Diakses pada 23 Februari 2017.

















LAMPIRAN

Lampiran 1. Hasil *Training* Sistem

No.	Data set ke-				Huruf	Akurasi (%)
	1	2	3	4		
1.	A1.jpg	A2.jpg	A3.jpg	A4.jpg	A	100 %
						
	Status: Benar	Benar	Benar	Benar		
2.	B1.jpg	B2.jpg	B3.jpg	B4.jpg	B	100 %
						
	Status: Benar	Benar	Benar	Benar		
3.	C1.jpg	C2.jpg	C3.jpg	C4.jpg	C	100 %
						
	Status: Benar	Benar	Benar	Benar		
4.	D1.jpg	D2.jpg	D3.jpg	D4.jpg	D	100 %
						
	Status: Benar	Benar	Benar	Benar		
5.	E1.jpg	E2.jpg	E3.jpg	E4.jpg	E	100 %
						
	Status: Benar	Benar	Benar	Benar		
6.	F1.jpg	F2.jpg	F3.jpg	F4.jpg	F	100 %
						
	Status: Benar	Benar	Benar	Benar		

7.	G1.jpg	G2.jpg	G3.jpg	G4.jpg	G	100 %
						
Status:	Benar	Benar	Benar	Benar		
8.	H1.jpg	H2.jpg	H3.jpg	H4.jpg	H	50 %
						
Status:	Benar	Benar	Salah	Salah		
9.	I1.jpg	I2.jpg	I3.jpg	I4.jpg	I	100 %
						
Status:	Benar	Benar	Benar	Benar		
10.	K1.jpg	K2.jpg	K3.jpg	K4.jpg	K	25 %
						
Status:	Benar	Salah	Salah	Salah		
11.	L1.jpg	L2.jpg	L3.jpg	L4.jpg	L	100 %
						
Status:	Benar	Benar	Benar	Benar		
12.	M1.jpg	M2.jpg	M3.jpg	M4.jpg	M	100 %
						
Status:	Benar	Benar	Benar	Benar		
13.	N1.jpg	N2.jpg	N3.jpg	N4.jpg	N	100 %
						
Status:	Benar	Benar	Benar	Benar		
14.	O1.jpg	O2.jpg	O3.jpg	O4.jpg		

					O	100 %
Status:	Benar	Benar	Benar	Benar		
	P1.jpg	P2.jpg	P3.jpg	P4.jpg		
15.					P	100 %
Status:	Benar	Benar	Benar	Benar		
	Q1.jpg	Q2.jpg	Q3.jpg	Q4.jpg		
16.					Q	100 %
Status:	Benar	Benar	Benar	Benar		
	R1.jpg	R2.jpg	R3.jpg	R4.jpg		
17.					R	100 %
Status:	Benar	Benar	Benar	Benar		
	S1.jpg	S2.jpg	S3.jpg	S4.jpg		
18.					S	100 %
Status:	Benar	Benar	Benar	Benar		
	T1.jpg	T2.jpg	T3.jpg	T4.jpg		
19.					T	100 %
Status:	Benar	Benar	Benar	Benar		
	U1.jpg	U2.jpg	U3.jpg	U4.jpg		
20.					U	100 %
Status:	Benar	Benar	Benar	Benar		
21.	V1.jpg	V2.jpg	V3.jpg	V4.jpg		

					V	100 %
Status:	Benar	Benar	Benar	Benar		
	W1.jpg	W2.jpg	W3.jpg	W4.jpg		
22.					W	100 %
Status:	Benar	Benar	Benar	Benar		
	X1.jpg	X2.jpg	X3.jpg	X4.jpg		
23.					X	100 %
Status:	Benar	Benar	Benar	Benar		
	Y1.jpg	Y2.jpg	Y3.jpg	Y4.jpg		
24.					Y	100 %
Status:	Benar	Benar	Benar	Benar		
Total Akurasi						94,8 %

Lampiran 2. Hasil *Testing* Setiap Data

Nama Citra	<i>Minimum Distances</i>	<i>Output</i>	<i>Target</i>
A5.jpg	0,311678541086664	0	0
A6.jpg	0,267867725477696	0	0
A7.jpg	0,507796467100764	0	0
A8.jpg	0,311678541086664	0	0
A9.jpg	0,130284274096983	0	0
A10.jpg	0,74525267714057	0	0
A11.jpg	0,507796467100764	0	0
A12.jpg	0,420839789992932	0	0
A13.jpg	0,113862529307188	0	0
A14.jpg	0,809594283352443	0	0
A15.jpg	1,06520666590683	0	0
A16.jpg	0,873759044934189	0	0
A17.jpg	0,809594283352443	0	0
A18.jpg	1,08638284553037	0	0
A19.jpg	0,680732755982739	0	0
A20.jpg	0,223972576977022	0	0
A21.jpg	0,289783644521592	0	0
A22.jpg	1,31808010724806	0	0
A23.jpg	1,29710979912091	0	0
A24.jpg	1,27612102307625	0	0
B5.jpg	0,325247423629246	1	1
B6.jpg	0,273538634092173	1	1
B7.jpg	0,359771263747867	1	1
B8.jpg	0,325247423629246	1	1
B9.jpg	0,515640818232185	1	1
B10.jpg	0,46359071229621	1	1
B11.jpg	0,0334385793992276	1	1
B12.jpg	0,289296257889362	1	1
B13.jpg	0,0847225236032081	1	1
B14.jpg	0,0688607271286408	1	1
B15.jpg	0,0334385793992276	1	1
B16.jpg	0,18756135888605	1	1
B17.jpg	1,00605751983831	1	1
B18.jpg	0,98839639899553	1	1
B19.jpg	2,10855522880189	1	1
B20.jpg	1,56269716485814	1	1
B21.jpg	1,23704778259441	1	1
B22.jpg	1,343738007789	1	1
B23.jpg	1,33802676388582	2	1
B24.jpg	1,55912706498491	1	1
C5.jpg	1,02020038101474	2	2
C6.jpg	0,940007583398607	2	2
C7.jpg	1,08051488024376	2	2
C8.jpg	0,327031792852747	2	2

C9.jpg	0,405297932646054	2	2
C10.jpg	0,800286202831259	2	2
C11.jpg	1,06039378768009	2	2
C12.jpg	1,16116277071231	2	2
C13.jpg	0,860071404708865	2	2
C14.jpg	1,20158538837565	2	2
C15.jpg	0,760508326173952	2	2
C16.jpg	0,48075798789597	2	2
C17.jpg	0,423880167646782	2	2
C18.jpg	0,366879279197629	2	2
C19.jpg	0,24900427100533	2	2
C20.jpg	0,271602130425389	2	2
C21.jpg	0,137628864598565	2	2
C22.jpg	0,612998745850998	2	2
C23.jpg	0,744583674954566	2	2
C24.jpg	0,631836517711307	2	2
D5.jpg	0,0744986336084388	3	3
D6.jpg	0,250127789759297	3	3
D7.jpg	0,00534547891572146	3	3
D8.jpg	0,110403120253977	3	3
D9.jpg	0,0640292038690902	3	3
D10.jpg	0,770449330970891	3	3
D11.jpg	0,203451759264777	3	3
D12.jpg	0,651062348791992	3	3
D13.jpg	0,29690584472937	3	3
D14.jpg	0,556039239063004	3	3
D15.jpg	0,461442248374901	3	3
D16.jpg	0,417014141921918	3	3
D17.jpg	0,627266307163353	3	3
D18.jpg	0,532350266849118	3	3
D19.jpg	1,27951356132863	3	3
D20.jpg	0,390770723322692	3	3
D21.jpg	0,461442248374901	3	3
D22.jpg	1,0112949082417	3	3
D23.jpg	0,914620995464272	3	3
D24.jpg	0,6511062348791992	3	3
E5.jpg	0,249761107501502	4	4
E6.jpg	0,209138870489351	4	4
E7.jpg	0,229441609816849	4	4
E8.jpg	0,19323104702816	5	4
E9.jpg	0,0735176728672187	5	4
E10.jpg	0,0130969293129155	4	4
E11.jpg	0,470367871267808	5	4
E12.jpg	0,229441609816849	4	4
E13.jpg	0,249761107501502	4	4
E14.jpg	0,494917536227348	4	4

E15.jpg	0,229441609816849	4	4
E16.jpg	0,763319702414599	4	4
E17.jpg	0,825684561070563	4	4
E18.jpg	0,84650885999919	4	4
E19.jpg	0,78409002896726	4	4
E20.jpg	0,412925414257959	4	4
E21.jpg	0,331207515736665	4	4
E22.jpg	0,148330789542253	4	4
E23.jpg	1,37311967138819	4	4
E24.jpg	2,19700555418694	4	4
F5.jpg	1,99934170853974	5	5
F6.jpg	1,71934286181568	5	5
F7.jpg	1,32232716968059	5	5
F8.jpg	0,957875883956788	5	5
F9.jpg	0,0266831311553162	5	5
F10.jpg	0,113485723965049	5	5
F11.jpg	0,19323104702816	5	5
F12.jpg	0,233008923685468	5	5
F13.jpg	0,0130969293129155	4	5
F14.jpg	0,00702416325074751	4	5
F15.jpg	0,351966992396296	5	5
F16.jpg	0,0735176728672187	5	5
F17.jpg	0,470367871267808	5	5
F18.jpg	1,8130015871091	5	5
F19.jpg	1,37941054879903	5	5
F20.jpg	1,60651599571042	5	5
F21.jpg	1,22691133411126	5	5
F22.jpg	1,4173974175062	5	5
F23.jpg	1,34136874599461	5	5
F24.jpg	2,55078100093539	5	5
G5.jpg	0,0195949239648066	6	6
G6.jpg	0,0188802233370104	7	6
G7.jpg	0,169257703486007	7	6
G8.jpg	0,550275928219872	7	6
G9.jpg	0,808120118294049	6	6
G10.jpg	0,998167790403059	6	6
G11.jpg	0,422757079740219	6	6
G12.jpg	0,543954144114288	6	6
G13.jpg	0,688459934536745	6	6
G14.jpg	1,04541337895418	6	6
G15.jpg	1,25672783756978	6	6
G16.jpg	1,18652228623863	6	6
G17.jpg	0,760338238903959	6	6
G18.jpg	0,51977159830891	6	6
G19.jpg	1,1630690311996	6	6
G20.jpg	0,736406296358187	6	6

G21.jpg	1,23335160496903	6	6
G22.jpg	0,808120118294049	6	6
G23.jpg	1,3964525070751	6	6
G24.jpg	3,16192160170366	7	6
H5.jpg	0,34614901424559	7	7
H6.jpg	0,202783227912889	6	7
H7.jpg	0,00607303792562419	7	7
H8.jpg	0,0438646178377979	7	7
H9.jpg	0,524643321894661	7	7
H10.jpg	0,664445325221898	6	7
H11.jpg	0,104250009853672	6	7
H12.jpg	0,300841578797957	6	7
H13.jpg	0,627376778890198	7	7
H14.jpg	0,678947886800714	7	7
H15.jpg	0,244874744614467	7	7
H16.jpg	0,524643321894661	7	7
H17.jpg	0,998167790403059	6	7
H18.jpg	1,71356183801444	7	7
H19.jpg	1,46968899683109	7	7
H20.jpg	1,60477570325202	7	7
H21.jpg	1,90550747062074	7	7
H22.jpg	2,0438646178378	7	7
H23.jpg	1,25557038978186	7	7
H24.jpg	1,68630486715198	7	7
I5.jpg	0,0728552564943925	9	8
I6.jpg	0,13724935489163	8	8
I7.jpg	0,60691912335194	8	8
I8.jpg	0,875520866421653	8	8
I9.jpg	0,407556080086902	8	8
I10.jpg	0,0941078157643389	9	8
I11.jpg	0,718444402598273	8	8
I12.jpg	0,144434295236742	8	8
I13.jpg	0,188077893076976	8	8
I14.jpg	0,584680419192186	8	8
I15.jpg	0,830528716309182	8	8
I16.jpg	1,03371649254757	8	8
I17.jpg	0,60691912335194	8	8
I18.jpg	1,01104768688226	8	8
I19.jpg	0,830528716309182	8	8
I20.jpg	0,808066726276085	8	8
I21.jpg	0,407556080086902	8	8
I22.jpg	0,696094891996779	8	8
I23.jpg	0,518095832239215	8	8
I24.jpg	0,407556080086902	8	8
K5.jpg	0,0356869339903447	8	9
K6.jpg	0,0336969546760493	9	9

K7.jpg	0,0791241775961709	8	9
K8.jpg	0,0550656490622998	9	9
K9.jpg	0,13655546898789	9	9
K10.jpg	0,13655546898789	9	9
K11.jpg	0,431575996511345	9	9
K12.jpg	0,157750666296593	9	9
K13.jpg	0,0356869339903447	8	9
K14.jpg	0,305587397960533	9	9
K15.jpg	0,144434295236742	9	9
K16.jpg	0,452509514581742	9	9
K17.jpg	0,326631871894126	9	9
K18.jpg	0,389653799510668	9	9
K19.jpg	0,13655546898789	9	9
K20.jpg	0,347657730625336	9	9
K21.jpg	0,0728552564943925	9	9
K22.jpg	0,180282521629845	8	9
K23.jpg	0,165608887326833	10	9
K24.jpg	0,0515834728261595	9	9
L5.jpg	0,121127229267323	10	10
L6.jpg	0,0431340174507149	10	10
L7.jpg	0,141738122326235	10	10
L8.jpg	0,245056492991242	10	10
L9.jpg	0,203676183394869	10	10
L10.jpg	0,100533825308279	10	10
L11.jpg	0,489261180958941	10	10
L12.jpg	0,388573958437217	10	10
L13.jpg	0,589542565303077	10	10
L14.jpg	1,51321826568672	10	10
L15.jpg	0,828590997546467	10	10
L16.jpg	3,01392978379893	10	10
L17.jpg	2,44236630170696	10	10
L18.jpg	2,7204339951614	10	10
L19.jpg	1,76294423295353	10	10
L20.jpg	2,25537576065794	10	10
L21.jpg	2,19902217232615	10	10
L22.jpg	2,84922751546195	10	10
L23.jpg	2,59102569817357	10	10
L24.jpg	2,60955057170948	10	10
M5.jpg	0,549884430594489	11	11
M6.jpg	0,202467353131091	11	11
M7.jpg	0,623561052064051	11	11
M8.jpg	0,177420340952917	11	11
M9.jpg	0,500617930406207	11	11
M10.jpg	0,152341842864189	11	11
M11.jpg	0,451231413849598	11	11
M12.jpg	0,794446798896047	11	11

M13.jpg	0,721384529612109	11	11
M14.jpg	0,401723999524929	11	11
M15.jpg	0,623561052064051	11	11
M16.jpg	0,672531297953785	11	11
M17.jpg	0,227482997692913	11	11
M18.jpg	0,304594668375302	11	11
M19.jpg	0,177420340952917	11	11
M20.jpg	0,127231739825312	11	11
M21.jpg	0,302342897577645	11	11
M22.jpg	0,252467392193701	11	11
M23.jpg	0,794446798896047	11	11
M24.jpg	1,93996250218008	11	11
N5.jpg	0,103267504003068	12	12
N6.jpg	0,386431331855423	12	12
N7.jpg	0,291621999838636	12	12
N8.jpg	0,410200470075971	12	12
N9.jpg	0,386431331855423	12	12
N10.jpg	0,481669671947646	12	12
N11.jpg	0,450708204806368	15	12
N12.jpg	0,197235930109308	12	12
N13.jpg	0,0798400076419874	12	12
N14.jpg	0,17370488623531	12	12
N15.jpg	0,103267504003068	12	12
N16.jpg	0,747076501114226	12	12
N17.jpg	0,747076501114226	12	12
N18.jpg	0,859500297903345	12	12
N19.jpg	0,769606799469909	12	12
N20.jpg	0,993664980301194	12	12
N21.jpg	0,769606799469909	12	12
N22.jpg	0,679347875791372	12	12
N23.jpg	0,769606799469909	12	12
N24.jpg	1,03820854253962	12	12
O5.jpg	0,239105267242998	13	13
O6.jpg	1,39269711627666	13	13
O7.jpg	0,566017106048132	13	13
O8.jpg	0,169809324442365	13	13
O9.jpg	2,88635930481823	13	13
O10.jpg	0,804578438955687	13	13
O11.jpg	1,04122352347821	13	13
O12.jpg	1,1420675651459	13	13
O13.jpg	0,0307152892053217	13	13
O14.jpg	0,962496438104324	13	13
O15.jpg	0,639409834112982	13	13
O16.jpg	0,643226488299163	14	13
O17.jpg	0,320006655747274	13	13
O18.jpg	0,908387435333566	13	13

O19.jpg	0,117727853519536	13	13
O20.jpg	0,871702631772934	14	13
O21.jpg	0,497005605699336	13	13
O22.jpg	1,8244455563593	13	13
O23.jpg	0,212932483012469	14	13
O24.jpg	1,22762870839595	14	13
P5.jpg	0,355523232153978	14	14
P6.jpg	0,527444933017122	14	14
P7.jpg	1,05435847789476	14	14
P8.jpg	0,233251980697123	14	14
P9.jpg	0,233251980697123	14	14
P10.jpg	0,114882267518514	15	14
P11.jpg	0,083438279645403	12	14
P12.jpg	0,083438279645403	12	14
P13.jpg	0,083438279645403	12	14
P14.jpg	1,0182379336135	14	14
P15.jpg	1,59209965595031	14	14
P16.jpg	0,560677534171617	14	14
P17.jpg	1,20797611115938	14	14
P18.jpg	1,20797611115938	14	14
P19.jpg	1,30363749009954	12	14
P20.jpg	0,474905788502035	12	14
P21.jpg	0,192629743684972	14	14
P22.jpg	0,519508170908374	14	14
P23.jpg	0,643226488299163	14	14
P24.jpg	0,30961044675826	14	14
Q5.jpg	0,492504350375935	15	15
Q6.jpg	0,108315009431358	15	15
Q7.jpg	0,416402265150438	15	15
Q8.jpg	0,280228376978936	15	15
Q9.jpg	0,0153782436720498	15	15
Q10.jpg	0,377818660892203	15	15
Q11.jpg	0,619999637589871	15	15
Q12.jpg	0,568901981016513	15	15
Q13.jpg	2,55800306162293	15	15
Q14.jpg	2,0540496470071	15	15
Q15.jpg	1,27037629559075	15	15
Q16.jpg	0,157581509619639	15	15
Q17.jpg	0,5773428970904	12	15
Q18.jpg	0,441736999965915	15	15
Q19.jpg	0,0899600233972038	15	15
Q20.jpg	0,571621430045784	15	15
Q21.jpg	1,32337624186516	15	15
Q22.jpg	3,37196429953527	15	15
Q23.jpg	3,61187456603509	15	15
Q24.jpg	0,034189953442219	15	15

R5.jpg	0,0364344745712941	16	16
R6.jpg	0,275729766067741	16	16
R7.jpg	0,140290479568563	17	16
R8.jpg	0,0740960417176524	17	16
R9.jpg	0,451793873116689	17	16
R10.jpg	0,609187671377747	17	16
R11.jpg	0,0291246826161782	16	16
R12.jpg	0,0961393800288164	17	16
R13.jpg	0,210348761022487	16	16
R14.jpg	0,162398366623457	17	16
R15.jpg	0,162398366623457	17	16
R16.jpg	0,0800365662900902	16	16
R17.jpg	0,0800365662900902	16	16
R18.jpg	0,145284866234771	16	16
R19.jpg	0,275229805372547	16	16
R20.jpg	0,544983869026119	19	16
R21.jpg	0,575634798909245	16	16
R22.jpg	0,296816452305031	16	16
R23.jpg	0,275229805372547	16	16
R24.jpg	0,210348761022487	16	16
S5.jpg	0,295502423293854	17	17
S6.jpg	0,362350975130006	17	17
S7.jpg	0,340045985532281	17	17
S8.jpg	0,0300734757339391	17	17
S9.jpg	0,184527896265656	17	17
S10.jpg	0,496650026217999	17	17
S11.jpg	0,58663446471925	17	17
S12.jpg	0,654363090042104	17	17
S13.jpg	0,541596706645215	17	17
S14.jpg	0,541596706645215	17	17
S15.jpg	0,973212759834716	17	17
S16.jpg	0,744991931992846	17	17
S17.jpg	0,0520740943369908	17	17
S18.jpg	0,295502423293854	17	17
S19.jpg	0,118204171725836	17	17
S20.jpg	0,0300734757339391	17	17
S21.jpg	0,0358010244446731	17	17
S22.jpg	0,609187671377747	17	17
S23.jpg	0,813209077437936	17	17
S24.jpg	0,00809412405600085	17	17
T5.jpg	0,219590197104257	18	18
T6.jpg	0,312158785990569	18	18
T7.jpg	1,21351534670957	18	18
T8.jpg	0,214452113969514	18	18
T9.jpg	0,8833034013735	18	18
T10.jpg	0,70775224815192	18	18

T11.jpg	0,459601709939413	18	18
T12.jpg	0,263247152193987	18	18
T13.jpg	1,0350210390605	18	18
T14.jpg	0,959016709367518	18	18
T15.jpg	0,984318984521209	18	18
T16.jpg	0,123916971961503	18	18
T17.jpg	0,051873048173821	18	18
T18.jpg	0,287688342391835	18	18
T19.jpg	0,533726765928552	18	18
T20.jpg	0,31482853719648	18	18
T21.jpg	0,123916971961503	18	18
T22.jpg	0,147876375168394	18	18
T23.jpg	0,0687583228808819	18	18
T24.jpg	0,141476360678706	18	18
U5.jpg	0,333314866226466	19	19
U6.jpg	0,0817782850722146	19	19
U7.jpg	0,566255652694352	19	19
U8.jpg	0,447363579173299	16	19
U9.jpg	0,481283656532622	19	19
U10.jpg	0,270181394895175	19	19
U11.jpg	0,0401116027071851	19	19
U12.jpg	0,16532961093877	19	19
U13.jpg	0,417755066562034	19	19
U14.jpg	0,532955821231347	16	19
U15.jpg	0,291207253626386	19	19
U16.jpg	1,82342230945376	19	19
U17.jpg	1,41454250407818	19	19
U18.jpg	0,778548795112631	19	19
U19.jpg	1,19772597899437	19	19
U20.jpg	1,19772597899437	19	19
U21.jpg	0,556131348546529	19	19
U22.jpg	1,29650876457612	19	19
U23.jpg	1,74603989949207	19	19
U24.jpg	1,17792302661211	19	19
V5.jpg	0,753356509319111	20	20
V6.jpg	0,375713045464263	20	20
V7.jpg	0,794951041422415	20	20
V8.jpg	0,46015324579983	20	20
V9.jpg	0,649053696593281	20	20
V10.jpg	0,690829311812362	20	20
V11.jpg	0,417970775044505	20	20
V12.jpg	0,753356509319111	20	20
V13.jpg	0,163281238110446	20	20
V14.jpg	0,269736802270124	20	20
V15.jpg	0,640050791639993	20	20
V16.jpg	0,640050791639993	20	20

V17.jpg	0,640050791639993	20	20
V18.jpg	0,00806109038349945	20	20
V19.jpg	0,684288208337087	20	20
V20.jpg	0,354589314830317	20	20
V21.jpg	0,529833787805369	20	20
V22.jpg	0,898624157579629	20	20
V23.jpg	0,267467789609515	20	20
V24.jpg	0,137400777168761	20	20
W5.jpg	0,195675265579936	21	21
W6.jpg	0,275357273249671	21	21
W7.jpg	0,63710737507607	21	21
W8.jpg	0,657360087026873	21	21
W9.jpg	0,312184455493675	20	21
W10.jpg	0,960615089561919	20	21
W11.jpg	0,738537693035319	21	21
W12.jpg	0,753356509319111	20	21
W13.jpg	0,942663610232071	21	21
W14.jpg	0,395357618853154	21	21
W15.jpg	0,88124692425804	21	21
W16.jpg	1,44157023245876	21	21
W17.jpg	1,38549298624177	21	21
W18.jpg	1,08440418171281	21	21
W19.jpg	0,316225231228312	21	21
W20.jpg	1,14111914421445	21	21
W21.jpg	1,62765207989252	21	21
W22.jpg	1,57196264861441	21	21
W23.jpg	2,1965143503394	21	21
W24.jpg	1,47888991005084	21	21
X5.jpg	0,128903439564574	23	22
X6.jpg	0,0213913868813371	22	22
X7.jpg	0,0746119401132361	22	22
X8.jpg	0,336373434158109	22	22
X9.jpg	0,14202275793604	22	22
X10.jpg	0,242101404935024	23	22
X11.jpg	0,117840212130663	22	22
X12.jpg	0,0790005028289578	23	22
X13.jpg	4,99461277686597	22	22
X14.jpg	3,00150726294301	22	22
X15.jpg	0,263362493533727	22	22
X16.jpg	1,64875701703985	22	22
X17.jpg	1,10828547411566	22	22
X18.jpg	0,906252335575466	22	22
X19.jpg	0,656559398097979	22	22
X20.jpg	0,336511587249145	22	22
X21.jpg	0,681389280209757	22	22
X22.jpg	0,532866144994571	22	22

X23.jpg	0,60699120098371	22	22
X24.jpg	4,83728165503655	22	22
Y5.jpg	0,243551377708474	23	23
Y6.jpg	0,312052468094336	23	23
Y7.jpg	0,266408896872317	23	23
Y8.jpg	0,334838668733177	23	23
Y9.jpg	0,00947694639266317	23	23
Y10.jpg	0,0558010785191492	23	23
Y11.jpg	0,493684656129009	23	23
Y12.jpg	0,561413281451863	23	23
Y13.jpg	0,403055814178266	23	23
Y14.jpg	0,148749177783085	23	23
Y15.jpg	0,47106230300755	23	23
Y16.jpg	0,493684656129009	23	23
Y17.jpg	0,808001760638831	23	23
Y18.jpg	0,334838668733177	23	23
Y19.jpg	0,174834986336396	23	23
Y20.jpg	0,785696771041106	23	23
Y21.jpg	0,673837078240982	23	23
Y22.jpg	0,561413281451863	23	23
Y23.jpg	0,763369554232419	23	23
Y24.jpg	0,583943579807546	23	23

Lampiran 3. Hasil *Testing* Keseluruhan

Huruf	Jumlah Data									
	96 (4 set)		192 (8 set)		288 (12 set)		384 (16 set)		480 (20 set)	
	B	S	B	S	B	S	B	S	B	S
A	4	0	8	0	12	0	16	0	20	0
B	4	0	8	0	12	0	16	0	19	1
C	4	0	8	0	12	0	16	0	20	0
D	4	0	8	0	12	0	16	0	20	0
E	3	1	5	3	9	3	13	3	17	3
F	4	0	8	0	10	2	14	2	18	2
G	1	3	5	3	9	3	13	3	16	4
H	3	1	4	4	8	4	11	5	15	5
I	3	1	6	2	10	2	14	2	18	2
K	2	2	6	2	8	4	12	4	14	6
L	4	0	8	0	12	0	16	0	20	0
M	4	0	8	0	12	0	16	0	20	0
N	4	0	7	1	11	1	15	1	19	1
O	1	3	5	3	8	4	11	5	13	7
P	4	0	5	3	8	4	10	6	14	6
Q	4	0	8	0	12	0	15	1	19	1
R	2	2	3	5	5	7	8	8	12	8
S	4	0	8	0	12	0	16	0	20	0
T	4	0	8	0	12	0	16	0	20	0
U	3	1	7	1	10	2	14	2	18	2
V	4	0	8	0	12	0	16	0	20	0
W	4	0	5	3	9	3	13	3	17	3
X	3	1	5	3	9	3	13	3	17	3
Y	4	0	8	0	12	0	16	0	20	0
Akurasi (%)	84,38 %		82,82 %		85,42 %		87,5 %		88,75 %	

Keterangan:

Set : Jumlah set setiap hurufnya.

B : Jumlah keluaran huruf yang benar.

S : Jumlah keluaran huruf yang salah.

Lampiran 4. Uji Coba Sistem di SLB Islam Yasindo Malang

Form Uji Coba Sistem oleh User 1

FORM UJI COBA SISTEM

Nama : IDA AYU JULIA

Jabatan : Guru.

*) Beri tanda check (✓) pada salah satu kolom penilaian yang sesuai

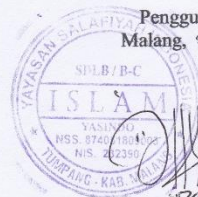
No.	Kriteria	Penilaian				
		SS	S	N	TS	STS
A. Kemudahan Penggunaan Sistem						
1.	Menu yang tersedia pada sistem mudah dimengerti.	✓				
2.	Sistem mudah dioperasikan.		✓			
B. Manfaat Sistem						
1.	Sistem dapat menerjemahkan citra huruf bahasa isyarat ke dalam huruf tertulis.		✓			
2.	Sistem dapat membantu dalam pengenalan huruf bahasa isyarat.	✓				

Catatan:

.....

.....

.....



Pengguna Sistem,
Malang, 31 Juli 2017

NIK.

Keterangan:

SS = Sangat Setuju
S = Setuju
N = Netral

TS = Tidak Setuju
STS = Sangat Tidak Setuju

Form Uji Coba Sistem oleh User 2

FORM UJI COBA SISTEM

Nama : AMINATU2 ZUHRIZAH

Jabatan : GURU

*) Beri tanda *check* (✓) pada salah satu kolom penilaian yang sesuai

No.	Kriteria	Penilaian				
		SS	S	N	TS	STS
A. Kemudahan Penggunaan Sistem						
1.	Menu yang tersedia pada sistem mudah dimengerti.	✓				
2.	Sistem mudah dioperasikan.		✓			
B. Manfaat Sistem						
1.	Sistem dapat menerjemahkan citra huruf bahasa isyarat ke dalam huruf tertulis.		✓			
2.	Sistem dapat membantu dalam pengenalan huruf bahasa isyarat.	✓				

Catatan:

.....

.....

.....

Pengguna Sistem,
Malang, 31 Juli 2017AMINATU2 ZUHRIZAH
NIK.

Keterangan:

SS = Sangat Setuju
S = Setuju
N = Netral

TS = Tidak Setuju
STS = Sangat Tidak Setuju

Form Uji Coba Sistem oleh User 3

FORM UJI COBA SISTEM

Nama : ETIK NORMAWATI

Jabatan : GURU

*) Beri tanda *check* (✓) pada salah satu kolom penilaian yang sesuai

No.	Kriteria	Penilaian				
		SS	S	N	TS	STS
A. Kemudahan Penggunaan Sistem						
1.	Menu yang tersedia pada sistem mudah dimengerti.	✓				
2.	Sistem mudah dioperasikan.		✓			
B. Manfaat Sistem						
1.	Sistem dapat menerjemahkan citra huruf bahasa isyarat ke dalam huruf tertulis.	✓				
2.	Sistem dapat membantu dalam pengenalan huruf bahasa isyarat.	✓				

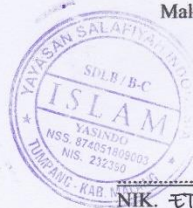
Catatan:

.....

.....

.....

Pengguna Sistem,
Malang, 31 Juli 2017



NIK. ETIK NORMAWATI

Keterangan:

SS = Sangat Setuju
S = Setuju
N = Netral

TS = Tidak Setuju
STS = Sangat Tidak Setuju

Form Uji Coba Sistem oleh User 4

FORM UJI COBA SISTEM

Nama : HAERUL AKBAR

Jabatan : GURU

*) Beri tanda *check* (✓) pada salah satu kolom penilaian yang sesuai

No.	Kriteria	Penilaian				
		SS	S	N	TS	STS
A. Kemudahan Penggunaan Sistem						
1.	Menu yang tersedia pada sistem mudah dimengerti.		✓			
2.	Sistem mudah dioperasikan.		✓			
B. Manfaat Sistem						
1.	Sistem dapat menerjemahkan citra huruf bahasa isyarat ke dalam huruf tertulis.		✓			
2.	Sistem dapat membantu dalam pengenalan huruf bahasa isyarat.		✓			

Catatan:

.....

.....

.....

.....

 Pengguna Sistem,
 Malang, 31 Juli 2017



HAERUL AKBAR

NIK.

Keterangan:

 SS = Sangat Setuju
 S = Setuju
 N = Netral

 TS = Tidak Setuju
 STS = Sangat Tidak Setuju

Form Uji Coba Sistem oleh User 5

FORM UJI COBA SISTEM

Nama : imam Syuhro W

Jabatan : Guru

*) Beri tanda *check* (✓) pada salah satu kolom penilaian yang sesuai

No.	Kriteria	Penilaian				
		SS	S	N	TS	STS
A. Kemudahan Penggunaan Sistem						
1.	Menu yang tersedia pada sistem mudah dimengerti.			✓		
2.	Sistem mudah dioperasikan.			✓		
B. Manfaat Sistem						
1.	Sistem dapat menerjemahkan citra huruf bahasa isyarat ke dalam huruf tertulis.			✓		
2.	Sistem dapat membantu dalam pengenalan huruf bahasa isyarat.			✓		

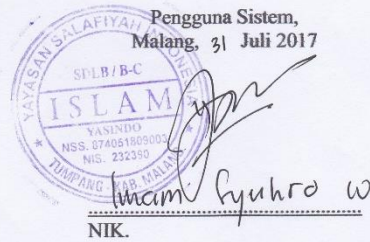
Catatan:

.....

.....

.....

Pengguna Sistem,
Malang, 31 Juli 2017



Imam Syuhro W









































NIK.

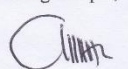
Keterangan:

SS = Sangat Setuju
S = Setuju
N = Netral









































TS = Tidak Setuju
STS = Sangat Tidak Setuju

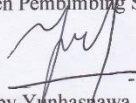
Lampiran 5. Lembar Bimbingan Skripsi Pembimbing I

	KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI POLITEKNIK NEGERI MALANG JURUSAN TEKNOLOGI INFORMASI PROGRAM STUDI TEKNIK INFORMATIKA Jl. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122			
<div style="border: 1px solid black; padding: 2px 5px; display: inline-block;">NO SKRIPSI: 100</div>				
LEMBAR BIMBINGAN SKRIPSI 2016/2017				
JUDUL : Penerapan Algoritma <i>Learning Vector Quantization with Adaptive Prototype Addition and Removal</i> untuk Pengenalan Huruf Bahasa Isyarat (SLB: Islam Yasindo Malang)				
NAMA : Yusliana Gadis Buata NIM : 1341180150				
No.	Tanggal	Materi Bimbingan	Tanda Tangan	
			Mahasiswa	Dosen
1.	03/03/2017	Pembahasan data training		
2.	17/03/2017	Pembahasan data training, validasi, testing		
3.	24/03/2017	Skema Alur Sistem		
4.	31/03/2017	Tahapan Preprocessing Data		
5.	07/04/2017	Pembagian Folder Data dan APAR		
6.	21/04/2017	Programming database		
7.	28/04/2017	Pengelasan alur testing + excel.		
8.	12/05/2017	Excel LVA		
9.	26/05/2017	Konsep overall Adaptive LVA		
10.	09/06/2017	Pembahasan training & Testing LVA		
11.	14/06/2017	Pengolahan Data Training		
12.	16/06/2017	Testing		
13.	19/06/2017	Testing, Perbaikan inisialisasi bobot		
14.	21/06/2017	Perbaikan pembagian matrix X		
15.	10/07/2017	Perbaikan Akurasi Sistem		
16.	14/07/2017	Adaptive Addition + Removal (Validasi)		
17.	16/07/2017	Laporan BAB I - IV		
18.	19/07/2017	Laporan BAB V - VII		
19.	26/07/2017	Lampiran Pengujian		


Malang, 26... 7... 2017
 Dosen Pembimbing Skripsi,

Ariadi Retno Tri Hayati Ririd, S.Kom., M.Kom
 NIP. 19810810 200501 2 002

Lampiran 6. Lembar Bimbingan Skripsi Pembimbing II


	KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI POLITEKNIK NEGERI MALANG JURUSAN TEKNOLOGI INFORMASI PROGRAM STUDI TEKNIK INFORMATIKA Jl. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122			
<div style="border: 1px solid black; padding: 2px 5px; display: inline-block;">NO SKRIPSI: 100</div>				
LEMBAR BIMBINGAN SKRIPSI 2016/2017				
JUDUL : Penerapan Algoritma <i>Learning Vector Quantization with Adaptive Prototype Addition and Removal</i> untuk Pengenalan Huruf Bahasa Isyarat (SLB: Islam Yasindo Malang)				
NAMA : Yusliana Gadis Buata NIM : 1341180150				
No.	Tanggal	Materi Bimbingan	Tanda Tangan	
			Mahasiswa	Dosen
1.	6 / 3 / 17	PRETEST PROGRAMMING + PRESENTASI		
2.	10 / 3 / 17	PEMAHAMAN LVQ + CODING INPUT VECTOR		
3.	14 / 3 / 17	PEMAHAMAN LVQ + DASAR VB.NET		
4.	17 / 3 / 17	FUNGSI EUCLIDEAN DISTANCE		
5.	21 / 3 / 17	FUNGSI PERBAIKAN BOBOT		
6.	24 / 3 / 17	PENGUNAAN ARRAYLIST PADA CORE LVQ		
7.	29 / 3 / 17	CORE LVQ + PREPROCESSING AWAL		
8.	31 / 3 / 17	CORE LVQ		
9.	5 / 4 / 17	CORE LVQ + DENOISING IMAGE		
10.	12 / 4 / 17	CORE LVQ PERBAIKAN BOBOT		
11.	26 / 4 / 17	CORE LVQ + INISIALISASI TESTING		
12.	3 / 5 / 17	PERBAIKAN CORE LVQ BAGIAN TESTING		
13.	10 / 5 / 17	SYSTEM.DRAMING IMAGE → MATRIX.LVQFEED		
14.	17 / 5 / 17	IMAGEFROMPATH . OVER-ALL TRAINING PROCESS		
15.	22 / 5 / 17	CONVERT IMAGE DARI DE KE MATRIX		
16.	26 / 5 / 17	DENOISING , TESTING , VALIDATION		
17.	09 / 6 / 17	SERIALIZE + DESERIALIZE ARRAYLIST		
18.	12 / 6 / 17	EKSTRAKSI FITUR BINER		
19.	22 / 6 / 17	OVERHAUL LVQ ENGINE		

Malang, 25 Juli 2017....
 Dosen Pembimbing Skripsi,

 Yoppy Yunhasnawa, S.ST., M.sc

Lampiran 7. Lembar Revisi Skripsi Penguji I



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
 JL. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122

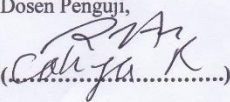


FORM REVISI SKRIPSI

No. Skripsi : 100

Nama Mahasiswa : Yuslana Gadis Buata **NIM** : 1341180150
Tanggal Ujian : 10 Agustus 2017
Judul : Penerapan Algoritma *Learning Vector Quantization with Adaptive Addition and Removal* untuk Pengenalan Huruf Bahasa Isyarat (Studi Kasus: SLB Islam Yasindo Malang)

NO	SARAN PERBAIKAN	PARAF
1	biri contoh dilaporan untuk 3x3 pixel	RPN
2	cara proses LVQ. 3x3 ⊕ adaptive.	RPN


Malang, Kamis, 10-8-2017
 Dosen Penguji,

 (Cahaya K.)

FORM VERIFIKASI:
 Laporan Akhir telah diperbaiki sesuai dengan saran perbaikan dari dosen penguji.


PENGUJI/PEMBIMBING	NAMA	TTD	TANGGAL
Penguji	Cahaya K	RPN	14-08-17
Pembimbing 1	<u>Ariadi Retno Tri Hayati Ririd,</u> <u>S.Kom., M.Kom</u>	[Signature]	15-08-17
Pembimbing 2	<u>Yoppy Yunhasnawa, SST., M.Sc</u>	[Signature]	15-08-17

FRM.RTI.01.35.03

Lampiran 8. Lembar Revisi Skripsi Penguji II



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK INFORMATIKA
 Jl. Soekarno Hatta PO Box 04 Malang Telp. (0341) 404424 pes. 1122



FORM REVISI SKRIPSI

No. Skripsi : 100

Nama Mahasiswa : Yusliana Gadis Buata **NIM** : 1341180150
Tanggal Ujian :
Judul : Penerapan Algoritma *Learning Vector Quantization with Adaptive Addition and Removal* untuk Pengenalan Huruf Bahasa Isyarat (Studi Kasus: SLB Islam Yasindo Malang)

NO	SARAN PERBAIKAN	PARAF
1.	Revisi Hg WA adaptive sederhana	Aee
2.		

Malang, ... Kanis 10-8-2017

Dosen Penguji,

(Dosen Andrie A.)

FORM VERIFIKASI:
 Laporan Akhir telah diperbaiki sesuai dengan saran perbaikan dari dosen penguji.

PENGUJI/PEMBIMBING	NAMA	TTD	TANGGAL
Penguji	<u>Dosen Andrie A.</u>	/	14-08-17
Pembimbing 1	<u>Ariadi Retno Tri Havati Ririd</u> <u>S.Kom., M.Kom</u>	/	15-08-17
Pembimbing 2	<u>Yoppy Yunhasnawa, SST., M.Sc</u>	/	15-08-17

FRM.RTI.01.35.03

Lampiran 9. Profil Penulis**PROFIL PENULIS****DATA PRIBADI**

Nama	:	Yusliana Gadis Buata
Tempat, Tanggal Lahir	:	Malang, 12 April 1995
Alamat	:	Perum. Sekarsari Indah B.16 Malang
Email	:	yuslianagadis@gmail.com
No. Hp	:	081231581628

DATA PENDIDIKAN

(2001-2007)	Sekolah Dasar	:	SD Brawijaya Smart School Malang
(2007-2010)	SMP	:	SMP Negeri 4 Malang
(2010-2013)	SMA	:	SMA Negeri 9 Malang
(2013-2017)	Perguruan Tinggi	:	Politeknik Negeri Malang