

**PENGENALAN CITRA HURUF ALPHABET TULISAN  
TANGAN MENGGUNAKAN METODE  
*NAÏVE BAYES CLASSIFIER***

**SKRIPSI**

Oleh :

**MORWATI**  
**NIM. 10650091**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK  
IBRAHIM MALANG  
2014**

**PENGENALAN CITRA HURUF ALPHABET TULISAN  
TANGAN MENGGUNAKAN METODE  
*NAÏVE BAYES CLASSIFIER***

**SKRIPSI**

**Diajukan Kepada:  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**oleh:  
MORWATI  
NIM: 10650091 / S-1**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI  
MAULANA MALIK IBRAHIM MALANG  
2014**


**PENGENALAN CITRA HURUF ALPHABET TULISAN  
TANGAN MENGGUNAKAN METODE  
NAÏVE BAYES CLASSIFIER**

**SKRIPSI**

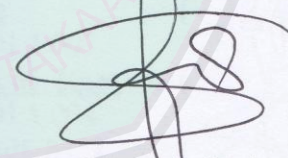
oleh :  
**MORWATI**  
**NIM. 10650091**

Telah Diperiksa dan Disetujui untuk Diuji :  
Tanggal : 12 November 2014


Pembimbing I,

  
Irwan Badi Santoso, M.Kom  
NIP. 19770103 201101 1 004

Pembimbing II,

  
Dr. M. Amin Hariyadi, M.T  
NIP. 19670118 200501 1 001

Mengetahui,  
Ketua Jurusan Teknik Informatika

  
Dr. Cahyo Crysdiyan  
NIP. 19740424 200901 1 008

**Pengenalan Citra Huruf Alfabeta Tulisan Tangan Menggunakan Metode Naïve Bayes Classifier**

**SKRIPSI**

oleh :

**MORWATI**


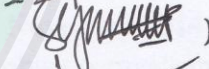
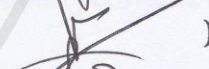
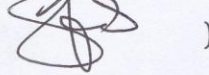
**NIM. 10650091**

Telah Dipertahankan di Depan Dewan Penguji Skripsi  
Dan dinyatakan sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal 20 November 2014


**Susunan Dewan Penguji :**

1. **Penguji Utama** : Dr. Cahyo Crysdian  
NIP. 19740424 200901 0008
2. **Ketua** : A'la Syaqui, M.Kom  
NIP. 19771201 200801 1 007
3. **Sekretaris** : Irwan Budi Santoso, M. Kom  
NIP. 19770103 201101 1 004
4. **Anggota** : Dr. M. Amin Hariyadi, M. T  
NIP. 19670118 200501 1 001

**Tanda Tangan**

(  )  
(  )  
(  )  
(  )

Mengesahkan,  
Ketua Jurusan Teknik Informatika



**Dr. Cahyo Crysdian**  
NIP. 19740424 200901 1 008

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini :

Nama : Morwati  
NIM : 10650091  
Jurusan : Teknik Informatika  
Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa tugas akhir/skripsi yang saya tulis yang berjudul “Pengenalan Citra Huruf Alphabet Tulisan Tangan Menggunakan Metode Naïve Bayes Classifier” ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pemikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan tugas akhir/skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, November 2014  
Yang membuat pernyataan,



Morwati  
NIM. 10650091

## MOTTO

إِنَّ اللَّهَ لَا يُغَيِّرُ مَا بِقَوْمٍ حَتَّىٰ يُغَيِّرُوا مَا بِأَنْفُسِهِمْ

*Sesungguhnya Allah tidak merubah keadaan sesuatu kaum sehingga mereka merubah keadaan yang ada pada diri mereka sendiri. (QS. Ar-Ra'd ayat 11)*

*Waktu akan terasa lambat bagi mereka yg menunggu, terlalu panjang bagi mereka yg gelisah, dan terlalu singkat bagi mereka yg bahagia, tetapi waktu akan terasa abadi bagi mereka yg mampu bersyukur.*

\*\*\*



## *Halaman Persembahan*

*Karya sederhana ini ku persembahkan untuk orang yang paling berarti dan paling aku cintai Bapak dan Ma'e terima kasih atas semua do'a yang selalu tereurah, atas ketulusan dan keikhlasannya dalam membimbingku, atas pengorbanan dan kerja kerasnya sehingga aku dapat menikmati manisnya ilmu. Dan terima kasih atas segalanya yang tak mungkin dapat ku balas hingga kapan pun. Semoga aku bisa menjadi anak yang bisa membanggakan Bapak dan Ma'e, amiin...*

*Untuk Mbak Nur dan Kakak terima kasih atas semua doa, dukungan, semangat dan nasihatnya, jazakumullah khoir, aku sayang kalian...*

*Terima kasih untuk Khullatur Rosyidah yang selalu setia berbagi suka dan duka dalam menyelesaikan skripsi ini, teman - teman seperjuangan Miftahur Rizqiyah, Aeny Nurwahdah, Anindita Caesarini, Amalia Eka Astutik, Dzakiyatur Rosyidah dan Fuad yang selalu ada dan memotivasi dalam menyelesaikan skripsi ini...*

*Serta kepada semua pihak yang tak dapat kusebutkan satu persatu yang telah membantu dan memotivasi dari awal kuliah hingga terselesaikannya skripsi ini, jazakumullah khoir...*

## KATA PENGANTAR



*Assalamu'alaikum Wr. Wb.*

Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang. Segala puji dan syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan rahmat, taufiq dan inayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pengenalan Citra Huruf Alphabet Tulisan Tangan Menggunakan Metode Naïve Bayes Classifier” ini dengan baik. Shalawat serta salam semoga tetap tercurahkan kepada junjungan kita Nabi Muhammad SAW dan yang telah menuntun umat manusia dari zaman kegelapan ke masa yang terang benderang dengan agama Islam.

Selanjutnya penulis haturkan ucapan terima kasih seiring do'a dan harapan *jazakumullah ahsanal jaza'* kepada semua pihak yang telah membantu terselesaikannya skripsi ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Irwan Budi Santoso, M.Kom, selaku dosen pembimbing I yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan dan memberi masukan dalam pengerjaan skripsi ini
2. Dr. M. Amin Hariyadi M.T, selaku dosen pembimbing II yang selalu memberikan masukan, nasehat serta petunjuk dalam penyusunan laporan skripsi ini.

3. Dr. Cahyo Crysdiyan, M.Kom selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang, yang mendukung dan mengarahkan dalam pengerjaan skripsi ini.
4. Suhartono, M.Kom selaku dosen wali yang selalu memberikan masukan dan arahan kepada penulis dari awal perkuliahan hingga terselesaikannya skripsi ini.
5. Segenap Dosen Teknik informatika dan teman-teman TI angkatan 2010 yang telah memberikan bimbingan keilmuan dan segala dukungan kepada penulis selama ini.
6. Semua pihak yang telah membantu dalam penyelesaian skripsi ini.  
Semoga Allah SWT memberikan balasan yang sesuai atas jasa dan bantuan yang telah diberikan.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat kekurangan dan penulis berharap semoga skripsi ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. *Amin Ya Rabbal Alamin.*

*Wassalamu'alaikum Wr. Wb.*

Malang, 12 November 2014

Penulis

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGAJUAN .....	ii
HALAMAN PERSETUJUAN .....	iii
HALAMAN PENGESAHAN .....	iv
HALAMAN PERNYATAAN.....	v
MOTTO .....	vi
HALAMAN PERSEMBAHAN .....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR TABEL .....	xii
DAFTAR GAMBAR.....	xiii
ABSTRAK .....	xiv
ABSTRACT .....	xv
المخلص .....	xvi
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	7
1.3 Tujuan Penelitian .....	7
1.4 Manfaat Penelitian .....	7
1.5 Batasan Masalah .....	8
1.6 Sistematika Penulisan .....	8
<b>BAB II KAJIAN Pustaka .....</b>	<b>10</b>
2.1 Kajian Pustaka .....	10
2.1.1 Pengenalan Pola.....	10
2.1.2 Definisi dan Jenis – Jenis Huruf .....	11
2.1.3 Definisi Citra .....	12
2.1.4 JPEG.....	13
2.1.5 <i>Greyscale</i> .....	14
2.1.6 <i>Projection Profile</i> .....	15
2.1.7 <i>Cropping</i> .....	17
2.1.8 Penskalaan ( <i>Scalling</i> ) .....	18
2.1.9 Ekstraksi Fitur .....	19
2.1.10 Distribusi Normal.....	20
2.1.11 <i>Naïve Bayes Classifier</i> (NBC) .....	21
2.2 Penelitian Terkait .....	24
<b>BAB III ANALISIS DAN PERANCANGAN SISTEM.....</b>	<b>28</b>
3.1 Deskripsi Data.....	28
3.2 Deskripsi Sistem .....	33
3.3 Desain Sistem.....	33
3.4 Desain <i>Interface</i> .....	39

<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>43</b>
4.1 Platform yang digunakan .....	43
4.2 Implementasi Sistem .....	43
4.2.1 Halaman Utama Aplikasi .....	43
4.2.2 <i>Interface</i> Proses <i>Training</i> .....	44
4.2.3 <i>Interface</i> Proses <i>Testing</i> .....	47
4.3 Uji Coba Sistem .....	50
4.3.1 Perancangan Uji Coba .....	50
4.3.2 Hasil Uji Coba Proses <i>Training</i> .....	51
4.3.3 Hasil Uji Coba Proses <i>Testing</i> .....	55
4.4 Pembahasan.....	64
 <b>BAB V KESIMPULAN DAN SARAN .....</b>	 <b>69</b>
5.1 Kesimpulan .....	69
5.2 Saran .....	70
 <b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	



## DAFTAR TABEL

Tabel 3.1 Data <i>training</i> huruf tulisan tangan .....	29
Tabel 4.1 Akurasi proses <i>training</i> berdasarkan dimensi pixel baris dan kolom ....	53
Tabel 4.2 Nilai confusion matrik hasil uji coba akurasi training .....	54
Tabel 4.3 Hasil uji coba proses <i>testing</i> dengan <i>input</i> huruf tunggal .....	55
Tabel 4.4 Hasil uji coba proses <i>testing</i> dengan <i>input</i> kata dan kalimat .....	46



## DAFTAR GAMBAR

Gambar 2.1 Citra hasil <i>Projection Profile</i> .....	16
Gambar 2.2 Sebuah citra di <i>cropping</i> .....	17
Gambar 2.3 Sebuah citra di <i>cropping</i> sebesar WxH.....	18
Gambar 2.4 Scalling citra huruf 'a' .....	19
Gambar 2.5 Contoh vektor.....	19
Gambar 2.6 Contoh transpose vektor.....	20
Gambar 3.1 Tahapan Penelitian .....	28
Gambar 3.2 Blok Diagram Desain Proses .....	35
Gambar 3.3 <i>Flowchart</i> proses estimasi parameter .....	36
Gambar 3.4 <i>Flowchart Projection Profile</i> .....	37
Gambar 3.5 <i>Flowchart NBC</i> .....	38
Gambar 3.6 Desain <i>interface</i> halaman utama .....	39
Gambar 3.7 Desain <i>interface</i> proses <i>training</i> .....	40
Gambar 3.8 Desain <i>interface</i> proses <i>testing</i> .....	41
Gambar 4.1 Halaman utama aplikasi .....	44
Gambar 4.2 <i>Interface</i> proses <i>training</i> .....	45
Gambar 4.3 <i>Source code</i> pembentukan data <i>training</i> .....	46
Gambar 4.4 <i>Source code</i> untuk mengambil data perkelas .....	46
Gambar 4.5 <i>Souce code</i> estimasi parameter.....	47
Gambar 4.6 <i>Interface</i> proses <i>testing</i> .....	48
Gambar 4.7 <i>Source code</i> proses pengenalan dengan NBC .....	49
Gambar 4.8 <i>Source code</i> perhitungan akurasi sistem .....	49
Gambar 4.9 Uji coba proses <i>training</i> .....	52
Gambar 4.10 Uji coba proses <i>testing</i> .....	59
Gambar 4.11 Citra uji hasil <i>Projection Profile</i> .....	60
Gambar 4.12 (a) hasil <i>cropping</i> citra <i>testing</i> (b) citra <i>training</i> .....	65

## ABSTRAK

Morwati. 2014. **Pengenalan Citra Huruf Alphabet Tulisan Tangan Menggunakan Metode *Naïve Bayes Classifier***. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Irwan Budi Santoso, M.Kom (II) Dr. M. Amin Haryadi, M.T.

**Kata Kunci:** *Pengenalan Tulisan Tangan, Naïve Bayes Classifier.*

Perkembangan teknologi yang sangat pesat saat ini tidak membuat pendokumentasian data/informasi dengan tulisan tangan dihilangkan. Bahkan dalam berbagai penelitian, menulis dengan tangan terbukti memiliki efek yang sangat baik untuk kecerdasan otak. Dalam penelitian ini akan dibangun aplikasi pengenalan tulisan tangan menggunakan metode *Naïve Bayes Classifier* dengan data training sebanyak 468 data dengan masing – masing huruf terdiri dari 18 sampel yang digunakan untuk pelatihan sistem. Dalam pengenalan pola, data training memiliki peran yang sangat penting dalam menentukan akurasi sistem, selain itu ukuran dimensi pixel baris dan kolom citra juga sangat berpengaruh. Dari hasil uji coba terhadap data training diperoleh akurasi training terbaik dengan dimensi baris dan kolom 7x4 sebesar 81,1966%. Dan berdasarkan hasil uji coba testing dengan inputan berupa citra huruf sebanyak 104 data diperoleh akurasi sebesar 79,81% dengan 83 jawaban benar dan 21 jawaban salah. Sedangkan pada uji coba dengan input berupa kata dan kalimat yang terdiri dari 173 huruf dan tanpa ROI diperoleh akurasi 27,75% dengan jumlah jawaban benar sebanyak 48 dan jawaban salah sebanyak 125 huruf, sedangkan dengan inputan yang sama dan dengan tambahan ROI diperoleh akurasi yang lebih baik yaitu 63,01% dengan 109 huruf yang berhasil dikenali dengan benar dan 64 huruf yang tidak dapat dikenali dengan benar.

## ABSTRACT

Morwati. 2014. **Handwriting Recognition with Naïve Bayes Classifier Method.**

Theses, Infomatic Engineering Programme Faculty of Science and Technology the State of Islamic University Maulana Malik Ibrahim Malang. Promotor: (I) Irwan Budi Santoso, M.Kom (II) Dr. M. Amin Haryadi, M.T.

**Keywords:** *Handwriting Recognition, Naïve Bayes Classifier.*

The currently rapid development of technology does not eliminate documentation of data / information by the handwriting. Even in the variety of researches, handwriting is to be proved as having a very good effect on human's intelligence. The application of handwriting within *Naïve Bayes Classifier* method with 468 training data which each letter is composed by 18 samples will be analyzed in this research). In the pattern's recognition, training data is the most important role in determining the accuracy of the system. Also, the size of the pixel dimensions of the image is highly influenced. From the test results of the training data obtains accuracy of the best training with dimensions 7x4 rows and columns of 81,1966%. Moreover, the results of trials testing with 104 letters image obtain an accuracy of 79,81% with 83 correct answers and 21 incorrect answers. While on trial with the input of a word and a sentence image composes 173 letters without ROI obtains an accuracy of 27.75% with the letters that can be recognized correctly by 48 letters and the number of letters that cannot be recognized as much as 125 letters. However, with the similar input and additional ROI obtained better accuracy is 63.01% with 109 letters that successfully detected properly and the 64 letters that cannot be recognized correctly.

## الملخص

موروتي. 2014. التعريف من الصورالوسائل الأبعدية من الخط بلستعمال الطريقة النائف بايز  
كلاسيفير Naïve Bayes Classifier. أطروحة. قسم المعلوماتية. كلية العلوم والتكنولوجيا.  
الجامعة الإسلامية الحكومية مولانا مالك إبراهيم مالانج . المشرف: إيروان بودي سانتوسو الماجستير  
و الدكتور م. أمين حريدي الماجستير،

### الكلمات الرئيسية: التعرف الخط، النائف بايز كلاسيفير Naïve Bayes Classifier

التطور السريع للتكنولوجيا لا تجعل حاليا توثيق البيانات / المعلومات مع الخط القضاء عليها . حتى  
في مجموعة متنوعة من البحث، والكتابة باليد أثبتت أن يكون لها تأثير جيد جدا على معلومات  
استخباراتية . في هذه الدراسة سيتم بناء تطبيق التعرف على الخط باستخدام النائف بايز كلاسيفير  
Naïve Bayes Classifier مع بيانات التدريب مع البيانات 468 التي تكون كل حرف من 18  
عينة التي تستخدم لتدريب النظام . في التعرف على الأنماط ، بيانات التدريب له دور مهم جدا في  
تحديد دقة النظام، بالإضافة إلى حجم أبعاد بكسل من الصورة من الصفوف والأعمدة هي أيضا  
مؤثرة جدا . من نتائج الاختبار من البيانات التي تم الحصول عليها التدريب أفضل التدريب مع  
الصفوف أبعاد وأعمدة 9x7 يعني 81,1966% . وبناء على نتائج التجارب اختبار المدخلات في  
شكل بيانات الصورة المكتسبة دقة يعني 104 البيانات الخطابات 79,81% مع 83 الإجابات  
الصحيحة و 21 إجابات خطأ . بينما يحاكم مع مدخلات من الصورة للكلمة والجملة المكونة من  
173 حروف بدون ROI الحصول عليها دقة 27,75% مع عدد من الإجابات الصحيحة 48  
وإجابات خاطئة ما يصل الى 125 حروف، في حين أن نفس المدخلات ودقة الحصول على  
ROI إضافي أفضل هو 63,01% مع 109 حروف تم الكشف بنجاح صحيحة و 64 حروف  
التي لا يمكن ان يعرف صحيحة

## الملخص

موروتي. ٢٠١٤. التعريف من الصورالرسائل الأبجدية من الخط باستعمال الطريقة النائف بايز  
كلاسيفير Naïve Bayes Classifier. أطروحة. قسم المعلوماتية. كلية العلوم والتكنولوجيا.  
الجامعة الإسلامية الحكومية مولانا مالك إبراهيم مالانج. المشرف: إيروان بودي سانتوسو الماجستير  
و الدكتور م. أمين حريدي الماجستير،

الكلمات الرئيسية: التعرف الخط، النائف بايز كلاسيفير Naïve Bayes Classifier

التطور السريع للتكنولوجيا لا تجعل حاليا توثيق البيانات / المعلومات مع الخط القضاء عليها. حتى  
في مجموعة متنوعة من البحث، والكتابة باليد أثبتت أن يكون لها تأثير جيد جدا على معلومات  
استخباراتية. في هذه الدراسة سيتم بناء تطبيق التعرف على الخط باستخدام النائف بايز كلاسيفير  
Naïve Bayes Classifier مع بيانات التدريب مع البيانات ٤٦٨ التي تكون كل حرف من ١٨  
عينة التي تستخدم لتدريب النظام. في التعرف على الأنماط، بيانات التدريب له دور مهم جدا في  
تحديد دقة النظام، بالإضافة إلى حجم أبعاد بكسل من الصورة من الصفوف والأعمدة هي أيضا  
مؤثرة جدا. من نتائج الاختبار من البيانات التي تم الحصول عليها التدريب أفضل التدريب مع  
الصفوف أبعاد وأعمدة ٩X٧ يعني ٨١،١٩٦٦%. وبناء على نتائج التجارب اختبار المدخلات في  
شكل بيانات الصورة المكتسبة دقة يعني ١٠٤ البيانات الخطابات ٧٩،٨١% مع ٨٣ الإجابات  
الصحيحة و ٢١ إجابات خطأ. بينما يحاكم مع مدخلات من الصورة للكلمة والجملة المكونة من  
١٧٣ حروف بدون ROI الحصول عليها دقة ٢٧،٧٥% مع عدد من الإجابات الصحيحة ٤٨  
وإجابات خاطئة ما يصل الى ١٢٥ حروف، في حين أن نفس المدخلات ودقة الحصول على  
ROI إضافي أفضل هو ٦٣،٠١% مع ١٠٩ حروف تم الكشف بنجاح صحيحة و ٦٤ حروف  
التي لا يمكن ان يعرف صحيحة

## ABSTRACT

Morwati. 2014. **Handwriting Recognition with Naïve Bayes Classifier Method.**

Theses, Infomatic Engineering Programme Faculty of Science and Technology the State of Islamic University Maulana Malik Ibrahim Malang. Promotor: (I) Irwan Budi Santoso, M.Kom (II) Dr. M. Amin Haryadi, M.T.

**Keywords:** *Handwriting Recognition, Naïve Bayes Classifier.*

The currently rapid development of technology does not eliminate documentation of data / information by the handwriting. Even in the variety of researches, handwriting is to be proved as having a very good effect on human's intelligence. The application of handwriting within *Naïve Bayes Classifier* method with 468 training data which each letter is composed by 18 samples will be analyzed in this research). In the pattern's recognition, training data is the most important role in determining the accuracy of the system. Also, the size of the pixel dimensions of the image is highly influenced. From the test results of the training data obtains accuracy of the best training with dimensions 7x4 rows and columns of 81,1966%. Moreover, the results of trials testing with 104 letters image obtain an accuracy of 79,81% with 83 correct answers and 21 incorrect answers. While on trial with the input of a word and a sentence image composes 173 letters without ROI obtains an accuracy of 27.75% with the letters that can be recognized correctly by 48 letters and the number of letters that cannot be recognized as much as 125 letters. However, with the similar input and additional ROI obtained better accuracy is 63.01% with 109 letters that successfully detected properly and the 64 letters that cannot be recognized correctly.

## ABSTRAK

Morwati. 2014. **Pengenalan Citra Huruf Alphabet Tulisan Tangan Menggunakan Metode *Naïve Bayes Classifier***. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Irwan Budi Santoso, M.Kom (II) Dr. M. Amin Haryadi, M.T.

**Kata Kunci:** *Pengenalan Tulisan Tangan, Naïve Bayes Classifier.*

Perkembangan teknologi yang sangat pesat saat ini tidak membuat pendokumentasian data/informasi dengan tulisan tangan dihilangkan. Bahkan dalam berbagai penelitian, menulis dengan tangan terbukti memiliki efek yang sangat baik untuk kecerdasan otak. Dalam penelitian ini akan dibangun aplikasi pengenalan tulisan tangan menggunakan metode *Naïve Bayes Classifier* dengan data training sebanyak 468 data dengan masing – masing huruf terdiri dari 18 sampel yang digunakan untuk pelatihan sistem. Dalam pengenalan pola, data training memiliki peran yang sangat penting dalam menentukan akurasi sistem, selain itu ukuran dimensi pixel baris dan kolom citra juga sangat berpengaruh. Dari hasil uji coba terhadap data training diperoleh akurasi training terbaik dengan dimensi baris dan kolom 7x4 sebesar 81,1966%. Dan berdasarkan hasil uji coba testing dengan inputan berupa citra huruf sebanyak 104 data diperoleh akurasi sebesar 79,81% dengan 83 jawaban benar dan 21 jawaban salah. Sedangkan pada uji coba dengan input berupa kata dan kalimat yang terdiri dari 173 huruf dan tanpa ROI diperoleh akurasi 27,75% dengan jumlah jawaban benar sebanyak 48 dan jawaban salah sebanyak 125 huruf, sedangkan dengan inputan yang sama dan dengan tambahan ROI diperoleh akurasi yang lebih baik yaitu 63,01% dengan 109 huruf yang berhasil dikenali dengan benar dan 64 huruf yang tidak dapat dikenali dengan benar.

# BAB I

## PENDAHULUAN

### 1.1.Latar Belakang Masalah

Dewasa ini, perkembangan teknologi telah berkembang sangat pesat di berbagai bidang, salah satunya di bidang komputerisasi. Saat ini komputer mulai dilatih untuk dapat mengenali pola yang di inputkan untuk mengolah data maupun dalam pengambilan keputusan. Perkembangan komputer juga terlihat dari bentuk komputer yang semakin lama semakin praktis, dari seperangkat komputer yang terdiri dari CPU dan monitor, laptop, notebook, tablet hingga smartphone. Bahkan beberapa perangkat elektronik layar sentuh mulai dapat menerima dan mengenali inputan berupa tulisan tangan. Namun hal tersebut tidak membuat dokumentasi data dengan tulisan tangan pada media kertas ditinggalkan, mendokumentasikan data dengan tulisan tangan pada media kertas masih sering dilakukan di berbagai hal, misalnya dalam bidang perkantoran dan perkuliahan.

Bahkan dalam sebuah penelitian yang dipublikasikan pada *Advances in Haptics journal* dijelaskan bahwa anak-anak dan pelajar yang menulis dengan tangan ternyata lebih cepat belajar daripada anak-anak yang mengetik di komputer. Hal ini karena, saat membaca dan menulis, anak-anak melibatkan berbagai indera. Untuk mengenali huruf-huruf, seseorang akan melibatkan bagian otak yang bernama sensorimotor. Penelitian ini dilakukan oleh Profesor Anne Mangen dari Stavanger University Norwegia dan Jean-Luc Velay dari Marseille University (Darmawan, 2011).

Penelitian serupa juga dilakukan psikolog dari Princeton dan Universitas California, Los Angeles, Pam Mueller dan Daniel Oppenheimer. Mereka menguji efek menulis catatan pada mahasiswa dalam dua seri percobaan. Hasil studi menunjukkan, mahasiswa yang menggunakan laptop lebih “miskin” soal ide. Penelitian tersebut menunjukkan bahwa menulis dapat menyimpan dan mengendapkan ide dalam kurun waktu yang lebih lama dibandingkan dengan mengetik (Sulistyawati, 2014).

Berdasarkan studi pada tahun 2008 ditemukan bahwa orang dewasa dapat lebih mudah mengenali karakter-karakter huruf baru seperti tulisan Cina, simbol-simbol matematika atau not musik yang ditulis dengan tangan ketimbang karakter yang dibuat dengan komputer. Menulis dengan tangan juga merupakan olahraga bagi otak yang merupakan latihan kognitif yang baik bagi lansia yang ingin otaknya tetap tajam meskipun semakin menua (Ilham, 2013).

Selain itu menulis juga terbukti memiliki peran yang sangat penting dalam penyebaran dan perkembangan peradaban Islam, menulis juga telah menjadi tradisi turun menurun dalam sejarah umat Islam. Dengan tradisi membaca dan menulis, umat Islam dapat mencapai puncak peradabannya. Menulis juga telah menjadi kebiasaan ulama dan intelektual muslim tempo dulu sehingga dapat mengikat dan menyebarkan ilmunya, sebagaimana ungkapan “Ikatlah ilmu dengan menuliskannya”. Di dalam *al-Qur'an* pun banyak ayat – ayat yang mengutarakan penghargaan yang tinggi terhadap huruf, pena dan tulisan. Salah satunya dalam wahyu pertama yang diturunkan kepada nabi Muhammad *shallallaahu 'alaihi wa sallam* dalam surat *al – 'Alaq* ayat 1-5.

أَقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ۝ خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ ۝ أَلَمْ يَكُنْ الْأَكْرَمُ  
الَّذِي عَلَّمَ بِالْقَلَمِ ۝ عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمْ ۝

Artinya : Bacalah dengan (menyebut) nama Tuhanmu yang Menciptakan, Dia Telah menciptakan manusia dari segumpal darah. Bacalah, dan Tuhanmulah yang Maha pemurah, yang mengajar (manusia) dengan perantaraan kalam, Dia mengajar kepada manusia apa yang tidak diketahuinya (QS *al – Alaq* : 1 – 5 ).

Surat *al – ‘Alaq* ayat 1-5 berisi penegasan tentang keutamaan membaca (iqra’) dan menulis (‘allama bil qalam) yang maksudnya, Allah mengajar manusia dengan perantaraan baca dan tulis. Imam Ali bin Abi Thalib *radhiyallahu ta’ala ‘anhu* juga pernah berkata “ikatlah ilmu dengan menuliskannya”.

Pentingnya menulis juga ditegaskan Allah SWT dalam ayat pertama surat *al – Qalam*.

ن وَالْقَلَمِ وَمَا يَسْطُرُونَ ۝

Artinya : Nun, demi kalam dan apa yang mereka tulis.

Surat *al – Qalam* adalah surat ke-2 yang turun setelah surat *al – Alaq* yang berisi perintah tentang membaca dan menulis. Dan kedua surat paling awal ini sama – sama menyinggung peranan pena sebagai alat belajar mengajar. Bahkan surat ini diberi nama *al – Qalam* yang berarti pena. Dalam ayat pertama surat Al-Qalam tersebut, Allah SWT bersumpah dengan dua hal, yakni kalam/pena dan apa yang ditulis. Dalam Tafsir Departemen Agama (Depag) sumpah dapat memiliki arti bahwa yang dipakai bersumpah itu adalah sesuatu yang mulia, bernilai, bermanfaat, dan berharga. Dengan kitab dan kalam itulah

kemajuan dan kemunduran suatu bangsa diukur. Sumpah Allah itu untuk menarik perhatian kita kepada kalam dan kitab. Dan yang dimaksud dengan kalam adalah semua alat yang dipergunakan untuk mewujudkan tulisan. Hal ini menjelaskan bahwa dua surat paling awal tersebut mendorong manusia untuk belajar menulis dan membaca.

Dalam salah satu riwayat, Abdullah bin ‘Amr bin ‘Ash berkata, “Dulu aku menulis semua perkara yang aku dengar dari Rasulullah SAW untuk aku hafalkan. Namun, orang-orang Quraisy melarangku dan bertanya, “Kamu menulis semua yang kamu dengar dari Rasulullah *shallallaahu ‘alaihi wa sallam*. Beliau adalah manusia yang berbicara ketika senang dan ketika marah?” Aku pun berhenti menulis. Lalu, aku menceritakannya kepada Rasulullah *shallallaahu ‘alaihi wa sallam*. Beliau memberi isyarat ke mulutnya dengan jarinya seraya bersabda, “Tulislah! Demi Dzāt yang jiwaku ada di tangan-Nya, tidak keluar dari mulutku kecuali kebenaran”.

Dari Abu Hurairah, dia mengatakan: “Tidak ada seorang pun dari sahabat Nabi *shallallaahu ‘alaihi wa sallam* yang memiliki hadits lebih banyak daripadaku, kecuali apa yang ada pada ‘Abdullah bin ‘Amr, karena sesungguhnya dia menulis, sementara aku tidak menulis” (Al-Albani, 2010).

Para sahabat adalah teladan bagi kita untuk tidak meninggalkan menulis. Aktivitas menulis para sahabat dilakukan untuk mencatat kejadian – kejadian penting pada zaman Rasulullah *shallallaahu ‘alaihi wa sallam* serta perkataan dan perbuatan beliau. Dalam pemeliharaan wahyu pun menulis tetap dilakukan,

kekuatan hafalan para sahabat tidak menafikan pencatatan wahyu yang diterima Rasulullah *shallallaahu 'alaihi wa sallam*. Pentingnya menulis pada awal perkembangan Islam tidak hanya untuk menuliskan wahyu melainkan juga untuk menulis naskah perjanjian, surat – surat dan naskah lainnya. Penulisan dan pengiriman surat kepada para kepala negara sebagai ajakan untuk masuk Islam merupakan bersejarah yang berperan penting dalam penyebaran dan perkembangan agama Islam. Para ulama dan intelektual zaman dulu juga sangat aktif menyalurkan ilmu dan karya – karyanya dengan menulis, diantaranya Imam Syafi'i dengan karyanya yang mencapai 174 kitab, Imam Ahmad bin Hanbal yang telah menerbitkan kitab pertamanya sejak usia 18 tahun, Muhammad bin Jarir ath – Thabari yang konon dapat menulis sebanyak 40 lembar setiap harinya, Ibnu Taimiyah yang karyanya mencapai 500 judul, dan masih banyak lagi ulama – ulama dan intelektual muslim yang terkenal dengan karya – karya tulisannya yang mendunia (Sugiantoro, 2014).

Tradisi menulis dalam Islam memiliki berbagai tujuan. Pertama, menulis untuk mengikat ilmu. Setiap ilmu yang didapatkan akan terekam kuat jika dituliskan. Ilmu adalah sesuatu yang berharga sehingga perlu menjaganya dengan menuliskannya. Kedua, menyampaikan ilmu. Ilmu yang didapatkan tentu perlu disebarluaskan untuk kemaslahatan bersama. Tidak hanya belajar, tapi juga mengajarkan ilmu dengan menyebarkan lewat tulisan seperti yang dicontohkan dalam sebuah hadits Dari Abdullah bin Amr *radhiyallahu ta'ala 'anhu*, bahwa nabi Muhammad *shallallaahu 'alaihi wa sallam* bersabda, “Sampaikanlah dariku walau hanya satu ayat” (HR. Bukhari) . Ketiga, menyeru

kepada kebaikan (amar ma'ruf). Menulis untuk mengajak manusia menuju pada kebaikan. Keempat, mencegah kemungkaran (nahi munkar). Tulisan-tulisan yang dihasilkan berawal dari kegelisahan untuk menentang kezaliman dan berkata tidak pada setiap kemungkaran. Kelima, meneguhkan keimanan manusia. Menulis untuk mengarahkan kehidupan pada kesadaran ketuhanan (Sugiantoro, 2014).

Dari berbagai hasil penelitian dan penjelasan tersebut, menulis dengan tangan terbukti memiliki efek yang lebih baik dari pada mengetik. Namun dengan banyaknya dokumen yang ditulis dengan tangan pada media kertas menimbulkan sebuah permasalahan dalam pemrosesannya karena untuk dapat memproses data yang ditulis pada media kertas dengan komputer data harus diketik kembali dalam bentuk teks digital, hal ini sangat tidak efisien waktu dan tenaga karena harus melakukan pekerjaan yang sama berulang kali.

Untuk mengatasi masalah tersebut dibutuhkan sebuah aplikasi yang dapat mengenali karakter tulisan tangan sehingga pemrosesan data teks tulisan tangan dapat dilakukan dengan lebih cepat dan efisien tanpa harus mengetik kembali data yang ditulis dengan tulisan tangan. Oleh karena itu, dalam penelitian ini akan dibangun sebuah aplikasi yang dapat mengenali teks tulisan tangan dengan inputan berupa citra tulisan tangan yang diperoleh menggunakan scanner dengan output berupa teks digital dari citra yang diinputkan.

## 1.2.Rumusan Masalah

Dari latar belakang tersebut, dapat diidentifikasi beberapa masalah sebagai berikut.

- a. Bagaimana membangun aplikasi untuk mengenali tulisan tangan menggunakan metode *Naïve Bayes Classifier*?
- b. Berapa akurasi metode *Naïve Bayes Classifier* dalam mengenali tulisan tangan?

## 1.3.Tujuan Penelitian

Tujuan dari penelitian ini yaitu :

- a. Membangun aplikasi pengenalan tulisan tangan menggunakan metode *Naïve Bayes Classifier*.
- b. Mengukur akurasi metode *Naïve Bayes Classifier* dalam mengenali tulisan tangan.

## 1.4.Manfaat Penelitian

Manfaat yang diharapkan dengan adanya penelitian ini yaitu, dapat membantu mempermudah pengguna untuk mengubah tulisan tangan yang ditulis pada media kertas menjadi teks digital. Sehingga efek baik dari menulis dengan tangan tetap dapat diperoleh tanpa harus mengetik kembali dokumen yang ditulis dengan tangan.

### 1.5. Batasan Masalah

Batasan masalah dalam penelitian ini sebagai berikut:

- a. Proses pengambilan citra tulisan tangan dilakukan menggunakan scanner dengan hasil citra yang berformat JPEG.
- b. Kertas yang digunakan untuk penulisan teks tulisan tangan berupa kertas HVS yang berwarna putih.
- c. Citra tulisan tangan yang digunakan sebagai *input* berupa citra tulisan tangan yang menggunakan alphabet bahasa indonesia atau bahasa inggris dan berhuruf kecil (huruf a-z).
- d. Jenis tulisan tangan yang dapat dikenali berupa tulisan tegak lurus (tidak bersambung) baik dalam bentuk huruf tunggal maupun kata atau satu baris kalimat.

### 1.6. Sistematika Penulisan

Adapun sistematika penulisan dalam penelitian ini terdiri dari beberapa bab sebagai berikut :

#### BAB I PENDAHULUAN

Bab pendahuluan merupakan bab pertama dalam sistematika penulisan penelitian ini yang membahas tentang hal – hal yang melatarbelakangi dilakukannya penelitian ini, untuk apa penelitian ini, serta manfaat yang diharapkan dari penelitian ini. Oleh karena itu, bab ini terdiri atas : latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan.

## BAB II TINJAUAN PUSTAKA

Pada bab ini akan diuraikan teori – teori serta argumentasi ilmiah yang berhubungan dengan pengenalan pola dan metode yang digunakan dalam penelitian ini serta bahasan mengenai penelitian – penelitian terdahulu yang berhubungan dengan penelitian ini. Bahan pustaka yang digunakan diperoleh dari berbagai sumber, seperti : jurnal penelitian, buku, serta beberapa situs internet.

## BAB III METODE PENELITIAN

Pada bab ini akan dijabarkan tahapan yang dilakukan dalam penelitian ini, serta desain interface dan desain sistem yang dilengkapi dengan blok diagram dan flowchart.

## BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas tentang metode dalam menyelesaikan masalah yang diimplementasikan dalam kode – kode program aplikasi yang dibuat, serta uji coba sistem untuk mengetahui apakah sistem telah berjalan sesuai harapan.

## BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari seluruh rangkaian penelitian yang telah dilakukan serta saran untuk pengembangan selanjutnya.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Kajian Pustaka**

Pada sub bab ini akan dijabarkan teori – teori dan argumentasi ilmiah yang terkait dengan pengenalan pola serta metode yang akan digunakan untuk menyelesaikan masalah dalam penelitian ini.

##### **2.1.1. Pengenalan Pola**

Pola adalah suatu entitas yang terdefinisi dan dapat diidentifikasi serta diberi nama. Pola bisa merupakan kumpulan hasil pengukuran atau pemantauan dan bisa dinyatakan dalam notasi vektor atau matriks. Secara umum pengenalan pola (pattern recognition) adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif fitur (ciri) atau sifat utama dari suatu obyek (Putra, 2010). Ciri – ciri tersebut digunakan untuk membedakan suatu pola dengan pola lainnya. Ciri yang baik adalah ciri yang memiliki daya pembeda yang tinggi sehingga pengenalan pola berdasarkan ciri yang dimiliki dapat dilakukan dengan baik. Pengenalan pola ini bersifat *conceptually driven processing* yang berarti bahwa proses dimulai dari pembentukan konsep pada objek yang dijumpai. Ada banyak aplikasi yang bisa dijadikan implementasi dari pengenalan pola, di antaranya adalah pengenalan wajah pada manusia, pengenalan gambar mata, pengenalan penyakit berdasarkan gejala – gejala yang ditemukan pada objek dan lain – lain. Pengenalan pola bisa

dibuat melalui pendekatan pemrosesan citra yang mana tujuan akhir dari pemrosesan citra tersebut digunakan untuk pengelompokan objek sehingga menghasilkan output yang diinginkan. Salah satu contoh implementasi dari pengenalan pola adalah pengenalan tulisan tangan (Hastiana, 2010).

Pengenalan tulisan tangan (*handwriting recognition*) adalah kemampuan komputer untuk menerima dan menafsirkan *input* tulisan tangan yang dapat dimengerti dari sumber seperti dokumen kertas, foto, layar sentuh dan perangkat lainnya. Dalam penelitian ini, untuk mengenali pola tulisan tangan digunakan metode *Naïve Bayes Classifier* (NBC), metode ini memiliki beberapa kelebihan diantaranya yaitu, sederhana, cepat dan berakurasi tinggi (Uluwiyah *et al*, 2013).

### **2.1.2. Definisi dan Jenis – Jenis Huruf**

Dalam Kamus Besar Bahasa Indonesia (KBBI) huruf didefinisikan sebagai tanda aksara dalam tata tulis yg merupakan anggota abjad yang melambangkan bunyi bahasa. Berdasarkan pengertian tersebut, dapat kita katakan kalau huruf adalah lambang dari bunyi. Misalnya bunyi *be* lambangnya atau hurufnya adalah *b*, bunyi *el* lambangnya adalah *l*, dan seterusnya. Secara umum bunyi huruf dalam bahasa Indonesia dibedakan atas vokal, konsonan, diftong, dan konsonan rangkap (Godam, 2006).

#### **a. Huruf Vokal**

Huruf Vokal atau huruf hidup adalah bunyi ujaran akibat adanya udara yang keluar dari paru-paru tidak terkena hambatan atau halangan. Jumlah huruf vokal ada 5, yaitu a, i, u, e, dan o. Bunyi vokal dibedakan berdasarkan posisi tinggi rendahnya lidah, bagian lidah yang bergerak, struktur, dan bentuk

bibir. Dengan demikian, bunyi vokal tidak dibedakan berdasarkan posisi artikulatornya karena pada bunyi vokal tidak terdapat artikulasi. Artikulator adalah bagian alat ucap yang dapat bergerak.

#### **b. Huruf Konsonan**

Huruf Konsonan atau huruf mati adalah bunyi ujaran akibat adanya udara yang keluar dari paru-paru mendapatkan hambatan atau halangan. Jumlah huruf konsonan ada 21 buah, yaitu b, c, d, f, g, h, j, k, l, m, n, p, q, r, s, t, v, w, x, y, dan z. Huruf konsonan dibedakan menurut cara hambat (cara artikulasi) atau cara pengucapannya, tempat hambat (tempat artikulasi), hubungan posisional antara penghambat-penghambat atau hubungan antara artikulator pasif dan bergetar tidaknya pita suara.

#### **c. Huruf Diftong**

Huruf diftong atau huruf vokal rangkap adalah gabungan dua buah huruf vokal yang menghasilkan bunyi rangkap. Dalam Bahasa Indonesia huruf diftong berbentuk ai, au, dan oi. Contoh : Bangau, Pakai, Sengau, Perangai dan lain – lain.

#### **d. Huruf Konsonan Rangkap**

Huruf konsonan rangkap adalah gabungan dua huruf konsonan ada 4 buah dalam bahasa indonesia, yaitu : kh, ng, ny, dan sy. Contohnya : nyamuk, syarat, kumbang, khawatir dan lain – lain.

### **2.1.3. Definisi Citra**

Citra adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis, citra adalah sebuah fungsi intensitas cahaya dua

dimensi  $f(x,y)$  dimana  $x$  adalah posisi baris dan  $y$  adalah posisi kolom sedangkan  $f$  adalah fungsi intensitas atau kecerahan dari citra pada koordinat  $(x,y)$ . Pada umumnya citra berbentuk empat persegi panjang dan dimensi ukurannya dinyatakan sebagai (tinggi x lebar). Citra dengan tinggi  $N$  piksel, lebarnya  $M$  piksel, dan memiliki intensitas  $f$  dapat dipresentasikan sebagai suatu matriks yang berukuran  $N$  baris dan  $M$  kolom seperti yang ditunjukkan pada Persamaan 2.1.

$$f = [f(x,y)] = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix} \quad (2.1)$$

Indeks baris ( $x$ ) dan indeks kolom ( $y$ ) menyatakan suatu koordinat titik pada citra. Masing-masing titik  $(x,y)$  di citra disebut *picture element* atau pixel, sedangkan  $f(x,y)$  merupakan intensitas (derajat keabuan) pada titik  $(x,y)$  (Munir, 2004).

#### 2.1.4. JPEG

JPEG (Joint Photographic Experts Group) adalah standar kompresi file yang dikembangkan oleh Joint Photographic Experts Group menggunakan kombinasi DCT dan pengkodean Huffman untuk mengkompresikan suatu file citra. JPEG adalah suatu algoritma kompresi yang bersifat lossy. JPEG juga mampu menayangkan warna dengan kedalaman 24-bit true color. Dan umumnya digunakan untuk menyimpan gambar-gambar hasil foto. JPEG adalah teknik kompresi grafis high color bit-mapped. Merupakan teknik dan standar universal untuk kompresi dan dekompresi citra tidak bergerak untuk digunakan pada kamera digital dan system pencitraan menggunakan komputer yang

dikembangkan oleh Joint Photographic Experts Group. Umumnya digunakan untuk kompresi citra berwarna maupun grayscale. Format file ini mampu mengompres objek dengan tingkat kualitas sesuai dengan pilihan yang disediakan. Format file sering dimanfaatkan untuk menyimpan gambar yang akan digunakan untuk keperluan halaman web, multimedia, dan publikasi elektronik lainnya. Format file ini mampu menyimpan gambar dengan mode warna RGB, CMYK, dan Grayscale. Format file ini juga mampu menyimpan alpha channel, namun karena orientasinya ke publikasi elektronik maka format ini berukuran relatif lebih kecil dibandingkan dengan format file lainnya (Nurchahyo, 2010).

#### 2.1.5. Grayscale

Citra *grayscale* merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pixelnya, dengan kata lain nilai bagian  $RED = GREEN = BLUE$ . Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih. Tingkat keabuan merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih (Putra, 2010). Untuk mengubah citra berwarna menjadi citra *grayscale* digunakan rumus dalam Persamaan 2.2.

$$Grayscale = 0,299R + 0,587G + 0,114B \quad (2.2)$$

*Grayscale* (skala keabuan) merupakan suatu istilah untuk menyebutkan satu citra yang memiliki warna putih, abu-abu dan hitam. Format citra ini disebut skala keabuan karena pada umumnya warna yang dipakai adalah antara hitam sebagai warna minimal dan warna putih sebagai warna maksimalnya, sehingga warna antaranya adalah abu-abu. Citra skala keabuan memberi kemungkinan

warna yang lebih banyak daripada citra biner, karena ada nilai-nilai lain diantara nilai minimum (biasanya = 0) dan nilai maksimumnya (Hastiana, 2010).

Jumlah warna pada citra *gray* adalah 256, karena citra *gray* jumlah bitnya adalah 8, sehingga jumlah warnanya adalah  $2^8 = 256$ , nilainya berada pada jangkauan 0 – 255. Sehingga nilai intensitas dari citra *gray* tidak akan melebihi 255 dan tidak mungkin kurang dari 0. Model penyimpanannya adalah  $f(x,y) =$  nilai intensitas, dengan  $x$  dan  $y$  merupakan pisisi nilai intensitas. Misalkan suatu citra dengan ukuran lebar = 512 dan tinggi = 512, maka jumlah byte yang diperlukan untuk penyimpanan citra (Purnomo *et al*, 2010).

$$\begin{aligned} \text{grayscale} &= 512 \times 512 \times 1 \\ &= 262,144 \text{ byte} \\ &= 0,262 \text{ MB} \end{aligned}$$

#### 2.1.6. *Projection Profile*

*Projection Profile* terdiri atas dua bagian, yakni *vertical* dan *horizontal*. Tujuannya adalah untuk memisahkan karakter untuk tiap baris dan tiap kolom secara otomatis dan akurat. Cara kerja *Projection Profile* yaitu dengan menjumlahkan nilai intensitas citra yang telah diubah dalam bentuk biner pada setiap baris (untuk *vertical*) dan kolomnya (untuk *horizontal*). Yang menjadi pemisahannya adalah adanya lembah untuk tiap bentuk isyarat yang merupakan spasi putih yang memisahkann setiap karakter. Metode ini baik untuk karakter yang saling terpisah (Hendry, 2011). *Projection Profile* merupakan struktur data yang digunakan untuk menyimpan jumlah pixel objek ketika gambar diproyeksikan di atas sumbu normal  $XY$  seperti yang ditunjukkan dalam

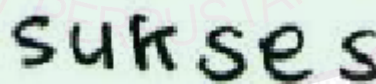
Persamaan 2.3. Setiap sel dari vektor proyeksi dikaitkan dengan jumlah piksel di atas ambang yang telah ditetapkan (biasanya warna latar belakang) seperti yang ditunjukkan pada Persamaan 2.4 dan Persamaan 2.5.

$$X, Y \rightarrow M(x, y) \quad (2.3)$$

$$X_n = \sum_{i=0}^h Y_i, n \in [0, v] \quad (2.4)$$

$$Y_n = \sum_{i=0}^v X_i, n \in [0, h] \quad (2.5)$$

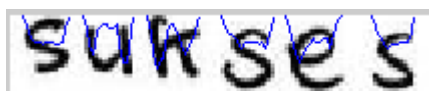
Dimana  $X$  dan  $Y$  merupakan sumbu *horizontal* dan *vertical*,  $h$  merupakan ketinggian gambar (ukuran *vertical*) untuk  $X$  atau lebar (ukuran *horizontal*) untuk  $Y$  dan  $v$  merupakan ukuran gambar (Roberto *et al*, 2002). Gambar 2.1 menunjukkan contoh citra hasil *Projection Profile*. Gambar 2.1(a) menunjukkan citra asli, Gambar 2.1(b) menunjukkan citra hasil *vertical projection* dan Gambar 2.1(c) menunjukkan citra hasil *horizontal projection*.



(a) Citra asli



(b) Citra hasil *vertical projection*



(c) Citra hasil *horizontal projection*

**Gambar 2.1.** Citra hasil *Projection Profile*

### 2.1.7. Cropping

*Cropping* pada pengolahan citra berarti memotong suatu bagian dari citra pada koordinat tertentu sehingga diperoleh citra yang diharapkan. Yang diperlukan dalam *cropping* suatu citra adalah menentukan koordinat dari citra yang akan di-*crop*. Kemudian menentukan batas dari citra yang akan di-*crop* (Fifin *et al*, 2012). Gambar 2.2 menunjukkan sebuah citra yang pada awalnya membentuk sebuah kata “nama”, setelah dilakukan proses *cropping* dengan mengambil koordinat tertentu maka diperoleh citra baru hasil *cropping* yang menunjukkan satu huruf yaitu huruf “n”. Gambar 2.2(a) menunjukkan citra asli sebelum melalui proses *cropping* dan Gambar 2.2(b) menunjukkan citra hasil *cropping*.



(a) Citra asli

(b) Citra hasil *cropping*

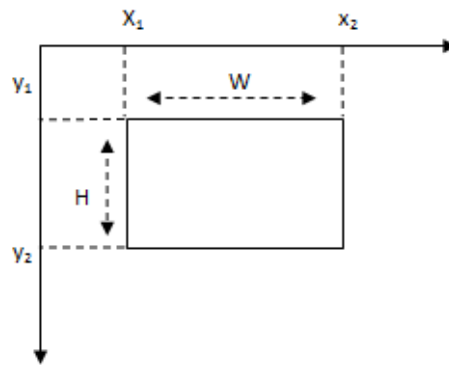
**Gambar 2.2.** Sebuah citra di *cropping*

Gambar 2.3 menunjukkan sebuah citra yang di-*cropping* sebesar  $W \times H$ . Titik  $(x_1, y_1)$  dan  $(x_2, y_2)$  adalah koordinat titik pojok kiri atas dan pojok kanan bawah citra yang akan di-*crop*. Rumus yang digunakan untuk *cropping* ditunjukkan pada Persamaan 2.6 dan Persamaan 2.7.

$$W = x_2 - x_1 \quad (2.6)$$

dan

$$H = y_2 - y_1 \quad (2.7)$$



**Gambar 2.3.** Sebuah citra di *cropping* sebesar  $W \times H$   
 Sumber : Sutoyo *et al*, 2009

### 2.1.8. Penskalaan (*scalling*)

Operasi penskalaan (*scalling*) dimaksudkan untuk memperbesar (*zoom-in*) atau memperkecil (*zoom-out*) citra sesuai dengan faktor skala  $K$  yang diinginkan. Pada prinsipnya, operasi penskalaan menggandakan jumlah piksel sebesar  $K$  kali semula bila  $K > 1$  dan  $1/K$  kali semula bila  $0 < K < 1$  dalam arah *vertical* dan *horizontal* (Sutoyo *et al*, 2009).

Dalam penskalaan ada dua cara yang digunakan, yaitu replikasi dan interpolasi. Replikasi bekerja dengan cara menggandakan piksel sejumlah faktor skala  $K$ . Interpolasi bekerja dengan cara memperhalus tingkat gradasi dan intensitas citra yang berdekatan selebar faktor skala  $K$ . Secara default operasi penskalaan (*imresize*) dalam matlab menggunakan pendekatan interpolasi *nearest neighbor* untuk menentukan nilai pixel citra. Gambar 2.4(a) menunjukkan contoh citra asli sebelum melalui proses *scalling* dengan dimensi pixel  $30 \times 30$  dan Gambar 24(b) menunjukkan citra hasil *scalling* dengan dimensi citra  $20 \times 20$ .



(a) Citra ukuran 30x30      (b) citra ukuran 20x20

**Gambar 2.4.** *Scalling* citra huruf 'a'

### 2.1.9. Ekstraksi Fitur

Ekstraksi fitur atau ekstraksi ciri dalam pengolahan citra berarti mengubah nilai-nilai intensitas koordinat piksel yang terdapat dalam citra menjadi susunan kode-kode nilai pada setiap pixel (Fifin *et al*, 2012). Dalam penelitian ini, matrik pixel citra akan diubah dalam bentuk vektor. Proses ini nantinya akan dilakukan pada data *training* maupun data *testing*. *Row vector* (vektor baris) adalah array berdimensi  $1 \times N$ . Elemen vektor yang diakses menggunakan pengindeksan satu dimensi. Maka,  $v(1)$  adalah elemen pertama vektor  $v$ , dan  $v(2)$  adalah elemen kedua dan seterusnya. Elemen vektor dalam Matlab dibatasi oleh kurung siku dan dipisahkan dengan spasi atau koma (Prasetyo, 2011). Contoh cara pembentukan vektor dan cara mengakses elemen vektor ditunjukkan pada Gambar 2.5.

```
>> v = [2 4 6 8]
v = 2 4 6 8

>> v(2)
Ans =
4
```

**Gambar 2.5.** Contoh vektor

Vektor baris dapat dikonversi menjadi vektor kolom dengan melakukan *transpose operator* (') seperti yang ditunjukkan pada Gambar 2.6.

```
>> w = v.'
W =
     2
     4
     6
     8
```

**Gambar 2.6.** Contoh transpose vektor

### 2.1.10. Distribusi Normal

Distribusi normal disebut pula distribusi Gauss, yaitu distribusi probabilitas yang paling banyak digunakan dalam berbagai analisis statistika. Distribusi ini juga dijuluki kurva lonceng (*bell curve*) karena grafik fungsi kepekatan probabilitasnya mirip dengan bentuk lonceng. Pada distribusi ini dikarakteristikan pada dua parameter input yaitu parameter mean ( $\mu$ ) dan parameter varian ( $\sigma^2$ ) untuk setiap kelas fitur yang di cari dan peluang kelas bersyarat, kemudian mencari peluang terbesar (Uluwiyah *et al*, 2013). Sehingga pernyataan tersebut dapat dinyatakan dalam persamaan 2.8.

$$P(F_i = f_i | C = c) = \frac{1}{\sqrt{2\pi\sigma_{ic}}} e^{-\frac{(x_i - \mu_{ic})^2}{2\sigma_{ic}^2}} \quad (2.8)$$

dimana :

$P$	=	peluang
$F_i$	=	fitur ke- $i$
$f_i$	=	nilai fitur ke- $i$
$C$	=	kelas yang di cari
$c$	=	nilai kelas yang dicari
$\mu_{ic}$	=	mean / rata-rata dari fitur ke $i$ pada kelas $c$
$\sigma_{ic}^2$	=	varian dari fitur ke $i$ pada kelas $c$

Parameter  $\mu_{ic}$  dapat diestimasi berdasarkan sampel mean  $F_i$  untuk seluruh record pada data latih yang dimiliki kelas  $c$ . Dengan cara yang sama,  $\sigma_{ic}^2$  dapat

diestimasi dari sampel varian *record* tersebut (Harini *et al*, 2007). Untuk menghitung parameter mean rumus yang digunakan ditunjukkan pada Persamaan 2.9.

$$\mu_{ic} = \frac{1}{n} \sum_{j=1}^n F_{ij} \quad (2.9)$$

dimana :

$n$  = banyaknya citra pada kelas  $c$

$F_{ij}$  = fitur ke- $i$  pada kelas  $c$

Sedangkan untuk menghitung varian ditunjukkan dalam Persamaan 2.10.

$$\sigma^2_{ic} = \frac{1}{n-1} \sum_{i=j}^n (F_{ij} - \mu_{ic})^2 \quad (2.10)$$

dimana :

$n$  = banyaknya citra pada kelas  $c$

$F_{ij}$  = fitur ke- $i$

$\mu$  = mean fitur ke- $i$  pada kelas  $c$

### 2.1.11. Naïve Bayes Classifier (NBC)

*Naive bayes* merupakan pengembangan dari metode *bayesian classification*. *Bayesian classification* adalah pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu kelas. *Bayesian classification* didasarkan pada teorema *bayes* yang memiliki

kemampuan klasifikasi serupa dengan *decision tree* dan *neural network*. *Bayesian classification* terbukti memiliki akurasi dan kecepatan yang tinggi saat diaplikasikan ke dalam database dengan data yang besar (Munir, 2004).

Dalam prosesnya, *Naive Bayes Classifier* mengasumsikan bahwa ada atau tidak adanya suatu fitur pada suatu kelas tidak berhubungan dengan ada atau tidaknya fitur lain di kelas yang sama. Probabilitas *Naive Bayes* dapat dirumuskan dalam Persamaan 2.11.

$$p(C|F_1, \dots, F_m) \quad (2.11)$$

Dimana  $C$  adalah peubah kelas yang dependen yang akan berisi salah satu kelas dari berbagai kelas, dan  $F_1$  sampai  $F_n$  adalah peubah fitur atau ciri-ciri dari masukan (Aribowo, 2010). Namun, jika nilai  $n$  terlalu besar atau ada beberapa fitur yang memiliki nilai yang sangat besar, maka dengan menggunakan teorema *bayes* persamaan di atas dapat disesuaikan menjadi seperti Persamaan 2.12.

$$p(C|F_1, \dots, F_m) = \frac{p(C)p(F_1, \dots, F_m|C)}{p(F_1, \dots, F_m)} \quad (2.12)$$

Dimana variabel  $C$  merepresentasikan kelas, sementara variabel  $F_1, \dots, F_n$  merepresentasikan karakteristik dari setiap fitur citra yang digunakan untuk melakukan klasifikasi. Jadi rumus tersebut menjelaskan bahwa peluang masuknya sampel dengan karakteristik tertentu dalam kelas  $C$  (*posterior*) adalah peluang munculnya kelas  $C$  (sebelum masuknya sampel tersebut, seringkali disebut *prior*), dikali dengan peluang kemunculan karakteristik fitur sampel pada kelas  $C$  (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga *evidence*) (Natalius, 2010).

Karena itu, secara sederhana rumus tersebut dapat ditulis seperti pada Persamaan 2.13.

$$Posterior = \frac{prior \times likelihood}{evidence} \quad (2.13)$$

Karena nilai  $F_i$  selalu diberikan dan dependen terhadap nilai  $C$ , maka nilai penyebut (*evidence*) pada persamaan di atas akan selalu konstan. Karenanya, yang bisa kita lakukan hanyalah memanipulasi pembilangnya sesuai dengan *joint probability model* sebagaimana yang ditunjukkan dalam Persamaan 2.14 (Aribowo, 2010).

$$\begin{aligned} p(C, F_1, \dots, F_m) &= p(C)p(F_1, \dots, F_m | C) \\ &= p(C)p(F_1 | C)p(F_2, \dots, F_m | C, F_1) \\ &= p(C)p(F_1 | C)p(F_2 | C, F_1)p(F_3, \dots, F_m | C, F_1, F_2) \\ &= p(C)p(F_1 | C)p(F_2 | C, F_1) \dots p(F_m | C, F_1, F_2, F_3, \dots, F_{m-1}) \end{aligned} \quad (2.14)$$

Diasumsikan setiap  $F_i$  independen secara kondisional terhadap  $F_j$  dengan  $j \neq i$ . Hal ini ditunjukkan dalam Persamaan 2.15.

$$p(F_i | C, F_j) = p(F_i | C) \quad (2.15)$$

Sehingga persamaan awal dapat ditulis kembali seperti pada Persamaan 2.16.

$$p(C | F_1, \dots, F_m) = p(C) \prod_{i=1}^n p(F_i | C) \quad (2.16)$$

Berdasarkan aturan diskriminan  $f$  pada kelas  $C$  jika  $g_i > g_j$ , untuk setiap  $j \neq i$  maka diperoleh rumusan seperti pada Persamaan 2.17.

$$g_j(f_j) = \log(p(C_j)) - \sum_{i=1}^n \log(\sigma_{ic}) - \frac{1}{2} \sum_{i=1}^n \frac{(f_j - \mu_{ic})^2}{\sigma_{ic}^2} \quad (2.17)$$

Selanjutnya, proses testing dilakukan dengan cara membandingkan nilai diskriminan dari setiap kelas dan mengambil nilai diskriminan tertinggi sebagai hasil dari testing. Sehingga dapat dirumuskan dengan Persamaan 2.18.

$$\hat{e} = \underset{c}{\operatorname{arg\,max}} g_j \quad (2.18)$$

## 2.2. Penelitian Terkait

Beberapa penelitian terdahulu yang berhubungan dengan penelitian ini, diantaranya :

Dalam sebuah survey komprehensif dijabarkan mengenai bahasa tulisan tangan, bagaimana mengenali tulisan tangan menjadi teks digital, dan konsep dasar algoritma pengenalan bahasa tulisan. Baik secara langsung maupun tidak langsung. Serta penjelasan mengenai algoritma preproses, pengenalan karakter dan huruf, serta beberapa aplikasi seperti verifikasi tanda tangan, pengenalan penulis serta alat pembelajaran tulisan tangan (Plamondon *et al*, 2000).

Dalam sebuah penelitian mengenai pengenalan karakter alfanumerik dengan algoritma *Neural Network Three-Layer Backpropagation* diperoleh hasil bahwa semakin tinggi jumlah layer yang dipakai akan semakin meningkatkan tingkat akurasi pengenalan karakter alfanumerik yang diinginkan, namun dengan pengorbanan di lamanya waktu *training* yang bertambah sangat signifikan (Emanuel *et al*, 2007).

Dalam sebuah penelitian lain tentang konversi teks tulisan tangan juga dijelaskan bahwa dari hasil uji coba menunjukkan selama pengambilan data karakter didapat jumlah keseluruhan *epoch* 164000, jumlah keseluruhan *squared error* 0,1031610851753050 dan jumlah keseluruhan klasifikasi piksel *error* 5,4930% dengan *neuron error* 36032 dan jumlah *output neuron* 656000. Dan hasil uji coba pada pengenalan setiap karakter teks menunjukkan prosentasi rata-rata keberhasilan 66,58% dan pengkonversian teks sebesar 50,00% (Arief, 2009).

Selain itu, hasil pengujian yang dilakukan dalam sebuah penelitian tentang pengenalan karakter alfabet dengan jaringan saraf tiruan menunjukkan bahwa arsitektur dan parameter jaringan yang memberikan unjuk kerja paling optimal untuk seluruh jenis font adalah 60 neuron pada lapisan tersembunyi, laju pembelajaran 0,01, momentum 0,9 dan iterasi 1000 kali. Prosentase pengenalan dari pelatihan jaringan dengan arsitektur dan parameter jaringan optimal ini untuk data latih dan data baru masing-masing adalah 100% dan 99,89% dengan jumlah karakter takdikenali untuk data latih sebanyak 0 karakter dan data baru sebanyak 2 karakter (Prasojo, 2011).

Dalam sebuah penelitian tentang implementasi dari algoritma *Backpropagation* dalam pengolahan citra teks tulisan tangan menjadi teks digital diperoleh akurasi sistem 67,99% yang diperoleh dari hasil uji coba pada 50 data uji, dengan menggunakan 660 data referensi, *learning rate* = 0,001, *error rate* = 0,01, *epoch* = 100, inisialisasi bobot awal = -0,5 sampai 0,5, *input patern* = 30 x 30 piksel, *input neuron* = 900, *hidden layer* 55, dan *output neuron* = 900. Dimana

dapat disimpulkan bahwa aplikasi ini cukup baik dalam proses pengenalan pola tulisan tangan (Fifin *et al*, 2012).

Dari hasil penelitian tentang klasifikasi teks untuk pengelompokkan teks berita dengan metode *Naïve Bayes Classifier* menunjukkan hasil bahwa pada dokumen berita akurasi maksimal dicapai 91% sedangkan pada dokumen akademik 82%. Seleksi kata dengan minimal muncul pada 4 atau 5 dokumen memberikan akurasi yang paling tinggi (Hamzah, 2012).

Dalam sebuah penelitian tentang pengenalan tulisan tangan huruf alfabet dengan metode *Modified Direction Feature (MDF)* untuk proses ekstraksi fitur dan *Learning Vector Quantization (LVQ)* untuk proses klasifikasi, berdasarkan hasil uji coba diperoleh hasil terbaik untuk pengenalan citra huruf tulisan tangan yaitu dengan jumlah data pelatihan sebanyak 260 citra dan data uji coba sebanyak 52 citra diperoleh akurasi sistem sebesar 71,15 % (Safitri, 2012).

Dalam suatu penelitian tentang deteksi manusia menggunakan *Histogram Of Oriented Gradients* dan *Naïve Bayes Classifier* diperoleh hasil uji coba aplikasi menggunakan perhitungan distribusi gaussian probabilitas pada *Naïve Bayes Classifier* dengan nilai threshold = 12,13,14 dan 15 pada skenario 1, 2 dan 3 diperoleh rata – rata akurasi sebesar 68,65% (Uluwiyah *et al*, 2013).

Dalam sebuah penelitian tentang pengenalan karakter tulisan tangan latin menggunakan metode *Backpropagation* dengan pengujian menggunakan citra karakter tulisan tangan dari 5 orang yang berbeda diperoleh hasil akurasi program sebesar 87.2%. Penentuan nilai learning rate sangat mempengaruhi pergesaran

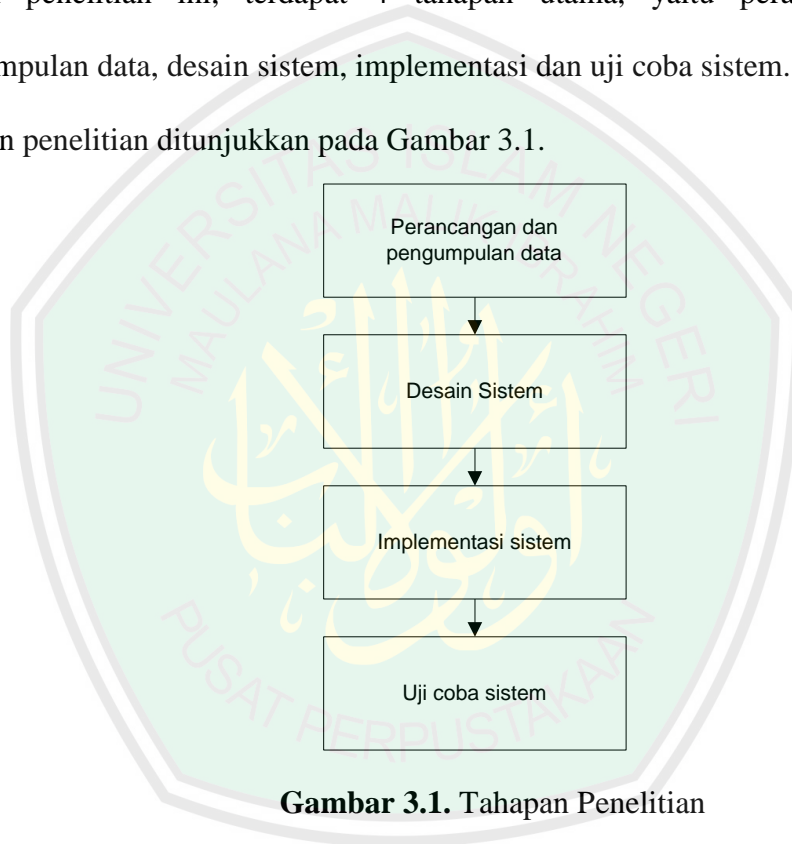
nilai pada bobot-bobot saat proses pelatihan jaringan syaraf tiruan (Sinambela *et al*, 2013).

Dari berbagai hasil penelitian tersebut sebagian besar peneliti yang meneliti tentang pengenalan tulisan tangan memilih menggunakan metode *Backpropagation* pada jaringan saraf tiruan untuk proses pengenalan. Jaringan saraf tiruan (*Artificial Neural Network*) merupakan salah satu sistem pemrosesan informasi yang menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah melalui proses belajar. Metode *Backpropagation* termasuk dalam kelompok *supervice learning* yang merupakan proses pembelajaran yang terawasi. Karena telah banyak peneliti sebelumnya yang telah membuktikan bahwa metode *Backpropagation* cocok untuk pengenalan tulisan tangan, maka dalam penelitian ini akan digunakan metode lain yaitu metode *Naïve Bayes Classifier* yang merupakan pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu kelas. Dan beberapa peneliti sebelumnya juga telah dibuktikan bahwa metode ini baik dalam kasifikasi namun belum dibuktikan secara langsung dalam pengenalan pola tulisan tangan. Oleh karena itu untuk proses pengenalan pola tulisan tangan dalam penelitian ini akan digunakan metode *Naïve Bayes Classifier* untuk proses pengenalan dan untuk mengukur akurasi metode *Naïve Bayes Classifier* dalam mengenali tulisan tangan.

## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

Dalam bab ini akan dijabarkan mengenai perancangan dan pengumpulan data, deskripsi dan desain sistem, desain interface serta perancangan uji coba. Dalam penelitian ini, terdapat 4 tahapan utama, yaitu perancangan dan pengumpulan data, desain sistem, implementasi dan uji coba sistem. Blok diagram tahapan penelitian ditunjukkan pada Gambar 3.1.



**Gambar 3.1.** Tahapan Penelitian

#### 3.1.Deskripsi Data

Dalam pembangunan aplikasi pengenalan tulisan tangan ini terdapat dua macam data yang dibutuhkan, yaitu data *training* dan data *testing*.

##### a. Data training

Data *training* berupa data tulisan tangan yang diperoleh dari beberapa responden dari berbagai umur dalam sebuah kalimat yang mengandung huruf a –

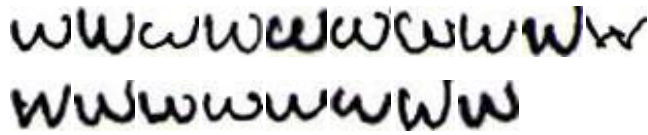

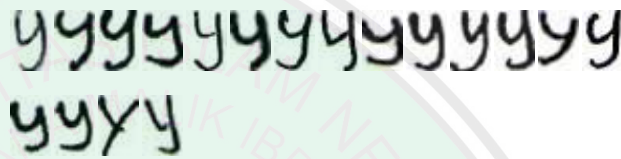
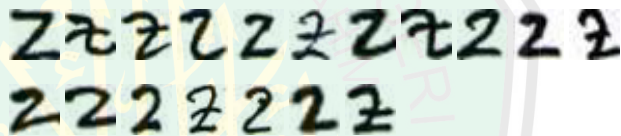
z dan berhuruf kecil serta tidak bersambung, media yang digunakan untuk menulis tulisan tangan yaitu kertas HVS putih. Kemudian data akan di-scan dengan scanner dalam bentuk citra yang berformat JPEG. Data citra tersebut kemudian akan di-*crop* per karakter huruf dan disimpan dalam sebuah folder data *training*. Setiap huruf dalam data *training* akan mewakili satu kelas dengan jumlah 18 sampel per huruf. Dalam Tabel 3.1 dijabarkan mengenai objek *training* yang mewakili kelas tertentu dan sampel citra dari setiap objek yang digunakan sebagai data training.

**Tabel 3.1.** Data *training* huruf tulisan tangan

Kelas Objek Training	Objek Training	Citra Objek Training
1	a	
2	b	
3	c	
4	d	
5	e	

Kelas Objek Training	Objek Training	Citra Objek Training
6	f	F F F F f F f F F f F f F f F f f f
7	g	G G G G G G G G G G G G G G G G
8	h	H H H H H H H H H H H H H H H H
9	i	I I I I I I I I I I I I I I I I
10	j	J J j j J J J J J J J J J J j j
11	k	K K K k K K K K K K K K K K K K
12	l	L L L L L L L L L L L L L L L L
13	m	M M M M M M M M M M M M M M M M M M M M
14	n	N N N N N N N N N N N N N N N N N N N N

Kelas Objek Training	Objek Training	Citra Objek Training
15	o	
16	p	
17	q	
18	r	
19	s	
20	t	
21	u	
22	v	

Kelas Objek Training	Objek Training	Citra Objek Training
23	w	
24	x	
25	y	
26	z	

#### b. Data testing

Data testing juga berupa data tulisan tangan yang ditulis pada media kertas HVS yang diperoleh dari beberapa responden yang nantinya akan di scan dengan scanner dalam bentuk citra yang berformat JPEG dan dikenali oleh aplikasi pada saat uji coba sistem untuk mengetahui apakah aplikasi dapat berjalan sesuai tujuan dan untuk mengukur akurasi metode *Naïve Bayes Classifier* dalam mengenali tulisan tangan. Data testing yang dapat dikenali dapat berupa citra huruf, kata atau satu baris kalimat yang berhuruf kecil dan tidak bersambung.

### 3.2.Deskripsi Sistem

Dalam penelitian ini aplikasi akan dibuat dalam sebuah halaman utama yang terdiri dari dua menu, yaitu menu proses *training* dan menu pengenalan. Yang masing – masing menu akan menampilkan *interface* proses *training* dan *interface* pengenalan. *Interface* proses *training* ditujukan untuk mempermudah pengembang dalam melakukan uji coba sistem dan untuk melakukan proses *training* untuk memperoleh parameter mean dan parameter varian yang akan digunakan pada saat proses pengenalan, selain itu *Interface* proses *training* juga bertujuan untuk mengukur akurasi *training*, sedangkan *interface* proses *testing* ditujukan untuk user secara umum untuk mengenali citra huruf atau kata tulisan tangan menjadi teks digital selain itu *interface* proses *testing* juga bertujuan untuk mengukur akurasi *testing* baik testing dengan *input* berupa huruf maupun *testing* dengan *input* berupa kata dan satu baris kalimat.

### 3.3.Desain Sistem

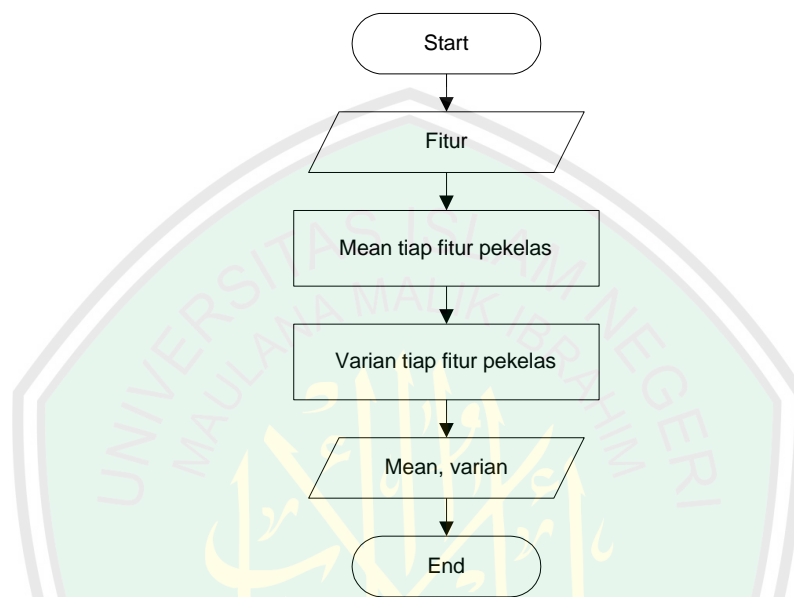
Dalam penelitian ini, terdapat dua tahapan utama yang akan dilakukan untuk dapat mengenali tulisan tangan, yaitu tahap *training* dan tahap *testing* / pengenalan. Namun, sebelum dilakukan dua tahapan tersebut, terlebih dahulu citra *training* dan citra *testing* akan melalui preproses atau proses awal yang biasa dilakukan dalam setiap pengolahan citra, yaitu berupa proses *grayscale*, *scalling* dan ekstraksi fitur. Untuk citra yang akan dikenali yang berbentuk sebuah kata atau kalimat akan melalui preproses tambahan yaitu proses *Projection Profile* dan proses *cropping*.

Pada proses training, pertama – tama citra training akan diubah dalam bentuk *grayscale* kemudian akan dilakukan proses *scalling* untuk memperbesar (*zoom-in*) atau memperkecil (*zoom-out*) citra yang akan diproses sesuai kebutuhan, setelah itu dilakukan ekstraksi fitur yaitu pengambilan ciri / fitur dari citra yang akan dikenali dalam hal ini fitur akan diubah dalam bentuk vektor kemudian dilakukan estimasi parameter yaitu menghitung nilai mean dan varian dari setiap fitur perkelas dan disimpan kedalam database yang nantinya akan digunakan pada proses *testing*. Sedangkan pada proses *testing*, citra yang akan dikenali akan diubah dalam bentuk *grayscale* kemudian akan dilakukan sebuah proses untuk memecah kalimat menjadi huruf dengan mencari batas baris dengan *vertical projection* dan batas kolom dari karakter dengan *horizontal projection* yang merupakan tahapan dalam *Projection Profile*, kemudian dilakukan proses *cropping* untuk memotong citra sesuai dengan koordinat yang telah diperoleh pada proses *Projection Profile* kemudian dilanjutkan dengan proses *scalling* untuk memperbesar (*zoom-in*) atau memperkecil (*zoom-out*) ukuran dimensi pixel citra sesuai kebutuhan dan ekstraksi fitur dalam bentuk vektor setelah itu akan dilakukan proses pengenalan dengan metode *Naive Bayes Classifier* dengan *input* berupa parameter mean dan parameter varian yang diperoleh dari proses *training* serta hasil ekstraksi fitur citra yang akan dikenali. Alur proses sistem ditunjukkan dalam blok diagram pada Gambar 3.2.



**Gambar 3.2.** Blok diagram desain sistem

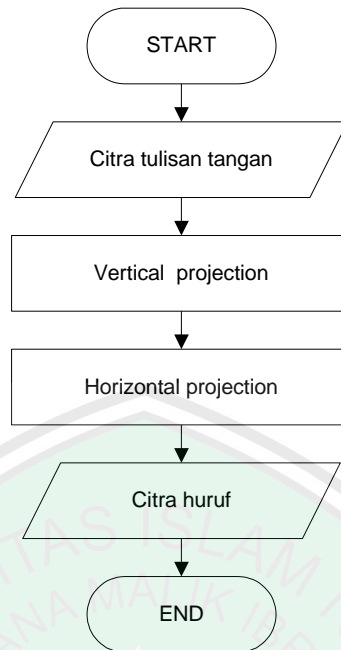
Dalam blok diagram pada Gambar 3.2, pada proses *training* dilakukan preprosesing pada citra karakter tulisan tangan yang berupa *grayscale*, *scaling* dan ekstraksi fitur. Selanjutnya dilakukan estimasi parameter, berikut rincian proses estimasi parameter yang digambarkan dalam *flowchart* pada Gambar 3.3.



**Gambar 3.3.** *Flowchart* proses estimasi parameter

Pada proses estimasi parameter ini akan dicari nilai mean seperti yang ditunjukkan dalam Persamaan 2.10 dan nilai varian dengan Persamaan 2.11 dari setiap fitur pekelas yang selanjutnya akan disimpan ke dalam database yang nantinya akan digunakan dalam proses testing.

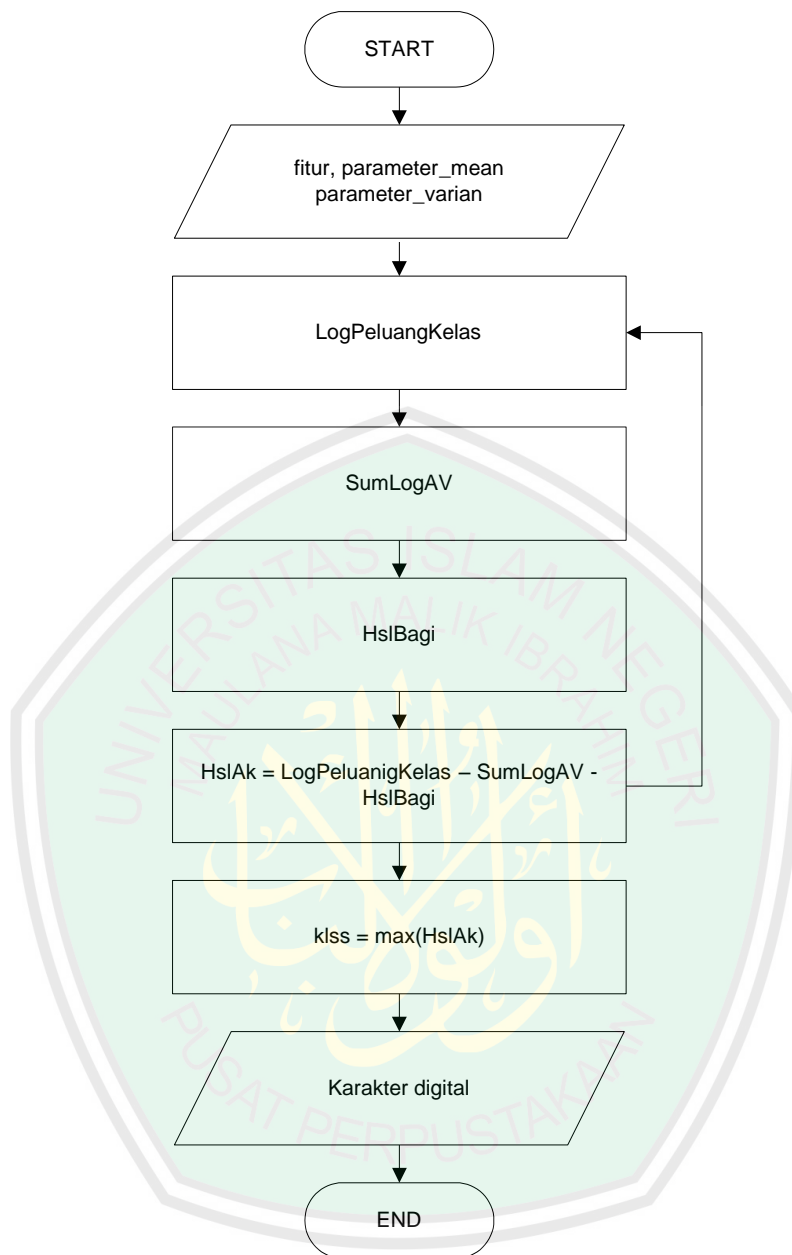
Kemudian sama seperti citra sampel, pada citra input yang akan dikenali juga akan dilakukan preproses berupa *grayscale*, kemudian dilakukan *Projection Profile* untuk memecah kata menjadi huruf. *Flowchart Projection Profile* ditunjukkan pada Gambar 3.4.



**Gambar 3.4.** *Flowchart Projection Profile*

*Projection Profile* dilakukan dalam dua tahap yaitu *vertical projection* seperti yang ditunjukkan dalam Persamaan 2.4 dan *horizontal projection* yang ditunjukkan pada Persamaan 2.5. *vertical projection* bertujuan untuk memperoleh batas baris dari huruf sedangkan *horizontal projection* bertujuan untuk memperoleh batas kolom dari setiap huruf. Setelah diketahui batas baris dan kolom dari sebuah kata dengan *Projection Profile* kemudian dilakukan proses *cropping*, *scalling* dan ekstraksi fitur yang selanjutnya akan dilakukan proses *testing* dengan menggunakan metode *Naive Bayes Classifier* dengan inputan hasil estimasi parameter pada proses *training* dan citra yang akan dikenali.

Proses pengenalan dengan NBC ditunjukkan dalam *Flowchart* pada Gambar 3.5.



**Gambar 3.5.** Flowchart NBC

Seperti yang terlihat dalam *flowchart* pada Gambar 3.5, dalam proses pengenalan dengan NBC pertama – tama akan dihitung nilai LogPeluangKelas ( $\log(p(C_j))$ ) dari setiap fitur, kemudian dihitung SumLogAV ( $\sum_{i=1}^n \log(\sigma_{ic})$ )

yaitu jumlah total nilai log akar varian dan HslBagi  $(\frac{1}{2} \sum_{i=1}^n \frac{(f_j - \mu_{ic})^2}{\sigma_{ic}^2})$  dimana  $f$  adalah fitur citra testing,  $\mu_{ic}$  adalah nilai parameter mean dan  $\sigma_{ic}^2$  adalah nilai parameter varian yang diperoleh dari proses training, setelah diperoleh hasil dari ketiga langkah tersebut kemudian dihitung nilai HslAk dengan mengurangi ketiganya  $(g_j(f_j) = \log(p(C_j)) - \sum_{i=1}^n \log(\sigma_{ic}) - \frac{1}{2} \sum_{i=1}^n \frac{(f_j - \mu_{ic})^2}{\sigma_{ic}^2})$  selanjutnya menghitung klss yaitu nilai maksimal dari HslAk dengan rumus yang ditunjukkan pada persamaan 2.18.

### 3.4. Desain *Interface*

Dalam penelitian ini, terdapat sebuah halaman utama yang memiliki dua menu yang masing masing akan menampilkan *interface training* dan *interface pengenalan*. Desain *interface* halaman utama ditunjukkan pada Gambar 3.6.



**Gambar 3.6.** Desain *interface* halaman utama

*Interface* proses *training* digunakan untuk melatih sekumpulan data karakter tulisan tangan yang sudah disimpan dalam folder tertentu dan untuk mengetahui akurasi proses *training*. Tampilan desain *interface* proses *training* yang ditunjukkan pada Gambar 3.7.

The image shows a software interface for handwriting recognition training. The title is "Proses Training Pengenalan Tulisan Tangan". It features several input fields for configuration: "Folder training", "Dimensi pixel citra" (with sub-fields for "Baris" and "Kolom"), and "Informasi data training" (with sub-fields for "Jumlah data", "Banyak kelas", and "Data perkelas"). A "Training" button is positioned to the right of the data training fields. Below these are three large rectangular areas labeled "Nilai mean", "Nilai varian", and "Confusion Matrik", which are currently empty. At the bottom left, there is an "Akurasi" field. A large, semi-transparent watermark of a university logo is overlaid on the interface.

**Gambar 3.7.** Desain *interface* proses *training*

Pada desain *interface* proses *training* tersebut terdapat tombol untuk memilih folder dimana kumpulan data untuk *training* disimpan, kemudian terdapat isian dimensi pixel citra untuk merubah ukuran citra sesuai kebutuhan, kemudian pada bagian informasi data training terdapat isian jumlah data, banyak

kelas dan data perkelas yang masing – masing harus diisi untuk dapat melakukan proses training dengan memilih tombol *training*. Setelah dilakukan proses *training* maka hasil proses *training* berupa nilai mean, nilai varian, confusion matrik dan akurasi sistem akan ditampilkan pada *interface*. Sedangkan *interface* proses *testing* digunakan untuk melakukan pengenalan tulisan tangan. Tampilan desain *interface* proses pengenalan ditunjukkan pada Gambar 3.8.

The image shows a software interface for handwritten character recognition. It is titled "Proses Pengenalan Tulisan Tangan". The interface is organized into two main functional areas, each enclosed in a rounded rectangle. The first area, "Pengenalan Huruf", includes a "Pilih Citra" button, a text input field, and "Preproses" and "Testing" buttons. The second area, "Pengenalan Kata", includes a "Pilih Folder" button, a text input field, and "Preproses" and "Testing" buttons. At the bottom of the interface, there are two empty rectangular boxes with diagonal lines, suggesting placeholders for images or data. A large, semi-transparent watermark of the Maulana Malik Ibrahim University logo is overlaid on the entire interface.

**Gambar 3.8.** Desain *interface* proses *testing*

Pada desain *interface* proses *testing* yang ditunjukkan pada Gambar 3.8 terdapat dua pilihan pengenalan, yaitu pengenalan huruf dan pengenalan kata yang masing – masing sebelum dilakukan proses *testing* akan terlebih dahulu melalui

preproses. Perbedaanya, pada bagian pengenalan huruf, yang dapat dikenali hanya berupa huruf tunggal, sedangkan pada pengenalan kata, sistem dapat mengenali beberapa huruf sekaligus yang membentuk kata pada citra testing yang diinputkan. Namun pada citra testing yang berbentuk kata atau satu baris kalimat, sebelum dilakukan proses pengenalan citra terlebih dahulu akan melalui proses *cropping* menjadi huruf dengan koordinat yang diperoleh dari proses *vertical projection* dan *horizontal projection* pada *Projecton Profile*.



## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1. Platform yang Digunakan**

Pada sub bab ini akan dijabarkan mengenai platform perangkat keras dan platform perangkat lunak untuk membangun aplikasi dan uji coba sistem.

##### **a. Platform Perangkat Keras**

Platform perangkat keras yang digunakan yaitu :

1. Laptop / komputer processor Intel 32 – bit
2. Memory 2048 MB
3. Hardisk 1 GB
4. Piranti masukan : *keyboard, mouse, scanner*

##### **b. Platform Perangkat Lunak**

Platform perangkat lunak yang digunakan yaitu :

1. Windows XP Service Pack 3
2. Matlab

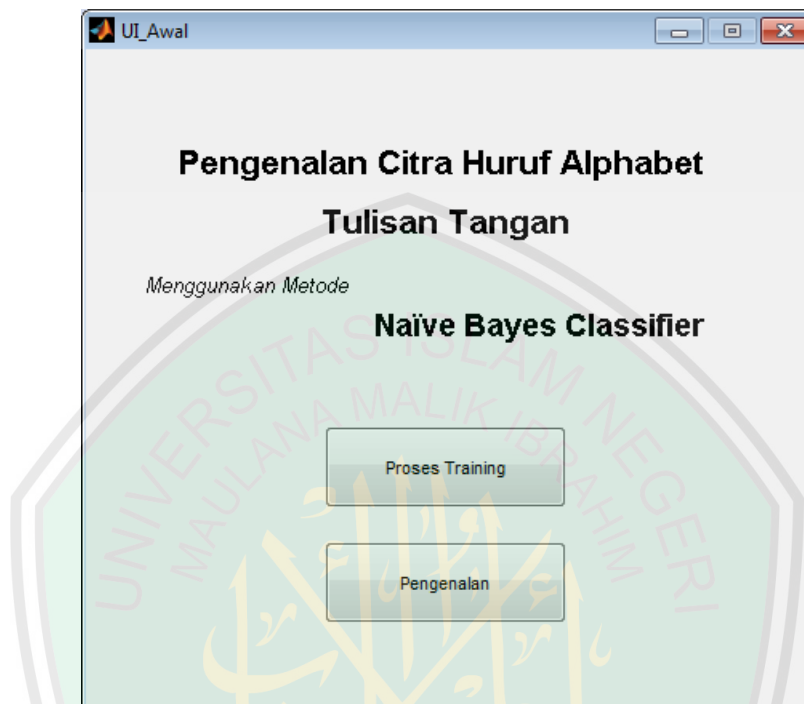
#### **4.2. Implementasi Sistem**

Pada bagian ini akan dijabarkan mengenai penjelasan program dari *interface* hingga implementasi metode dalam bahasa pemrograman komputer.

##### **4.2.1. Halaman Utama Aplikasi**

Sebelum masuk ke *interface* training dan pengenalan, terlebih dahulu akan tampil *halaman utama* yang berisi menu training dan menu pengenalan. Pada

halaman utama terdapat dua menu yaitu menu proses *training* dan menu proses pengenalan. Yang masing – masing menu akan diarahkan pada halaman sesuai menunya. Tampilan halaman utama aplikasi ditunjukkan pada Gambar 4.1.

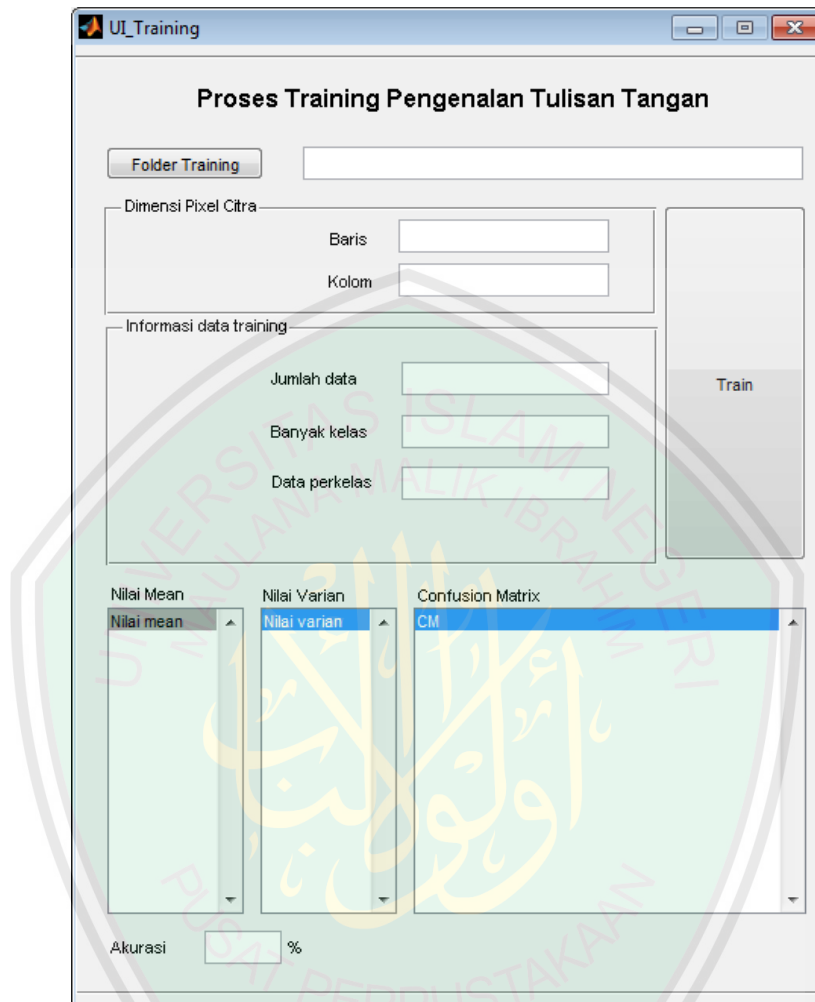


**Gambar 4.1.** Halaman utama aplikasi

#### 4.2.2. *Interface Proses Training*

Pada proses *training* ini, pengguna diharuskan mengisi semua informasi yang dibutuhkan untuk melakukan proses *training* yaitu memilih folder dimana kumpulan data *training* disimpan, mengisi dimensi pixel baris dan kolom citra yang diinginkan, jumlah data yaitu jumlah keseluruhan data training, banyak kelas yaitu banyaknya kelas dalam data training, data perkelas yaitu jumlah data setiap sampel huruf. Setelah dilakukan proses *training*, hasil estimasi parameter mean, varian, nilai confusion matrik dan akurasi sistem akan ditampilkan pada *interface*. Selanjutnya, nilai mean dan varian yang diperoleh dari proses *training*

akan disimpan yang nantinya akan digunakan pada saat pengenalan. Tampilan *interface* proses *training* ditunjukkan pada Gambar 4.2.



**Gambar 4.2.** *Interface* proses *training*

Pada pembentukan proses *training* terdapat beberapa tahapan yang harus dilakukan, yaitu pembentukan data *training*, mengambil data training perkelas dan perhitungan parameter mean dan parameter varian. *Source code* pembentukan data *training* ditunjukkan pada Gambar 4.3.

```

for data = 1:jlmData
    objek = [DirektoriDataTraining, '\' int2str(data) \'.JPEG'];
    objek = imread(objek);
    objek = rgb2gray (objek);
    objek = imresize(objek,[baris,kolom]);
    data_train(data,1:baris*kolom)=reshape(objek,1,baris*kolom);
end
c = 0;
for i = 1:banyakKls
    for j = 1:dataPerkls
        c = c+1;
        data_class(c) = i;
    end
end
data_train = double(data_train);
data_class = double(data_class');

```

**Gambar 4.3.** *Source code* pembentukan data training

*Source code* untuk mengambil data perkelas ditunjukkan pada Gambar 4.4.

```

[a b] = size(data_train);
banyak_kelas = max(data_class);
banyak_data_prkelas = [];
jml = 0;
%menentukan banyak data per kelas
for k = 1:banyak_kelas
    for j = 1:a
        if data_class(j) == k
            jml = jml+1;
        end
    end
    banyak_data_prkelas(k) = jml;
    jml = 0;
end

data_awal = 0;
%menentukan data perkelas
for k = 1:banyak_kelas
    if k == 1
        for baris = 1:banyak_data_prkelas(1)
            for kolom = 1:b
                data_perkelas (baris,kolom,k)=data_train (baris,kolom);
            end
        end
    else
        data_awal = data_awal + banyak_data_prkelas;
        data_akhir = data_awal + banyak_data_prkelas;
        for baris = (data_awal+1):data_akhir
            for kolom = 1:b
                data_perkelas ((baris-data_awal),kolom,k)=data_train (baris,kolom);
            end
        end
    end
end
end
end

```

**Gambar 4.4.** *Source code* untuk mengambil data perkelas

*Source code* perhitungan parameter mean dan parameter varian ditunjukkan pada Gambar 4.5.

```
for kelas = 1:banyak_kelas
    parameter_mean(:, :, kelas) = mean(data_perkelas(:, :, kelas));
    parameter_varian(:, :, kelas) = std(data_perkelas(:, :, kelas));
end
```

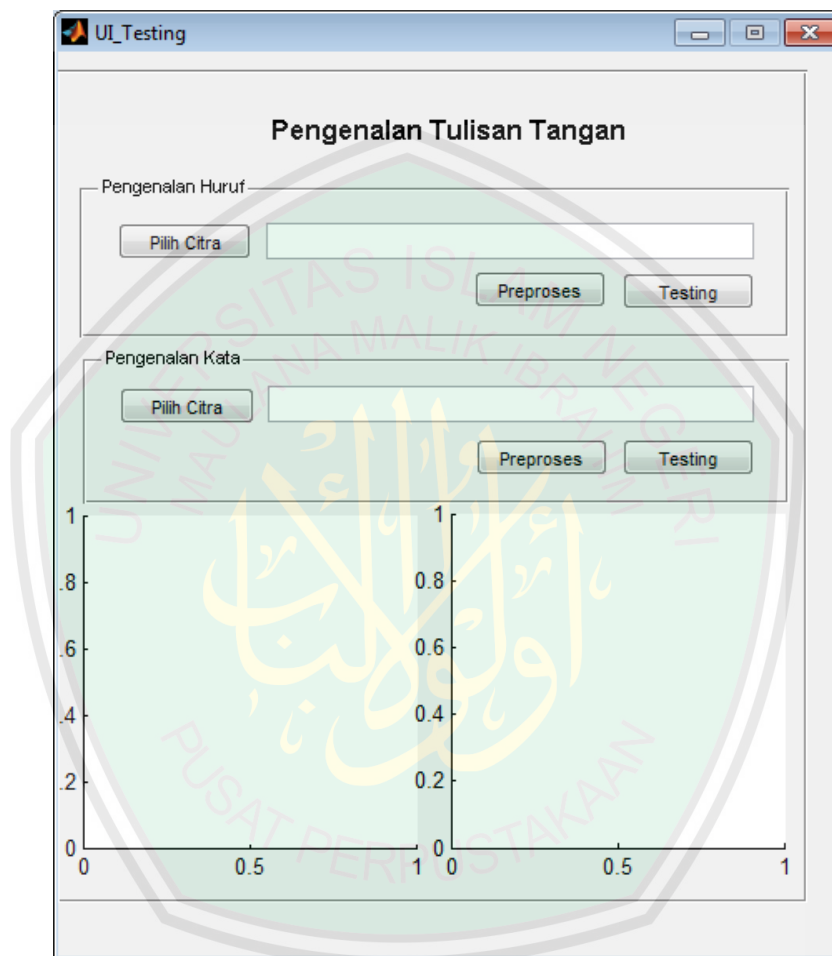
**Gambar 4.5.** *Source code* estimasi parameter

### 4.2.3. Interface Proses Testing

Pada proses *testing*, terdapat dua pilihan pengenalan, yaitu pengenalan huruf dan pengenalan kata. Pada pengenalan huruf, sesuai namanya pengenalan huruf hanya dapat digunakan untuk mengenali satu karakter huruf. Sedangkan pada pengenalan kata dapat dikenali beberapa huruf sekaligus yang tersusun dalam sebuah kata. Tampilan proses *testing* ditunjukkan pada Gambar 4.6. Untuk dapat melakukan pengenalan huruf, terlebih dahulu harus diambil citra huruf yang akan dikenali, citra yang akan dikenali berupa citra huruf yang berformat JPEG, kemudian dilakukan preproses dan dilanjutkan dengan proses *testing*.

Seperti pada pengenalan huruf, untuk melakukan pengenalan kata, terlebih dahulu juga harus dipilih citra kata yang ingin dikenali. Citra yang akan dikenali harus berupa citra kata atau satu baris kalimat tulisan tangan yang tidak bersambung dan berformat JPEG. Yang membedakan pada proses pengenalan huruf dan pengenalan kata adalah preprosesnya. Pada pengenalan kata atau kalimat, citra kata terlebih dahulu harus melalui proses *Projection Profile* untuk memperoleh batas baris dan batas kolom dari setiap huruf yang membentuk citra kata yang akan dikenali, kemudian di – *cropp* menggunakan korrdinat yang telah

diperoleh dari proses *Projection Profile* untuk memecah kata menjadi huruf – huruf yang selanjutnya akan disimpan pada folder tertentu. Setelah dilakukan preproses dengan memilih tombol preproses kemudian citra akan dikenali dengan metode NBC dengan memilih tombol *testing*.



**Gambar 4.6.** *Interface proses testing*

*Source code* proses pengenalan dengan *Naïve Bayes Classifier* ditunjukkan pada Gambar 4.7.

```

function [hs11 pl huruf2] = NBC_tes(data_testing)
load banyak_kelas
load banyak_data_prkelas
load parameter_varian
load parameter_mean
load data_perkelas
[a b] = size(data_testing);
for l=1:2:a
for k=1:banyak_kelas
    LogPeluangKelas = log(banyak_data_prkelas/a);
    AkarVarian(:, :, k) = sqrt(parameter_varian(:, :, k));
    LogAV(:, :, k) = log (AkarVarian(:, :, k));
    SumLogAV(:, :, k) = sum (LogAV(:, :, k));
    FitTest = data_testing(l, :, :);
    pem(l, :, k) = FitTest - parameter_mean(:, :, k);
    [brs klm] = size (pem(l, :, k));
    pemm = pem(l, :, k);
    for kl = 1:klm
        PemKuad(kl) = pemm(kl)^2;
    end
    PemKuad;
    SumBagi(l, :, k) = sum(PemKuad / parameter_varian(:, :, k));
    HslBagi2(l, :, k) = 1/2 * SumBagi(l, :, k);
    HslAk(l, :, k) = LogPeluangKelas(k) - SumLogAV(:, :, k) -
HslBagi2(l, :, k);
    [pl(l, :), klss] = max(HslAk(l, :));
    hs11(l) = klss;
    huruf(l) = tulis (hs11(l));
end
    hasil(1,1:1) = reshape(hs11,1,1);
    huruf2(1,1:1) = reshape(huruf,1,1);
end
end

```

**Gambar 4.7.** Source code proses pengenalan dengan NBC

Untuk mengetahui akurasi proses *training*, maka perlu dilakukan perhitungan akurasi hasil pengenalan yang dilakukan pada citra training. *Source code* perhitungan akurasi sistem ditunjukkan pada Gambar 4.8.

```

[CM, CH, P]=crosstab(data_class(:),hs11(:));
error=0;
for i=1:jumlahData
    Hasil=[i data_class(i) hs11(i)];
    if data_class(i)~=hs11(i)
        error=error+1;
    end
end
error;
benar = jumlahData - error;
akurasi = (benar)*100/jumlahData;

```

**Gambar 4.8.** Source code perhitungan akurasi sistem

### 4.3. Uji Coba Sistem

Pada sub bab ini akan dijabarkan mengenai rancangan uji coba dan hasil uji coba untuk menguji kinerja sistem dan mengukur akurasi sistem.

#### 4.3.1. Perancangan Uji Coba

Uji coba akan dilakukan pada proses training dan proses testing. Pada proses training uji coba bertujuan untuk melakukan proses training untuk memperoleh nilai parameter mean dan parameter varian yang nantinya akan digunakan pada proses testing selain itu uji coba proses training juga bertujuan untuk mengetahui akurasi training. Sedangkan pada uji coba proses testing terdapat dua macam uji coba, yaitu uji coba dengan input berupa citra huruf tunggal dan citra kata atau satu baris kalimat. Citra input berupa tulisan tangan yang ditulis pada media kertas HVS yang discan dengan scanner dalam format JPEG dan akan dikenali oleh sistem menggunakan metode *Naïve Bayes Classifier* untuk mengetahui apakah metode *Naïve Bayes Classifier* dapat digunakan untuk mengenali tulisan tangan dan mengukur akurasi metode dalam mengenali tulisan tangan.

Skenario pengujian dilakukan pada aplikasi pengenalan citra tulisan tangan baik pada proses training dan proses testing. Pada proses training uji coba dilakukan dengan memilih folder dimana kumpulan data training disimpan dan mengisikan semua informasi yang diperlukan dan memilih tombol training. Uji coba proses training bertujuan untuk mendapatkan nilai parameter mean dan parameter varian yang dibutuhkan untuk proses testing dan untuk mengukur akurasi proses training. Uji coba proses testing dilakukan dengan dua macam

input, yaitu uji coba proses testing dengan input berupa citra huruf tunggal dan uji coba testing dengan input berupa citra kata atau satu baris kalimat. Pada uji coba proses testing dengan input berupa citra kata atau kalimat, citra uji harus berupa citra tulisan tangan yang tegak lurus / tidak bersambung yang ditulis pada media kertas HVS putih dan di – scan dengan scanner dengan format JPEG. Uji coba proses testing bertujuan untuk mengetahui apakah aplikasi dapat berjalan sesuai rencana serta menghitung akurasi hasil pengenalan.

#### **4.3.2. Hasil Uji Coba Proses *Training***

Proses *training* dilakukan dengan data *training* sebanyak 468 data, dengan masing – masing huruf terdiri dari 18 sampel. Proses *training* dilakukan untuk memberikan pelatihan pada sistem agar dapat mengenali citra inputan yang akan dikenali. Akurasi training diperoleh dengan mengenali kembali setiap data training sebagai input testing dengan metode *Naïve Bayes Classifier*. Berdasarkan hasil uji coba dengan bermacam – macam kombinasi dimensi pixel baris dan kolom, diperoleh dimensi baris dan kolom terbaik yaitu 7x4 dengan akurasi training sebesar 81,1966%, hal ini ditunjukkan pada Tabel 4.1. Gambar 4.9 menunjukkan hasil uji coba *training* dengan memilih folder tempat data *training* disimpan dan mengisi beberapa inputan yang diperlukan maka hasil training berupa parameter mean dan varian serta confusion matrik dan akurasi training akan ditampilkan. Nilai parameter mean dan parameter varian inilah yang nantinya akan digunakan pada pada proses pengenalan. Nilai parameter mean dan parameter varian dapat dilihat pada Lampiran 2.



**Tabel 4.1.** Akurasi proses *training* berdasarkan dimensi pixel baris dan kolom

Baris	Kolom	Akurasi	Error
6	2	64,316	35,684
6	3	3,846	96,154
6	4	3,846	96,154
6	5	3,846	96,154
6	6	74,359	25,641
6	7	72,009	27,992
6	8	66,880	33,120
6	9	61,111	38,889
6	10	57,906	42,094
7	2	63,675	36,325
7	3	78,419	21,581
<b>7</b>	<b>4</b>	<b>81,197</b>	<b>18,803</b>
7	5	77,992	22,009
7	6	74,573	25,427
7	7	72,222	27,778
7	8	64,744	35,256
7	9	58,547	41,453
7	10	54,701	45,299
8	2	61,752	38,248
8	3	76,923	23,077
8	4	77,992	22,009
8	5	76,709	23,291
8	6	72,436	27,564
8	7	67,521	32,479
8	8	60,897	39,103
8	9	54,274	45,727
8	10	51,709	48,291
9	2	60,684	39,316
9	3	74,573	25,427
9	4	76,923	23,077
9	5	75,000	25,000
9	6	67,735	32,265
9	7	62,180	37,821

**Tabel 4.2.** Nilai confusion matrik hasil uji coba akurasi *training*

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0
b	0	13	0	0	0	0	0	3	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
c	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
d	0	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	0	0	1	0	13	1	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0
f	0	0	0	0	0	11	0	0	0	1	0	3	0	0	0	0	0	3	0	0	0	0	0	0	0	0
g	0	0	0	0	0	1	15	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	0	0	0	0	0	0	0	0	12	0	0	2	0	0	0	0	0	0	0	2	0	0	0	1	0	1
j	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
k	0	0	0	0	0	0	0	4	0	0	13	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
l	0	0	0	0	0	0	0	0	2	0	1	14	0	0	0	0	0	0	0	1	0	0	0	0	0	0
m	0	0	0	0	0	0	0	0	0	0	0	17	1	0	0	0	0	0	0	0	0	0	0	0	0	0
n	0	0	0	0	0	0	0	0	0	0	0	1	11	0	0	0	6	0	0	0	0	0	0	0	0	0
o	0	0	1	0	0	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	1	0	0	0	0	0	0	0	1
q	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	14	0	0	0	0	0	0	0	0	2
r	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	17	0	0	0	0	0	0	0	0
s	0	1	0	0	0	1	0	1	0	0	0	3	0	0	0	0	0	1	11	0	0	0	0	0	0	0
t	0	0	0	0	0	0	0	0	1	0	1	2	0	0	0	0	0	0	0	14	0	0	0	0	0	0
u	0	0	0	1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	14	1	0	0	0	0
v	0	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	1	14	0	0	0	0
w	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17	0	0	0
x	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	1	0	14	0	0
y	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	15	0
z	0	0	0	0	0	0	0	0	2	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	13

### 4.3.3. Hasil Uji Coba Proses *Testing*





Uji coba *testing* dilakukan dengan dua macam inputan, yaitu *testing* dengan inputan berupa huruf tunggal dan *testing* dengan inputan berupa kata. Berikut hasil uji coba aplikasi dengan 104 huruf yang ditampilkan pada Tabel 4.3. Dalam Tabel 4.3 citra uji merupakan citra yang akan dikenali dengan sistem, sedangkan hasil menunjukkan hasil pengenalan yang dilakukan oleh sistem.

**Tabel 4.3.** Hasil uji coba proses testing dengan input huruf tunggal

No	Citra uji coba	Hasil	Status	No	Citra uji coba	Hasil	Status
1	a	a	Benar	53	n	n	Benar
2	a	a	Benar	54	n	n	Benar
3	a	d	Salah	55	n	m	Salah
4	a	f	Salah	56	n	h	Salah
5	b	b	Benar	57	o	o	Benar
6	b	b	Benar	58	o	o	Benar
7	b	l	Salah	59	o	o	Benar
8	b	b	Benar	60	o	b	Salah
9	c	c	Benar	61	p	p	Benar
10	c	c	Benar	62	p	p	Benar
11	c	c	Benar	63	p	f	Salah
12	c	c	Benar	64	p	p	Benar
13	d	d	Benar	65	q	q	Benar

No	Citra uji coba	Hasil	Status	No	Citra uji coba	Hasil	Status
14	d	d	Benar	66	q	q	Benar
15	d	d	Benar	67	q	q	Benar
16	d	j	Salah	68	q	j	Salah
17	e	e	Benar	69	r	r	Benar
18	e	e	Benar	70	r	r	Benar
19	e	e	Benar	71	r	r	Benar
20	e	t	Salah	72	r	r	Benar
21	f	f	Salah	73	s	s	Benar
22	F	t	Benar	74	S	s	Benar
23	F	f	Benar	75	S	l	Salah
24	f	f	Benar	76	S	f	Salah
25	g	g	Benar	77	t	t	Benar
26	g	g	Benar	78	t	t	Benar
27	g	g	Benar	79	t	f	Salah
28	g	g	Benar	80	t	t	Benar
29	h	h	Benar	81	u	u	Benar
30	h	h	Benar	82	u	j	Salah
31	h	h	Benar	83	u	u	Benar
32	h	h	Benar	84	u	u	Benar

No	Citra uji coba	Hasil	Status	No	Citra uji coba	Hasil	Status
33	i	i	Benar	85	v	v	Benar
34	i	i	Benar	86	v	v	Benar
35	i	i	Benar	87	v	v	Benar
36	l	f	Salah	88	v	v	Benar
37	j	j	Benar	89	w	w	Benar
38	j	j	Benar	90	w	w	Benar
39	J	j	Benar	91	w	v	Salah
40	j	j	Benar	92	w	w	Benar
41	K	k	Benar	93	x	x	Benar
42	K	k	Benar	94	x	x	Benar
43	R	l	Salah	95	x	x	Benar
44	K	k	Benar	96	x	f	Salah
45	l	l	Benar	97	y	y	Benar
46	l	l	Benar	98	y	j	Salah
47	l	l	Benar	99	y	y	Benar
48	f	l	Benar	100	y	y	Benar
49	m	m	Benar	101	z	z	Benar
50	m	m	Benar	102	z	z	Benar

No	Citra uji coba	Hasil	Status	No	Citra uji coba	Hasil	Status
51		m	Benar	103		z	Benar
52		k	Salah	104		Z	Benar

Hasil uji coba proses *testing* dengan input berupa kata dan kalimat dapat dilihat pada Tabel 4.4. citra uji coba pada tabel menunjukkan citra tulisan tangan yang akan dikenali yang berupa kata dan kalimat, pada hasil pengenalan menunjukkan hasil yang dikenali oleh sistem dari citra yang diuji, hasil pengenalan terdapat 2 macam yaitu hasil pengenalan tanpa ROI (*Region of Interest*) dan hasil pengenalan dengan tambahan ROI setelah proses *cropping* dari hasil *Projection Profile*, dan jumlah benar menunjukkan jumlah huruf yang berhasil dikenali dengan benar baik dengan ROI maupun tanpa ROI. Berdasarkan hasil uji coba dengan input berupa kata dan kalimat yang terdiri dari 173 huruf dan tanpa ROI diperoleh akurasi 27,75% dengan jumlah jawaban benar sebanyak 48 dan jawaban salah sebanyak 125 huruf, sedangkan dengan inputan yang sama dan dengan tambahan ROI diperoleh akurasi sebesar 63,01% yang terdiri dari 109 huruf yang berhasil dikenali dengan benar dan 64 huruf yang tidak dapat dikenali dengan benar. Gambar 4.10 menunjukkan proses uji coba dengan input berupa kata dengan memilih file citra yang akan diuji dan memilih tombol preproses dan *testing*.



**Gambar 4.10.** Uji coba proses *testing*

Pada uji coba proses *testing* dengan *input* berupa citra kata atau satu baris kalimat, pada preproses akan dilakukan *Projection Profile* untuk mencari batas baris dan batas kolom dari setiap huruf pada citra kata yang akan di uji. Gambar 4.11 menunjukkan hasil *Projection Profile* pada salah satu contoh citra uji.



a. Citra hasil *vertical projection*



b. Citra hasil *horizontal projection*



c. Citra hasil *cropping* berdasarkan hasil *Projection Profile*

**Gambar 4.11** Citra uji hasil *Projection Profile*

Untuk hasil *cropping* dan *Projection Profile* pada seluruh citra uji yang terdapat pada Tabel 4.4 dapat dilihat pada lampiran 1.

Tabel 4.4. Hasil uji coba proses *testing* dengan *input* kata dan kalimat

No	Citra uji coba	Hasil pengenalan		Jumlah benar	
		Tanpa ROI	ROI	Tanpa ROI	ROI
1	semoga sukses	rrrrgqrhmdel	gerogasukdes	2	9
2	Semangat skripsi	wkhhfhkffffxplf	semdrqelsxrlpsi	1	8
3	ethiopia	kkkflfk	ebbioria	0	5
4	nisa	nela	nesd	2	2
5	universitas Islam	unkkwndyfudihtdw	uneveryqtultreoh	3	6
6	negeri malang	qfffrfmfkffd	neyeremrerng	2	7
7	sama	zama	zama	3	3
8	dan	ddd	dun	1	2
9	berikut	btkikuk	berikub	4	6

No	Citra uji coba	Hasil pengenalan		Jumlah benar	
		Tanpa ROI	ROI	Tanpa ROI	ROI
10	MAKANAN	mnhhnqn	mknrnqn	3	4
11	SORA	sora	sord	4	3
12	hamba	fdwhd	homha	0	3
13	nama	kamq	raaa	2	2
14	mozambique	mmqkmlkww	qozrrblqde	2	5
15	kyrgyzstan	kkfqhxlkn	karqhzlan	2	5
16	luxembourg	lwxlqluhrq	luxeklourq	3	10
17	bagi	ffyx	bqyi	0	2
18	nulis	nuels	nuelq	3	2
19	jihyo	jihuk	gthyo	3	3

No	Citra uji coba	Hasil pengenalan		Jumlah benar	
		Tanpa ROI	ROI	Tanpa ROI	ROI
20	dzikir	dtiklk	dzikir	3	6
21	pokok	lfkxf	pokok	0	5
22	negara	rqqqrm	reqrra	1	3
23	dt suruh	dkwwkdh	dequruh	2	5
24	dunia	dddid	ddnir	2	3

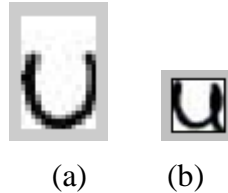
#### 4.4.Pembahasan

Berdasarkan uji coba yang telah dilakukan dengan data *training* sebanyak 468 data dengan masing – masing huruf terdiri dari 18 sampel serta uji coba dengan beberapa kombinasi dimensi pixel baris dan kolom, diperoleh dimensi pixel citra terbaik adalah 7x4 dengan akurasi training sebesar 81,1966 %. Dan berdasarkan uji coba testing dengan inputan berupa huruf tunggal sebanyak 104 data uji diperoleh jawaban benar sebanyak 83 dan jawaban salah sebanyak 21 dengan akurasi testing sebesar 79,81 %.

Sedangkan berdasarkan hasil uji coba dengan input berupa kata dan kalimat yang terdiri dari 173 huruf dan tanpa ROI diperoleh akurasi 27,75% dengan jumlah jawaban benar sebanyak 48 dan jawaban salah sebanyak 125 huruf, sedangkan dengan inputan yang sama dan dengan tambahan ROI diperoleh akurasi yang lebih baik yaitu 63,01% dengan 109 huruf yang berhasil dikenali dengan benar dan 64 huruf yang tidak dapat dikenali dengan benar.

Pada hasil uji coba tanpa tambahan ROI diperoleh akurasi yang lebih rendah jika dibandingkan dengan hasil pengenalan dengan tambahan ROI, hal ini dikarenakan hasil *Projection Profile* pada citra *testing* yang tidak menyerupai citra *training*. Pada hasil *cropping* berdasarkan *Projection Profile* pada citra *testing*, *cropping* huruf yang dihasilkan memiliki bagian putih yang lebih banyak jika dibandingkan dengan citra *training* seperti contoh yang ditunjukkan pada Gambar 4.12. Oleh karena itu, perlu adanya penambahan ROI untuk menghilangkan bagian putih pada citra testing agar hasil *cropping* citra yang akan dikenali bisa menyerupai citra testing, sehingga proses pengenalan dapat dilakukan dengan

baik. Gambar 4.12(a) menunjukkan citra uji coba yang di – *cropping* dengan *Projection Profile* dan Gambar 4.12(b) menunjukkan citra training.



**Gambar 4.12.** (a) hasil *cropping* citra *testing* (b) citra *training*

Berdasarkan hasil analisa terhadap data uji coba diketahui bahwa akurasi sistem sangat dipengaruhi oleh data *training* dan dimensi pixel citra. Data *training* berperan penting dalam proses pembelajaran sistem agar dapat mengenali citra inputan, oleh karena itu diperlukan data terbaik dengan dimensi terbaik untuk memperoleh akurasi terbaik.

Tujuan utama digunakannya metode *Naïve Bayes Classifier* dalam penelitian adalah untuk dapat mengklasifikasikan citra input pada salah satu sampel pada data training. Hal juga diajarkan dalam *al – Qur'an* surat *al – Hujurat* ayat 13.

يَتَأْتِيهَا النَّاسُ إِنَّا خَلَقْنَاهُمْ مِّن ذَكَرٍ وَأُنثَىٰ وَجَعَلْنَاكُمْ شُعُوبًا وَقَبَائِلَ لِتَعَارَفُوا ۗ إِنَّ  
 أَكْرَمَكُمْ عِنْدَ اللَّهِ أَتَقَىٰكُمْ ۗ إِنَّ اللَّهَ عَلِيمٌ خَبِيرٌ ﴿١٣﴾

Artinya : Hai manusia, Sesungguhnya kami menciptakan kamu dari seorang laki-laki dan seorang perempuan dan menjadikan kamu berbangsa - bangsa dan bersuku-suku supaya kamu saling kenal-mengenal. Sesungguhnya orang yang paling mulia diantara kamu disisi Allah ialah orang yang paling taqwa diantara kamu. Sesungguhnya Allah Maha mengetahui lagi Maha Mengenal (Q.S. *al – Hujurat* : 13).

Surat *al – Hujurat* ayat 13. ditujukan kepada umat manusia seluruhnya, tak hanya kepada kaum muslimin, bahwa manusia diturunkan dari sepasang suami-istri. Suku, ras dan bangsa mereka merupakan nama-nama saja untuk memudahkan, sehingga dengan itu kita dapat mengenali perbedaan sifat-sifat tertentu. Di hadapan Allah SWT mereka semua satu, dan yang paling mulia ialah yang paling bertakwa (Chirzin, 2011). Dalam *al – Qur'an* surat *al – Hujurat* ayat 13, secara tersirat Allah mengajarkan mengenai klasifikasi dengan menciptakan manusia dari seorang laki-laki (Adam) dan seorang perempuan (Hawa), dan menjadikannya berbangsa-bangsa, bersuku-suku, dan berbeda-beda warna kulit. Selain itu dalam dalam surat *ar – Ruum* ayat 22 Allah juga menyiratkan mengenai klasifikasi dengan menciptakan bahasa dan warna kulit yang berbeda – beda.

وَمِنْ آيَاتِهِ ۚ خَلَقَ السَّمَوَاتِ وَالْأَرْضِ وَأَخْتَلَفُ اللَّسَانِكُمْ وَأَلْوَانِكُمْ ۗ إِنَّ فِي ذَلِكَ لَآيَاتٍ لِّلْعَالَمِينَ ﴿٢٢﴾

Artinya : Dan di antara tanda-tanda kekuasaan-Nya ialah menciptakan langit dan bumi dan berlain-lainan bahasamu dan warna kulitmu. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda-tanda bagi orang-orang yang Mengetahui (Q.S. *ar – Ruum* : 22).

Dalam *al – quran* surat *an – Nisa'* ayat 4, 11 dan 12 dijelaskan mengenai pembagian warisan.

وَأَتُوا النِّسَاءَ صَدُقَاتِهِنَّ مِحْلَةً ۚ فَإِن طِبَّنَ لَكُمْ عَنْ شَيْءٍ مِّنْهُ نَفْسًا فَكُلُوهُ هَنِيئًا مَّرِيئًا



Artinya : Berikanlah maskawin (mahar) kepada wanita (yang kamu nikahi) sebagai pemberian dengan penuh kerelaan. Kemudian jika mereka menyerahkan kepada kamu sebagian dari maskawin itu dengan senang hati, Maka makanlah

(ambillah) pemberian itu (sebagai makanan) yang sedap lagi baik akibatnya (Q.S. *an – Nisa' : 4*).

يُوصِيكُمُ اللَّهُ فِي أَوْلَادِكُمْ لِلذَّكَرِ مِثْلُ حَظِّ الْأُنثِيَّاتِ ۚ فَإِن كُنَّ نِسَاءً فَوْقَ اثْنَتَيْنِ فَلَهُنَّ ثُلُثَا مَا تَرَكَ ۚ وَإِن كَانَتْ وَاحِدَةً فَلَهَا النِّصْفُ ۚ وَلِأَبَوَيْهِ لِكُلِّ وَاحِدٍ مِّمَّهَا السُّدُسُ ۚ مِمَّا تَرَكَ إِن كَانَ لَهُ وَلَدٌ ۚ فَإِن لَّمْ يَكُن لَّهُ وَلَدٌ وَوَرِثَهُ أَبُوَاهُ فَلِأُمِّهِ الثُّلُثُ ۚ فَإِن كَانَ لَهُ إِخْوَةٌ فَلِأُمِّهِ السُّدُسُ ۚ مِّن بَعْدِ وَصِيَّةٍ يُوصِي بِهَا أَوْ دَيْنٍ ۚ لِأَبَائِكُمْ وَأَبْنَاؤِكُمْ لَا تَدْرُونَ أَيُّهُمْ أَقْرَبُ لَكُمْ نَفَعًا ۚ فَرِيضَةٌ مِّنَ اللَّهِ ۚ إِنَّ اللَّهَ كَانَ عَلِيمًا حَكِيمًا ﴿١١﴾

Artinya : Allah mensyariatkan bagimu tentang (pembagian pusaka untuk) anak-anakmu. yaitu : bahagian seorang anak lelaki sama dengan bagahian dua orang anak perempuan dan jika anak itu semuanya perempuan lebih dari dua, Maka bagi mereka dua pertiga dari harta yang ditinggalkan; jika anak perempuan itu seorang saja, Maka ia memperoleh separo harta. dan untuk dua orang ibu-bapa, bagi masing-masingnya seperenam dari harta yang ditinggalkan, jika yang meninggal itu mempunyai anak; jika orang yang meninggal tidak mempunyai anak dan ia diwarisi oleh ibu-bapanya (saja), Maka ibunya mendapat sepertiga; jika yang meninggal itu mempunyai beberapa saudara, Maka ibunya mendapat seperenam. (Pembagian-pembagian tersebut di atas) sesudah dipenuhi wasiat yang ia buat atau (dan) sesudah dibayar hutangnya. (Tentang) orang tuamu dan anak-anakmu, kamu tidak mengetahui siapa di antara mereka yang lebih dekat (banyak) manfaatnya bagimu. Ini adalah ketetapan dari Allah. Sesungguhnya Allah Maha mengetahui lagi Maha Bijaksana (Q.S. *an – Nisa' : 11*).

وَلَكُمْ نِصْفُ مَا تَرَكَ أَزْوَاجُكُمْ إِن لَّمْ يَكُن لَّهُنَّ وَلَدٌ ۚ فَإِن كَانَ لَهُنَّ وَلَدٌ فَلَكُمْ الرُّبْعُ مِمَّا تَرَكَنَّ ۚ مِّن بَعْدِ وَصِيَّةٍ يُوصِينَ بِهَا أَوْ دَيْنٍ ۚ وَلَهُنَّ الرُّبْعُ مِمَّا تَرَكَتُمْ إِن لَّمْ يَكُن لَّكُمْ وَلَدٌ ۚ فَإِن كَانَ لَكُمْ وَلَدٌ فَلَهُنَّ الثُّمْنُ مِمَّا تَرَكَتُمْ ۚ مِّن بَعْدِ وَصِيَّةٍ تُوصُونَ بِهَا أَوْ دَيْنٍ ۚ وَإِن كَانَ رَجُلٌ يُورِثُ كَلَلَةً

أَوْ امْرَأَةً وَلَهُ إِخٌ أَوْ أُخْتُ فَلِكُلِّ وَاحِدٍ مِّنْهُمَا السُّدُسُ فَإِن كَانُوا أَكْثَرَ مِن  
ذَلِكَ فَهُمْ شُرَكَاءُ فِي الثُّلُثِ مِّنْ بَعْدِ وَصِيَّةٍ يُوصَىٰ بِهَا أَوْ دَيْنٍ غَيْرِ مُضَارٍّ وَصِيَّةً  
مِّنَ اللَّهِ وَاللَّهُ عَلِيمٌ حَلِيمٌ ﴿١٢﴾

Artinya : Dan bagimu (suami-suami) seperdua dari harta yang ditinggalkan oleh isteri-isterimu, jika mereka tidak mempunyai anak. jika Isteri-isterimu itu mempunyai anak, Maka kamu mendapat seperempat dari harta yang ditinggalkannya sesudah dipenuhi wasiat yang mereka buat atau (dan) sesudah dibayar hutangnya. para isteri memperoleh seperempat harta yang kamu tinggalkan jika kamu tidak mempunyai anak. jika kamu mempunyai anak, Maka para isteri memperoleh seperdelapan dari harta yang kamu tinggalkan sesudah dipenuhi wasiat yang kamu buat atau (dan) sesudah dibayar hutang-hutangmu. jika seseorang mati, baik laki-laki maupun perempuan yang tidak meninggalkan ayah dan tidak meninggalkan anak, tetapi mempunyai seorang saudara laki-laki (seibu saja) atau seorang saudara perempuan (seibu saja), Maka bagi masing-masing dari kedua jenis saudara itu seperenam harta. tetapi jika Saudara-saudara seibu itu lebih dari seorang, Maka mereka bersekutu dalam yang sepertiga itu, sesudah dipenuhi wasiat yang dibuat olehnya atau sesudah dibayar hutangnya dengan tidak memberi mudharat (kepada ahli waris). (Allah menetapkan yang demikian itu sebagai) syari'at yang benar-benar dari Allah, dan Allah Maha mengetahui lagi Maha Penyantun (Q.S. *an – Nisa'* : 12).

Makna yang terkandung dalam *al – quran* surat *an – Nisa'* ayat 4, 11 dan 12 mengenai pembagian warisan dapat dijadikan rujukan dan landasan dari penggunaan ROI. ROI bertujuan untuk memperoleh bagian tertentu pada citra uji sehingga diperoleh bagian yang hanya dibutuhkan dan membuang bagian yang tidak dibutuhkan sehingga proses selanjutnya dapat berjalan dengan baik.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Dari serangkaian tahapan penelitian yang telah dilakukan, dapat diperoleh kesimpulan sebagai berikut :

1. Untuk membangun aplikasi pengenalan tulisan tangan diperlukan data tulisan tangan sebagai data training selanjutnya perlu dilakukan preproses dan proses *training* untuk pembelajaran sistem sehingga akan diperoleh nilai estimasi parameter dari data training berupa mean dan varian, setelah itu dilakukan dilakukan proses pengenalan dengan inputan berupa citra yang ingin dikenali dan hasil estimasi parameter pada proses *training* menggunakan metode *Naïve Bayes Classifier*.
2. Berdasarkan hasil uji coba yang dilakukan dengan data *training* sebanyak 468 citra dengan masing – masing huruf terdiri dari 18 sampel diperoleh akurasi *training* sebesar 81,1966 % dengan ukuran dimensi pixel 7x4. Dan berdasarkan uji coba testing dengan data uji coba berupa huruf tunggal sebanyak 104 data diperoleh hasil 83 jawaban benar dan 21 jawaban salah dengan akurasi *testing* sebesar 79,81 %. Sedangkan berdasarkan hasil uji coba dengan input berupa kata dan kalimat yang terdiri dari 173 huruf dan tanpa ROI diperoleh akurasi 27,75% dengan jumlah jawaban benar sebanyak 48 dan jawaban salah sebanyak 125 huruf, sedangkan dengan inputan yang sama dan dengan tambahan ROI diperoleh akurasi yang lebih baik yaitu 63,01% dengan

109 huruf yang berhasil dikenali dengan benar dan 64 huruf yang tidak dapat dikenali dengan benar. Pada hasil uji coba dengan input berupa huruf dan kata tanpa tambahan ROI diperoleh akurasi yang lebih rendah jika dibandingkan dengan hasil pengenalan dengan tambahan ROI, hal ini dikarenakan hasil *Projection Profile* pada citra *testing* yang tidak menyerupai citra *training*.

## 5.2. Saran

Untuk pengembangan aplikasi pengenalan tulisan tangan ini di kemudian hari diperlukan beberapa perbaikan dan tambahan untuk mendapatkan hasil yang lebih maksimal, diantaranya :

1. Memperbanyak objek yang dapat dikenali, misalnya dengan menambahkan huruf kapital dan angka.
2. Penelitian dapat dikembangkan dengan objek berupa tulisan tangan latin (bersambung) yang memiliki tingkat kesulitan yang lebih tinggi dalam proses pemecahan kata.
3. Melakukan penelitian dengan mencoba berbagai metode yang lebih beragam.

## DAFTAR PUSTAKA

- Al-Albani, Muhammad Nashiruddin. 2010. *Ringkasan Sahih Bukhari*. Jakarta : Pustaka as – Sunnah
- Aribowo, Trilaksono. 2010. *Aplikasi Inferensi Bayes pada Data Mining terutama Pattern Recognition*. Bandung : Jurnal ITS bidang Sistem dan Teknik Informasi.
- Arief, Ahmad Fauzi. 2009. *Perangkat Lunak pengkonversi Teks Tulisan Tangan Menjadi Teks Digital*. Surabaya : Jurnal Sarjana ITS bidang Matematika.
- Darmawan, Indra. 2011. *Mengetik Membuat Anak Menjadi Lebih Bodoh?*. teknologi.news.viva.co.id/news/read/201490-mengetik-membuat-anak-menjadi-lebih-bodoh- (diakses pada tanggal 02 Agustus 2014).
- Emanuel, Andi Wahyu Rahardjo dan Hartono, Arie. 2007. *Pengembangan Aplikasi Pengenalan Karakter Alfanumerik dengan Menggunakan Algoritma Neural Network Three-layer Back Propagation*. Bandung : Jurnal Sarjana Universitas Kristen Maranatha bidang Teknik Informatika.
- Fifin Hietania, Wayan Santiyasa dan Ida Bagus Gede Dwidasmara. 2012. *Implementasi dari Algoritma Backpropagatin dalam Pengolahan Citra Teks Tulisan Tangan Menjadi Teks Digital*. Bali : Jurnal Sarjana Universitas Udayana bidang Teknik Informatika.
- Godam, 2006. *Macam-Macam Huruf atau Aksara dalam Bahasa Indonesia*. <http://www.organisasi.org/1970/01/macam-macam-huruf-atau-aksara-dalam-bahasa-indonesia.html> (diakses pada tanggal 05 Desember 2014).
- Hamzah, A. 2012. Klasifikasi Teks dengan Naive Bayes Classifier (NBC) untuk Pengelompokkan Teks Berita dan Abstrak Akademis. *Seminar Nasional Aplikasi Sains & Teknologi (NAST) Periode III*, (hal. B-269). Yogyakarta.
- Harini, S. dan Kusumawati. 2007. *Metode Statistik*. Jakarta : Prestasi Pustaka.
- Hastiana, Rumi. 2010. *Segmentasi Citra Digital Pembuluh Darah Mata Untuk Mendeteksi Tingkat Keparahan Diabetic Retinopathy*. Malang : Skripsi UIN Maulana Malik Ibrahim bidang Teknik Informatika.
- Hendry, Jans. 2011. *Using Profile Projection to Segmen Character in Image (Matlab)*. Yogyakarta : EE & IT UGM.

- Ilham, Muhammad. 2013. *4 Manfaat Menulis Dengan Tangan*. <http://oalamagz.blogspot.com/2013/01/4-manfaat-menulis-dengan-tangan.html> (diakses pada tanggal 02 Agustus 2014).
- Munir, Rinaldi. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung : Informatika.
- Natalius, Samuel. 2010. *Metoda Naive Bayes Classifier dan penggunaannya pada Klasifikasi Dokumen*. Bandung : Jurnal ITB bidang Sistem dan Teknik Informasi.
- Nurchahyo, Prasetya, Hendra. 2010. Pengertian JPEG, BMP, GIF, TIFF, PNG. <http://ilmucomputerhendra.blogspot.com/2010/11/pengertian-jpeg-bmp-gif-tiff-png.html> (diakses pada tanggal 05 Desember 2014).
- Plamondon, Rejean and Srihari, Sargur N. 2000. *On-Line and Off-Line Handwriting Recognition : a Comprehensive Survey*. IEEE.
- Prasetyo, Eko. 2011. *Pengolahan Citra Digital dan Aplikasinya menggunakan Matlab*. Yogyakarta : Penerbit ANDI.
- Prasojo, Andi. 2011. *Pengenalan Karakter Alfabet Menggunakan Jaringan Saraf Tiruan*. Semarang : Jurnal Sarjana Universitas Diponegoro bidang Teknik Elektro.
- Prof Dr H. Muhammad Chirzin. Mag. 2011. *Kerukunan Hidup Beragama dalam Perspektif Al-Qur'an*. <http://www.muhammadiyah.or.id/7-content-190-det-tafsir-alcquran.html> (diakses pada tanggal 04 Desember 2014).
- Purnomo, Mauridhi Hery dan Muntasa, Arif. 2010. *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Yogyakarta : Graha Ilmu.
- Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta : Penerbit Andi.
- R.A. Uluwiyah Nur O dan Kusumaningsih, Ari. 2013. *Deteksi Manusia dengan Menggunakan Histogram of Oriented Gradients Naive Bayes Classifier*. Madura : Jurnal Universitas Trunojoyo bidang Teknik Informatika.
- Roberto J. Rodrigues and Antonio Carlos Gay Thomé. 2002. *Cursive Character Recognition – a Character Segmentation Method Using Projection Profile-Based Technique*. Brasil : Universidade Federal do Rio de Janeiro.
- Safitri, Hilyati, Damayanti, Fitri, dan Kurniawan Eka Permana. 2012. *Pengenalan Tulisan Tangan Huruf Alfabet Dengan Metode Modified Direction*

*Feature (MDF) Dan Learning Vector Quantization (LVQ)*. Madura :  
Jurnal Universitas Trunojoyo bidang Teknik Informatika.

Sinambela, Dompok Petrus dan Sitorus, Sampe Hotlan. 2013. *Pengenalan Karakter Tulisan Tangan Latin pada Jaringan Saraf Tiruan Metode Backpropagation dengan Input Citra Kamera Digital*. Jakarta : Jurnal Sarjana Universitas Mpu Tantular.

Sugiantoro, Hendra. *Menulis, Tradisi Islam!*. baticnews.com (diakses pada tanggal 06 November 2014)

Sulistiyawati, Anik. 2014. *Tulis Tangan Ternyata Bisa Memperkuat Ingatan*. solopos.com/2014/02/08/tulis-tangan-ternyata-bisa-memperkuat-ingatan-487992 (diakses pada tanggal 02 Agustus 2014)

Sutoyo, T, S.Si, M.Kom., dkk. 2009. *Teori Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi.



Lampiran

1. Hasil *Projection Profile* dan *cropping* citra testing

semoga sukses

S e m o g a s u k s e s

Semangat skripsi

S e m a n g a t s k r i p s i

ethiopia

e t h i o p i a

nisa

n i s a

universitas Islam

u n i v e r s i t a s i s l a m

negeri malang

n e g e r i m a l a n g

sama

s a m a

dan

d a n

berikut

b e r i k u t

makanan

m a k a n a n

sora

s o r a

hamba

h a m b a

h a m a

h a m a

m o z a m b i q u e

m o z a m b i q u e

k u r g u z t a n

k u r g u z t a n

l u x e m b o u r g

l u x e m b o u r g

baqi

b a q i

nulis

n u l i s

jihyo

j i h y o

dunia

d u n i a

dzikir

d z i k i r

pokok

p o k o k

negara

n e g a r a

dit suruh

d i t s u r u h

## 2. Parameter mean dan parameter varian

parameter\_mean(:,,1) =

Columns 1 through 9

246.8889 183.6667 108.7778 105.1667 95.6667 125.0556 225.7778 175.3889 78.0000

Columns 10 through 18

197.6667 248.2778 163.3333 69.2778 225.2222 164.5556 76.8889 137.1667 119.2778

Columns 19 through 27

87.1667 189.3889 245.9444 234.7222 176.0000 142.3889 126.9444 115.4444 127.5556

Column 28

198.4444

parameter\_mean(:,,2) =

Columns 1 through 9

170.0556 128.8889 143.2778 149.5556 157.3333 181.5556 222.7222 218.2222 213.1111

Columns 10 through 18

182.0000 93.5556 122.2778 122.7778 126.0556 254.9444 250.3333 175.4444 136.8333

Columns 19 through 27

216.4444 148.2778 155.9444 254.6111 254.0556 220.2222 143.0000 116.7778 160.3333

Column 28

235.0000

parameter\_mean(:,,3) =

Columns 1 through 9

245.6111 188.4444 124.8333 105.7222 106.5556 147.1111 226.7222 149.4444 90.4444

Columns 10 through 18

178.0000 226.4444 214.1667 100.5000 113.4444 114.6667 137.5556 236.8889 254.5000

Columns 19 through 27

244.2778 126.7222 111.3333 225.1667 217.7778 244.8889 252.6667 225.3333 151.5000

Column 28

198.0556

parameter\_mean(:,,4) =

Columns 1 through 9

254.7778 254.5556 253.6111 196.4444 130.2222 112.8333 185.8889 250.8889 250.8333

Columns 10 through 18

226.2222 142.5000 204.3889 172.1667 143.8333 202.7222 171.7222 162.3889 106.8889

Columns 19 through 27

146.5556 137.8333 200.1667 190.6111 158.2778 155.5556 147.6111 129.7778 128.2222

Column 28

156.6667

parameter\_mean(:,5) =

Columns 1 through 9

238.6667 162.6667 112.0556 82.9444 106.7222 170.8333 240.5556 122.0556 96.0556

Columns 10 through 18

138.7778 63.1667 147.0000 89.1111 132.5556 126.1111 109.1667 120.6111 124.5556

Columns 19 through 27

235.5000 124.5000 103.7222 235.3889 187.6111 182.0556 231.5000 233.1111 143.3333

Column 28

192.6111

parameter\_mean(:,6) =

Columns 1 through 9

217.4444 194.1111 191.9444 180.9444 215.0000 230.1667 226.4444 165.7222 151.6667

Columns 10 through 18

138.5556 110.5556 167.7222 193.7222 193.0000 139.6667 201.2222 192.2778 173.5556

Columns 19 through 27

217.1667 206.5000 207.7222 205.2778 222.4444 237.2222 230.9444 254.0000 250.6111

Column 28

245.6667

parameter\_mean(:,7) =

Columns 1 through 9

239.5000 161.5000 132.6667 193.4444 247.8889 240.3333 222.3333 136.6667 159.0000

Columns 10 through 18

169.7222 140.5556 241.6111 226.5556 149.4444 127.0556 138.2778 130.0000 167.6111

Columns 19 through 27

214.8333 157.5556 156.2778 216.1667 160.9444 127.9444 125.0000 129.7778 167.3889

Column 28

232.5000

parameter\_mean(:,8) =

Columns 1 through 9

179.3333 127.2778 136.2222 149.3333 153.7778 174.7778 211.5556 230.7778 223.5556

Columns 10 through 18

212.1667 126.4444 125.8889 165.0556 217.3333 254.2222 254.9444 218.7778 116.8889

Columns 19 through 27

189.7222 227.0000 227.6667 254.6111 254.6667 244.3333 190.5000 145.1667 134.2222

Column 28

164.7778

parameter\_mean(:,9) =

Columns 1 through 9

231.8333 212.3889 235.8333 220.6111 223.7222 226.4444 239.1111 176.5556 171.1111

Columns 10 through 18

189.1667 80.0556 73.5000 116.3333 181.6667 150.9444 167.8333 198.3889 110.9444

Columns 19 through 27

94.2778 98.2222 144.5556 219.2778 227.8333 248.8333 244.1111 236.3889 216.8333

Column 28

227.5000

parameter\_mean(:,10) =

Columns 1 through 9

252.0556 246.7222 252.8333 254.2778 250.0000 221.7778 207.3333 203.9444 222.1667

Columns 10 through 18

231.4444 245.3889 251.7222 230.2778 126.7222 155.6667 164.8333 149.0556 165.3889

Columns 19 through 27

179.6111 164.0556 137.7778 237.4444 233.8333 221.9444 208.6111 191.3333 189.6111

Column 28

229.6111

parameter\_mean(:,11) =

Columns 1 through 9

175.0000 118.2222 113.0556 110.8333 130.4444 157.5556 199.8889 229.2778 227.3889

Columns 10 through 18

178.2222 93.7222 97.7778 151.5556 214.5556 254.3333 231.3889 155.0556 142.0000

Columns 19 through 27

149.9444 170.4444 235.2222 253.2222 239.4444 190.8333 216.1667 194.2222 185.9444

Column 28

226.4444

parameter\_mean(:,12) =

Columns 1 through 9

235.5556 192.1111 189.8889 220.3889 238.0000 249.1667 253.3333 178.1667 87.1111

Columns 10 through 18

69.7222 73.5000 97.3889 154.0000 220.8333 213.9444 178.6667 188.8889 176.5556

Columns 19 through 27

142.2778 99.0556 163.6111 251.7222 250.6111 253.9444 254.1111 248.9444 199.4444

Column 28

202.6667

parameter\_mean(:,13) =

Columns 1 through 9

208.8889 80.8889 91.8889 122.8889 129.0000 153.9444 218.0000 219.2778 103.7222

Columns 10 through 18

133.7222 174.7222 183.3889 205.7778 243.1667 229.7222 146.6111 127.3333 141.1667

Columns 19 through 27

164.7778 211.2222 247.2222 219.5000 138.8333 131.8889 148.3889 152.0556 166.1111

Column 28

213.6667

parameter\_mean(:,14) =

Columns 1 through 9

205.7222 107.5000 80.9444 98.5000 117.1111 139.5000 211.3889 179.3333 100.2778

Columns 10 through 18

156.1111 192.7778 191.0556 189.1111 227.9444 145.7222 88.4444 184.4444 221.6667

Columns 19 through 27

229.5556 241.0000 249.9444 226.5556 174.9444 136.0000 125.6667 114.7778 119.0556

Column 28

197.1111

parameter\_mean(:,15) =

Columns 1 through 9

247.5000 188.1667 121.5556 105.9444 105.5000 129.2222 219.3889 146.0000 46.5000

Columns 10 through 18

153.1111 226.6667 228.0000 114.3333 95.1667 132.8333 87.0000 203.6111 238.1667

Columns 19 through 27

204.5556 88.6111 153.2778 232.8333 165.0000 126.0000 121.0000 118.6667 175.6667

Column 28

248.6111

parameter\_mean(:,16) =

Columns 1 through 9

192.7222 132.3333 156.7778 176.7778 194.4444 207.3889 219.8333 115.0556 143.7778

Columns 10 through 18

180.7222 108.7222 132.7778 155.5000 157.8889 110.3333 181.2222 178.7222 141.0556

Columns 19 through 27

226.0556 241.4444 231.7222 208.8333 141.3333 157.8889 223.6111 253.8889 254.5000

Column 28

254.7222

parameter\_mean(:,17) =

Columns 1 through 9

231.6667 151.1111 115.7778 157.2222 235.3333 250.2778 253.5000 131.5000 152.6667

Columns 10 through 18

174.0556 144.3889 223.9444 230.2222 235.4444 120.5000 120.9444 107.1111 138.8889

Columns 19 through 27

174.1667 179.0000 199.1111 197.1667 175.2222 183.8333 190.5000 170.1667 145.6111

Column 28

179.9444

parameter\_mean(:,18) =

Columns 1 through 9

200.3889 115.5556 99.0556 123.6667 147.5556 175.9444 219.8889 127.1111 70.1111

Columns 10 through 18

143.5556 160.0556 145.3889 124.5556 172.8333 105.6667 104.1111 236.9444 252.8889

Columns 19 through 27

251.0000 246.2778 250.1667 192.6667 161.5556 238.7222 253.5000 252.8889 253.5556

Column 28

254.1667

parameter\_mean(:,,19) =

Columns 1 through 9

236.1667 147.8889 120.2778 166.5556 237.4444 209.7778 168.7222 143.3333 105.5000

Columns 10 through 18

173.0556 115.0556 164.0000 156.3333 103.6667 108.5556 160.8889 245.1111 200.5556

Columns 19 through 27

120.4444 82.3333 158.4444 201.6111 182.0556 240.9444 237.4444 191.0556 175.3889

Column 28

235.2222

parameter\_mean(:,,20) =

Columns 1 through 9

225.9444 205.3889 172.6111 135.4444 195.6667 241.8889 252.6667 178.3333 129.4444

Columns 10 through 18

100.7778 68.1111 121.2222 119.0000 182.5556 243.5000 230.9444 193.1111 144.6667

Columns 19 through 27

227.7778 162.7778 138.8333 254.4444 248.3889 231.7778 213.9444 234.5556 180.8333

Column 28

202.6667

parameter\_mean(:,,21) =

Columns 1 through 9

203.7222 129.4444 98.6111 105.1667 127.7222 182.8889 237.5556 232.2778 212.5000

Columns 10 through 18

222.7222 223.1111 175.6111 86.6111 162.5556 234.8889 209.6667 208.2778 206.0556

Columns 19 through 27

157.5000 89.0556 188.0556 214.6111 149.1111 112.3333 114.0000 112.8889 138.6111

Column 28

202.9444

parameter\_mean(:, :, 22) =

Columns 1 through 9

210.7222 151.1667 129.5556 166.5556 206.4444 234.8889 249.2778 244.3889 226.8333

Columns 10 through 18

197.3889 163.7222 118.3889 81.5556 166.7222 229.1111 201.1667 179.3889 148.9444

Columns 19 through 27

124.5556 135.5556 201.6667 170.7222 132.6111 174.7778 207.8889 233.8889 250.6667

Column 28

254.1667

parameter\_mean(:, :, 23) =

Columns 1 through 9

247.8889 197.7778 135.9444 138.8889 138.7222 142.8889 228.8889 253.3889 237.8889

Columns 10 through 18

190.4444 147.6111 111.1111 131.3889 225.5556 253.0000 244.0556 215.1667 188.0000

Columns 19 through 27

137.6667 109.7222 212.6111 225.8889 181.9444 156.1111 162.0000 144.4444 150.6667

Column 28

236.7778

parameter\_mean(:, :, 24) =

Columns 1 through 9

197.0000 124.3889 184.0556 227.8889 183.5000 153.2222 194.2778 230.3889 149.8333

Columns 10 through 18

65.7778 69.2222 122.4444 188.3333 225.3333 216.5000 123.5000 101.3889 126.5556

Columns 19 through 27

112.5556 173.2222 234.4444 203.6667 156.8889 203.7222 244.5556 192.1667 147.8333

Column 28

212.3333

parameter\_mean(:, :, 25) =

Columns 1 through 9

171.3333 115.6667 141.2778 215.2222 251.7778 225.5000 220.2222 220.0000 218.2778

Columns 10 through 18

163.2778 125.5556 233.4444 213.8333 171.1667 218.1667 179.3889 121.8333 147.1667

Columns 19 through 27

223.7778 183.0556 140.6111 192.2222 129.6111 123.2222 135.2778 137.7778 145.5556

Column 28

216.4444

parameter\_mean(:,26) =

Columns 1 through 9

202.2222 140.6667 220.8333 236.9444 233.0000 201.5000 194.5000 104.9444 96.7778

Columns 10 through 18

189.0000 149.9444 133.2222 98.0556 123.1667 140.4444 110.3889 115.6111 123.8889

Columns 19 through 27

182.5000 115.2222 130.6111 240.2778 222.7222 219.3333 230.4444 216.5000 157.4444

Column 28

197.3333

parameter\_varian(:,1) =

Columns 1 through 9

16.4420 42.9117 26.6272 19.0826 23.5397 53.9089 46.4765 63.2689 39.6455

Columns 10 through 18

41.7500 12.9288 48.0930 39.6125 23.4560 73.8657 38.1805 58.9449 52.4918

Columns 19 through 27

31.7624 43.3944 19.5913 28.9519 53.1325 40.2143 39.6373 45.5324 28.5559

Column 28

64.8711

parameter\_varian(:,2) =

Columns 1 through 9

51.5791 41.7597 37.0545 46.9633 54.5506 51.5823 35.4597 40.9398 46.3489

Columns 10 through 18

52.9206 48.3105 55.5569 53.7287 45.4668 0.2357 16.2481 58.9548 52.5897

Columns 19 through 27

45.5079 61.2793 40.4627 0.6077 2.6003 38.8913 53.1325 34.3058 38.7723

Column 28

32.8257

parameter\_varian(:,3) =

Columns 1 through 9

11.7630 47.8493 44.6809 37.0799 36.4695 40.2870 38.1819 62.9325 30.4249

Columns 10 through 18

45.4131 20.6376 29.9259 58.1603 38.2241 64.8809 58.6497 24.7074 0.9852

Columns 19 through 27

22.5436 58.8356 47.7912 35.6507 40.7121 14.5840 5.1220 39.7566 49.2440

Column 28

56.6366

parameter\_varian(:,4) =

Columns 1 through 9

0.4278 0.6157 2.2000 45.6125 41.7352 30.4809 41.4954 14.7763 13.8956

Columns 10 through 18

37.9234 45.9197 50.9684 53.6604 38.9362 37.0227 57.0220 59.3624 39.1812

Columns 19 through 27

44.7032 45.6589 46.5368 47.9106 49.6314 50.9154 45.5917 42.9015 41.0803

Column 28

50.1140

parameter\_varian(:,5) =

Columns 1 through 9

13.6468 42.8842 42.0245 50.2997 40.4138 37.0139 23.2865 52.2736 51.2232

Columns 10 through 18

69.7732 50.5397 53.7106 47.7800 51.6552 52.8025 44.4737 54.9279 61.1625

Columns 19 through 27

28.9650 45.9222 43.2446 31.2262 60.9118 66.4994 31.5599 33.3764 51.4976

Column 28

56.5094

parameter\_varian(:,6) =

Columns 1 through 9

52.8797 74.0261 59.5043 60.5800 55.6428 46.5217 50.0167 59.0332 43.8097

Columns 10 through 18

51.2390 46.4976 36.8507 41.2365 42.8596 38.1067 37.8115 45.8011 55.4789

Columns 19 through 27

42.2308 55.9782 50.6752 48.0977 31.3016 19.3560 28.2186 2.0000 10.8256

Column 28

21.7904

parameter\_varian(:,7) =

Columns 1 through 9

13.8575 38.8364 36.3124 50.4068 14.7604 34.2963 29.9784 49.5462 39.3012

Columns 10 through 18

52.2165 46.5709 21.9683 44.8110 43.3688 49.1462 59.0491 48.7213 45.1457

Columns 19 through 27

36.4970 58.2926 44.7053 44.9788 52.2702 39.5957 40.4373 35.1259 43.8730

Column 28

31.4797

parameter\_varian(:,8) =

Columns 1 through 9

49.1229 43.0688 34.2337 34.2568 45.8598 51.2417 45.9581 35.3496 39.7436

Columns 10 through 18

41.3055 60.4164 57.8689 52.9489 31.2862 2.3403 0.2357 38.5785 48.3284

Columns 19 through 27

44.0185 28.1801 28.9279 0.6077 0.4851 17.1052 43.0694 29.3784 36.0042

Column 28

45.8752

parameter\_varian(:,9) =

Columns 1 through 9

32.4477 59.6047 28.4114 47.1220 49.0316 53.1726 35.5526 59.1481 73.6453

Columns 10 through 18

72.1390 59.6741 55.8825 88.2110 62.8022 65.8702 67.0419 70.4783 64.7752

Columns 19 through 27

64.5079 50.2038 71.1041 44.3712 34.9003 8.0604 13.4510 32.1543 58.3209

Column 28

36.2868

parameter\_varian(:,10) =

Columns 1 through 9

5.9158 22.7307 8.2194 1.7083 15.7181 34.5127 37.8169 44.9346 59.9041

Columns 10 through 18

37.4735 22.1363 8.9889 28.8420 28.1295 43.6281 55.4100 45.3191 31.9469

Columns 19 through 27

36.5499 47.6932 43.5560 29.0278 22.0354 29.6915 36.0166 42.2346 47.7865

Column 28

31.4739

parameter\_varian(:,11) =

Columns 1 through 9

62.5356 43.8548 37.0143 44.4936 49.2857 53.7349 58.0263 36.6089 31.2111

Columns 10 through 18

48.8763 52.8091 58.6932 67.3055 34.6533 1.6088 43.2342 45.4927 67.8311

Columns 19 through 27

52.4208 69.3772 23.4434 5.0241 31.0090 56.5470 41.5257 62.5954 52.5888

Column 28

37.2167

parameter\_varian(:,12) =

Columns 1 through 9

26.7982 57.4537 75.6423 46.8366 28.3341 12.4440 1.6088 66.0457 65.6102

Columns 10 through 18

45.7728 30.6081 48.7697 72.5834 47.1584 48.8388 78.5493 61.9496 59.0814

Columns 19 through 27

72.4015 51.4604 64.4708 7.5210 13.4433 2.2089 3.2879 14.0607 52.6746

Column 28

49.0941

parameter\_varian(:,13) =

Columns 1 through 9

41.6525 36.5173 36.0390 35.7703 32.3437 42.1112 39.7951 36.3703 23.4164

Columns 10 through 18

32.3561 34.2238 34.1986 28.4058 11.6278 31.6819 51.3410 39.8896 42.4545

Columns 19 through 27

40.4774 39.2317 14.5826 44.2536 50.4827 43.0511 33.6105 23.5509 39.0473

Column 28

42.4514

parameter\_varian(:,14) =

Columns 1 through 9

40.6547 50.8993 44.9529 46.2299 45.2391 48.2655 41.8726 67.6661 44.0225

Columns 10 through 18

65.3892 54.5113 51.6065 48.8273 26.5651 72.7500 45.5738 52.7828 32.0129

Columns 19 through 27

25.6406 21.4805 11.0105 38.7302 50.0511 42.7345 34.5849 32.4392 33.1742

Column 28

49.2053

parameter\_varian(:,15) =

Columns 1 through 9

10.6509 35.8629 29.9199 25.0282 29.0987 42.5393 28.9587 45.9411 27.2229

Columns 10 through 18

60.9695 32.7396 31.1089 58.4908 54.2546 58.5865 46.6085 49.7750 16.8322

Columns 19 through 27

37.6395 46.7536 40.5518 26.9624 55.3300 32.9527 27.5724 28.5698 40.4955

Column 28

11.6676

parameter\_varian(:,16) =

Columns 1 through 9

44.4281 54.5258 33.4955 35.5454 37.6067 38.8029 35.0667 29.2363 36.0678

Columns 10 through 18

28.1776 41.3006 48.4641 43.0106 43.9624 30.5364 38.2093 42.8771 40.4118

Columns 19 through 27

43.6544 26.9187 33.8227 36.0363 31.9687 41.5889 35.7806 3.2519 0.7071

Column 28

0.7519

parameter\_varian(:,17) =

Columns 1 through 9

26.8701 44.7409 31.7815 56.4004 31.4736 13.3011 4.2875 51.6142 40.4649

Columns 10 through 18

60.6654 47.9376 29.2685 54.6632 39.5597 68.8231 58.2272 44.8250 44.9652

Columns 19 through 27

54.5228 61.2507 59.7887 59.4794 73.6371 70.3498 62.2077 58.3934 65.9636

Column 28

52.9333

parameter\_varian(:,18) =

Columns 1 through 9

48.3506 54.3196 47.6772 53.8910 54.0623 55.3709 33.1075 62.7131 46.7420

Columns 10 through 18

68.1117 54.5770 53.8824 52.5292 43.1581 54.0914 59.8359 28.3227 7.9918

Columns 19 through 27

16.2300 23.9111 7.2538 56.0598 41.9471 22.0102 2.1489 2.9682 2.6396

Column 28

1.2005

parameter\_varian(:,19) =

Columns 1 through 9

27.0060 71.3186 73.3972 65.9262 29.0683 44.1555 54.1260 41.4161 47.2755

Columns 10 through 18

58.0562 46.0683 59.6805 58.1347 55.2609 59.5406 52.3393 18.1753 52.8697

Columns 19 through 27

59.5712 47.6136 57.0647 46.2476 70.1565 25.4315 41.4377 69.5231 63.3043

Column 28

27.2301

parameter\_varian(:,20) =

Columns 1 through 9

33.3986 50.5140 62.5790 56.9698 54.0566 17.2009 3.0098 46.2767 51.1321

Columns 10 through 18

43.8011 34.6714 33.4955 54.2901 51.6427 17.6710 42.7434 47.6011 65.3587

Columns 19 through 27

39.8136 55.0984 58.2694 0.9218 18.7276 39.5393 48.0777 26.1021 63.3304

Column 28

53.6777

parameter\_varian(:, :, 21) =

Columns 1 through 9

49.5424 41.8797 26.2950 30.3843 40.1099 54.1945 27.2812 35.1196 49.0057

Columns 10 through 18

29.2350 22.9524 63.8907 35.9004 70.7000 22.5020 41.8639 40.5634 40.5426

Columns 19 through 27

63.3694 31.7184 58.4822 51.1090 58.7876 47.6655 41.1982 47.8784 58.0194

Column 28

54.9058

parameter\_varian(:, :, 22) =

Columns 1 through 9

68.7423 55.5044 34.1351 29.2881 32.2312 28.1632 11.4214 20.5430 24.9617

Columns 10 through 18

34.2175 38.5316 41.4660 36.6289 47.4240 28.0229 48.3872 46.7070 35.3578

Columns 19 through 27

30.2892 51.3766 43.4917 56.2137 31.6955 50.3430 45.3482 28.8727 10.7649

Column 28

1.5049

parameter\_varian(:, :, 23) =

Columns 1 through 9

17.7893 44.0889 28.6263 29.6408 20.5764 39.1075 27.0139 2.0041 19.3174

Columns 10 through 18

36.3500 45.2589 40.2330 50.4662 36.2235 3.4471 13.2508 32.4169 37.4794

Columns 19 through 27

40.9806 33.1340 47.7286 39.4788 41.0358 20.1725 19.0510 33.7137 44.9614

Column 28

20.8173

parameter\_varian(:, :, 24) =

Columns 1 through 9

41.5267 66.3796 52.0458 30.8047 61.9157 49.6073 45.7188 36.8816 59.3021

Columns 10 through 18

38.9503 48.4083 50.1448 58.6425 34.0467 41.5342 56.8768 39.1620 46.2897

Columns 19 through 27

30.8797 53.9592 32.4893 52.0373 54.5699 60.5755 18.5543 62.6308 59.9797

Column 28

53.0017

parameter\_varian(:, :, 25) =

Columns 1 through 9

46.1850 48.9910 55.9378 38.9034 5.0007 37.2547 43.0575 38.9796 38.5346

Columns 10 through 18

50.5089 54.4785 38.0034 48.2606 47.5422 37.1946 43.9105 33.6474 50.0273

Columns 19 through 27

40.9125 45.4293 50.5571 42.6788 43.0487 48.4738 44.9180 48.1671 56.4282

Column 28

38.0282

parameter\_varian(:, :, 26) =

Columns 1 through 9

53.4190 60.2036 35.2191 42.3007 30.1896 58.8490 62.9475 56.7839 53.0589

Columns 10 through 18

58.6204 50.6400 44.9517 61.2550 65.9628 61.9936 43.0542 38.7270 54.5289

Columns 19 through 27

54.4289 63.1111 56.0600 24.7667 37.9315 44.2347 34.9911 61.0557 62.3776

Column 28

56.9406