**­­KONVERSI AKSARA JAWA KE HURUF LATIN PADA CITRA DIGITAL MENGGUNAKAN METODE *DIAGONAL BASED FEATURE EXTRACTION***

**SKRIPSI**

Digunakan Sebagai Syarat Maju Ujian Diploma IV

Politeknik Negeri Malang

**Oleh:**

**ERSA RAHMAWATI AUROLLAH NIM. 1341180024**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**JULI 2017**

**KONVERSI AKSARA JAWA KE HURUF LATIN PADA CITRA DIGITAL MENGGUNAKAN METODE *DIAGONAL BASED FEATURE EXTRACTION***

**SKRIPSI**

Digunakan Sebagai Syarat Maju Ujian Diploma IV

Politeknik Negeri Malang

**Oleh:**

**ERSA RAHMAWATI AUROLLAH NIM. 1341180024**



HALAMAN JUDUL

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**JULI 2017**

HALAMAN PENGESAHAN

KONVERSI AKSARA JAWA KE HURUF LATIN PADA CITRA DIGITAL MENGGUNAKAN METODE *DIAGONAL BASED FEATURE EXTRACTION*

Disusun oleh:

ERSA RAHMAWATI AUROLLAH NIM. 1341180024

Skripsi ini telah diuji pada tanggal 19 Juli 2017

Disetujui oleh:



**PERNYATAAN**

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

Malang, 13 Juli 2017

Ersa Rahmawati Aurollah

HALAMAN PERNYATAAN

ABSTRAK

**Rahmawati, Ersa**. “Konversi Aksara Jawa ke Huruf Latin pada Citra Digital Menggunakan Metode *Diagonal Based Feauture Extraction*”. **Pembimbing: (1) Dr. Eng. Cahya Rahmad, S.T.,M.Kom (2) Putra Prima Arhandi, S.T., M.Kom**

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

Huruf Jawa atau yang selanjutnya disebut aksara Jawa adalah salah satu bentuk penulisan budaya Jawa selain menggunakan tulisan latin yang sering kita jumpai di beberapa peninggalan yang ada di Daerah Istimewa Yogyakarta (DIY). Bentuk – bentuk dari aksara Jawa sangat kompleks, sehingga sedikit sulit untuk memahami dikarenakan bentuknya yang tidak *familiar*.

Perkembangan teknologi menuntut untuk dapat membangun suatu aplikasi yang dapat membantu wisatawan domestik ataupun mancanegara mengartikan aksara Jawa yang di dalamnya terdapat informasi sejarah dari beberapa peninggalan yang ada di DIY. Tujuan utama dari penelitian ini adalah mengenali tulisan tangan dimana aplikasi tersebut mampu mengkonversi tulisan tangan aksara Jawa ke huuf latin. Metode identifikasi yang digunakan pada penelitian ini adalah ekstraksi ciri dengan menggunakan metode *diagonal based feature extraction*. Metode tersebut mampu mengenali ciri-ciri setiap aksara Jawa, khususnya aksara pokok yang terdiri dari 20 jenis huruf.

Berdasar hasil penelitian diketahui bahwa dengan menggunakan metode ekstrasi ciri, posisi pengambilan citra dan bentuk tulisan sangat menentukan proses perhitungan ekstrasi ciri pada aksara Jawa. Setiap masing-masing huruf pada aksara pokok menggunakan data set 3 jenis huruf yang berbeda, sehingga total kesuluruhan adalah 60 data set yang digunakan. Tingkat keakuratan aplikasi mencapai 75% untuk dapat mengkonversi aksara Jawa ke huruf Latin dengan tepat. Dari 20 data testing, menghasilkan 15 data yang benar dan 5 data yangsalah.

**Kata kunci** :pengenalan tulisan tangan, konversi aksara Jawa ke huruf latin, *diagonal based feature extraction*.

*ABSTRACT*

***Rahmawati, Ersa****. “Conversion of Java Character to Alphabets on Digital Image Using Diagonal Based Feature Extraction”.* ***Advisor: (1) Dr. Eng. Cahya Rahmad, S.T.,M.Kom (2) Putra Prima Arhandi, S.T., M.Kom***

***Bachelor Thesis, Informatics Enggineering Study Programme, Departement of Information Technology, State Polytechnic of Malang, 2017.***

*The Java character is one form of writing of Javanese culture in addition to using the alphabet that we often encounter in some relics that are in the special region of Yogyakarta (DIY). Characteristics of Java's character is very complex, so it is a bit difficult to understand due to it's not familiar.*

*Technological developments are demanding to be able to build an application that can help the domestic or international travelers to identify a Java character in which there is historical information from several existing relics at DIY. The main objective of this research is to handwriting recognition in which the application can convert Java's character to alphabets. Identification methods in this research use of digital image processing, namely the extraction of characteristics by method diagonal based feature extraction. The method can recognize the characteristics of each Java character, Especially the basic character in the Java character which consists of 20 types of letters.*

*Based on the results of the study known that by using the method of extracting the characteristics of image taking positions and the form of writing is to determine the process of calculating the extraction of these characteristics. Each character in the main, character uses 3 different sets of character, so the total is 60 data sets used. The application's accuracy rate reaches 75% in order to properly convert Java character to the alphabet. From 20 data testing, get 15 correct data and 5 wrong data.*

***Keywords****: handwriting recognition, conversion of Java character to alphabets, diagonal based feature extraction.*

KATA PENGANTAR

Puji Syukur kami panjatkan kehadirat Allah SWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul “Konversi Aksara Jawa ke Huruf Latin pada Citra Digital Menggunakan Metode *Diagonal Based Feature Extraction*” ini dengan tepat waktu dan sebaik-baiknya.

Laporan skripisi ini disusun sebagai persyaratan untuk menyelesaikan studi Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Kami menyadari tanpa adanya dukungan dan kerja sama dari berbagai pihak laporan skripsi ini tidak akan dapat berjalan baik. Untuk itu, kami ingin menyampaikan rasa terima kasih kepada:

1. Allah SWT yang selalu memberi rahmat dan hidayah-Nya sehingga diberi kemampuan untuk menyelesaikan skripsi ini.
2. Kedua orang tua (Jauhari dan Nur Ida Astutik), kakak (Anggari Prasasti), dan adik-adik (Innayyah Heina dan Miqdad Fairuz) serta keluarga besar tercinta yang dengan tulus memberikan dukungan moral maupun materi, sehingga penyusunan laporan skripsi ini dapat berjalan dengan lancar.
3. Bapak Ir. Tundung Subali Patma., M.T, selaku Direktur Politeknik Negeri Malang.
4. Bapak Rudy Ariyanto, ST., M.Cs., selaku ketua Jurusan Teknologi Informasi
5. BapakIr. Deddy Kusbianto P., M.MKom., selaku ketua Program Studi Teknik Informatika.
6. Bapak Dr. Eng. Cahya Rahmad, S.T.,M.Kom dan Bapak Putra Prima Arhandi, S.T., M.Kom, selaku Dosen Pembimbing atas segala bimbingan dan dukungannya.
7. Dea Pradana Anindisa Widiyantoro yang selalu setia menemani, memotivasi dan memberi semangat serta dukungan dengan tulus.
8. Naufal Ziyad, Robertus Romario, Aldio Putra, Vivid Ichtarosa, Eka Zulfa yang salalu setia memberi semangat dan menemani perjalanan selama di bangku perkuliahan.
9. Mas Irtafa’ Masrufi yang selalu memberikan dukungan dan semangat saat mengerjakan skripsi.
10. Anggieta Januariska, Rofidah Iman Sari sahabat sejak sekolah dasar yang terpisah jarak namun masih tetap ada untuk memberiku semangat dalam menyelesaikan kuliah hingga skripsi.
11. SMPN 1 Lumajang yang telah membantu untuk melakukan uji validasi skripsi.
12. Keluarga besar UKM Lembaga Pers Mahasiswa Kompen yang telah memberikan wadah untuk terus berkarya.
13. Keluarga besar Himpunan Mahasiswa Teknologi Informasi yang telah memberikan kesempatan dan pengalaman berharga dalam sebuah perjuangan.
14. Semua sahabat dan teman di Jurusan Teknologi Informasi Politeknik Negeri Malang angakatan 2013 yang telah berjuang bersama.
15. Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan laporan skripsi dari awal hingga akhir yang tidak dapat kami sebutkan satu persatu.

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

Malang, 13 Juli 2017

Penulis

DAFTAR ISI

[HALAMAN JUDUL i](#_Toc491184428)

[HALAMAN PENGESAHAN ii](#_Toc491184429)

[HALAMAN PERNYATAAN iii](#_Toc491184430)

[ABSTRAK iv](#_Toc491184431)

[*ABSTRACT* v](#_Toc491184432)

[KATA PENGANTAR vi](#_Toc491184433)

[DAFTAR ISI viii](#_Toc491184434)

[DAFTAR GAMBAR ix](#_Toc491184435)

[DAFTAR TABEL xi](#_Toc491184436)

[DAFTAR LAMPIRAN xii](#_Toc491184437)

[BAB I. PENDAHULUAN 1](#_Toc491184438)

[1.1 Latar Belakang 1](#_Toc491184439)

[1.2 Rumusan Masalah 2](#_Toc491184440)

[1.3 Tujuan 2](#_Toc491184441)

[1.4 Batasan Masalah 2](#_Toc491184442)

[1.5 Sistematika Penulisan 3](#_Toc491184443)

[BAB II. LANDASAN TEORI 5](#_Toc491184444)

[2.1 Citra Digital 5](#_Toc491184445)

[2.2 Aksara Jawa 6](#_Toc491184446)

[2.3 Konsep *HandwritingRecognition* 7](#_Toc491184447)

[2.4 *Diagonal Based Feature Extraction* 10](#_Toc491184448)

[2.5 Uji Kemiripan 12](#_Toc491184449)

[BAB III. METODOLOGI PENELITIAN 14](#_Toc491184450)

[3.1 Metode Penelitian 14](#_Toc491184451)

[BAB IV. ANALISIS DAN PERANCANGAN 16](#_Toc491184452)

[4.1 Analisis Sistem 16](#_Toc491184453)

[4.2 *Preprocessing* 17](#_Toc491184454)

[4.3 Ekstrasi Ciri 19](#_Toc491184455)

[4.4 Uji Kemiripan 22](#_Toc491184456)

[4.5 Analisa Kebutuhan Sistem 22](#_Toc491184457)

[4.6 Perancangan 23](#_Toc491184458)

[BAB V. IMPLEMENTASI 30](#_Toc491184459)

[5.1 Implementasi *Database* 30](#_Toc491184460)

[5.2 Implementasi Antar Muka 31](#_Toc491184461)

[BAB VI. PENGUJIAN DAN PEMBAHASAN 41](#_Toc491184462)

[6.1 Pengujian 41](#_Toc491184463)

[6.2 Pembahasan 50](#_Toc491184464)

[BAB VII. KESIMPULAN 51](#_Toc491184465)

[7.1 Kesimpulan 51](#_Toc491184466)

[7.2 Saran 51](#_Toc491184467)

[DAFTAR PUSTAKA 52](#_Toc491184468)

[LAMPIRAN – LAMPIRAN 55](#_Toc491184469)

DAFTAR GAMBAR

[Gambar 2.1 Koordinat Citra Digital 5](#_Toc491184783)

[Gambar 2.2 Bentuk Matriks Citra Digital 5](#_Toc491184784)

[Gambar 2.3 Aksara Pokok 7](#_Toc491184785)

[Gambar 2.4 Diagram Skema *RecognitionSystem* 8](#_Toc491184786)

[Gambar 2.5 Diagram Proses *Preprocessing* 8](#_Toc491184787)

[Gambar 2.6 Ilustrasi *8-neighbour*. 10](#_Toc491184788)

[Gambar 2.7 Citra Sebelum Proses *Thinning*. 10](#_Toc491184789)

[Gambar 2.8 Citra Setelah Proses *Thinning*. 10](#_Toc491184790)

[Gambar 2.9 Cara untuk Mengekstraksi Fitur dari Karakter 11](#_Toc491184791)

[Gambar 2.10 Histogram Diagonal Zona 11](#_Toc491184792)

[Gambar 3.1 Tahapan Penelitian 14](#_Toc491184793)

[Gambar 4.1 Gambaran Kerja Sistem 17](#_Toc491184794)

[Gambar 4.2 Skema *Preprocessing* 17](#_Toc491184795)

[Gambar 4.3 Tahap Ekstrasi Ciri 19](#_Toc491184796)

[Gambar 4.4 Citra Ukuran 30x30 piksel 19](#_Toc491184797)

[Gambar 4.5 Pembagian Zona (10x10 piksel) 19](#_Toc491184798)

[Gambar 4.6 Contoh Menghitung Nilai Zona 20](#_Toc491184799)

[Gambar 4.7 Pembagian 9 zona. 20](#_Toc491184800)

[Gambar 4.8 *Flowchart Preprocessing* 24](#_Toc491184801)

[Gambar 4.9 *Flowchart* Ekstrasi Ciri 25](#_Toc491184802)

[Gambar 4.10 *Flowchart* Membagi Zona 26](#_Toc491184803)

[Gambar 4.11 *Flowchart* Menghitung Zona 26](#_Toc491184804)

[Gambar 4.12 *Flowchart* Menghitung Rata – Rata Zona 27](#_Toc491184805)

[Gambar 4.13 *Flowchart* Menghitung Rata – Rata Baris dan Kolom 27](#_Toc491184806)

[Gambar 4.14 *Flowchart* Uji Kemiripan 28](#_Toc491184807)

[Gambar 4.15 Rancangan Halaman Utama 28](#_Toc491184808)

[Gambar 4.16 Rancangan Halaman *Trainning* 29](#_Toc491184809)

[Gambar 4.17 Rancangan Halaman *Testing* 29](#_Toc491184810)

[Gambar 5.1 Implementasi Tabel Citra 30](#_Toc491184811)

[Gambar 5.2 Implementasi Tabel Rata – Rata 30](#_Toc491184812)

[Gambar 5.3 Implementasi Tabel Hasil 30](#_Toc491184813)

[Gambar 5.4 Tampilan Halaman Utama 31](#_Toc491184814)

[Gambar 5.5 Tampilan Halaman Data *Trainning* 32](#_Toc491184815)

[Gambar 5.6 Tampilan *File Dialog untuk Input* Citra Aksar Jawa 32](#_Toc491184816)

[Gambar 5.7 Tampilan *Input* Citra Aksara Jawa 32](#_Toc491184817)

[Gambar 5.8 Tampilan Proses *Preprocesing*. 33](#_Toc491184818)

[Gambar 5.9 Tampilan Proses Perhitungan Ekstrasi Ciri. 34](#_Toc491184819)

[Gambar 5.10 Tampilan Proses Menyimpan *Dataset*. 34](#_Toc491184820)

[Gambar 5.11 Tampilan Proses Berhasil Menyimpan *Dataset*. 34](#_Toc491184821)

[Gambar 5.12 Tampilan Halaman Data *Testing* 35](#_Toc491184822)

[Gambar 5.13 Tampilan *File Dialog* untuk *Input* Citra Aksara Jawa. 36](#_Toc491184823)

[Gambar 5.14 Tampilan *Preprocessing* Tanpa Proses *Inverse*. 36](#_Toc491184824)

[Gambar 5.14 Tampilan *Preprocessing* dengan Proses *Inverse*. 37](#_Toc491184825)

[Gambar 5.15 Tampilan Proses Perhitungan Ekstrasi Ciri. 38](#_Toc491184826)

[Gambar 5.16 Tampilan Proses Uji Kemiripan 38](#_Toc491184827)

[Gambar 5.16 Tampilan Pesan *Error* jika Belum *Input* Citra. 38](#_Toc491184828)

[Gambar 5.17 Tampilan Pesan *Error* jika Tidak Ada Nilai pada Label 39](#_Toc491184829)

[Gambar 5.18 *Button* Halaman Bantuan 39](#_Toc491184830)

[Gambar 5.19 Tampilan Halaman Bantuan 40](#_Toc491184831)

[Gambar 5.20 Tampilan Halaman Tentang Aplikasi 40](#_Toc491184832)

DAFTAR TABEL

[Tabel 4.1 Tabel Citra 23](#_Toc491185308)

[Tabel 4.2 Tabel Rata-Rata 23](#_Toc491185309)

[Tabel 4.3 Tabel Hasil 24](#_Toc491185310)

[Tabel 6.1 Pengujian Unit *Training* 42](#_Toc491185311)

[Tabel 6.2 Pengujian Unit *Testing* 44](#_Toc491185312)

[Tabel 6.3 Pengujian Konversi Aksara Jawa ke Huruf Latin 47](#_Toc491185313)

[Tabel 6.4 Hasil Pengujian Tingkat Keakurasian 49](#_Toc491185314)

DAFTAR LAMPIRAN

Lampiran 1. Surat Pengantar Observasi

Lampiran 2. Surat Keterangan Uji Coba Sistem

Lampiran 3. Lembar Bimbingan Skripsi (Pembimbing 1)

Lampiran 4. Lembar Bimbingan Skripsi (Pembimbing 2)

Lampiran 5. Lembar Persetujuan Mengikuti Ujian Skripsi

Lampiran 6. Form Revisi Skripsi (Penguji 1)

Lampiran 7. Form Revisi Skripsi (Penguji 2)

Lampiran 8. Form Verifikasi Abstrak Bahasa Inggris dan Tata Tulis Buku Skripsi

Lampiran 9. Kode Program Browse

Lampiran 10. Kode Program Reset

Lampiran 11. Kode Program Grayscale

Lampiran 12. Kode Program Binnarization

Lampiran 13. Kode Program Inverse

Lampiran 14. Kode Program Thinning

Lampiran 15. Kode Program Fit

Lampiran 16. Kode Program Resize

Lampiran 17. Kode Program Zona

Lampiran 18. Kode Program Rata – Rata

Lampiran 19. Kode Program Simpan

Lampiran 20. Kode Program Konversi Aksara jawa ke Huruf Latin

BAB I. PENDAHULUAN

## Latar Belakang

Indonesia merupakan negara yang kaya akan budaya, dan bahasa. Kekayaan inilah yang harus kita lestarikan dan kita jaga seiring banyaknya budaya dan bahasa barat dan atau bahasa asing yang masuk ke Indonesia. Daerah Istimewa Yogyakarta (DIY) adalah salah satu provinsi di Indonesia yang menjadi pusat pariwisata untuk domestik maupun mancanegara. Hal ini dikarenakan DIY memiliki banyak lokasi wisata yang mudah ditempuh dan terjangkau. Selain itu budaya Jawa yang masih kental, kreativitas, seni dan sikap ramah masyarakat DIY.

Huruf Jawa atau yang selanjutnya disebut aksara Jawa adalah salah satu bentuk penulisan budaya Jawa selain menggunakan tulisan latin yang sering kita jumpai di beberapa peninggalan yang ada di DIY. Bentuk – bentuk dari aksara Jawa sangat kompleks, sehingga sedikit sulit untuk memahami dikarenakan bentuknya yang tidak *familiar*. Hal ini membuat wisatawan domestik ataupun mancanegara membutuhkan seorang *guide* untuk mengartikan aksara Jawa yang di dalamnya terdapat informasi sejarah dari beberapa peninggalan yang ada di DIY. Alat bantu menerjemahkan aksara Jawa ke bahasa latin yang sudah ada hanya berupa *software* yang masukannya dengan cara di ketik manual. Padahal tulisan aksara Jawa sering dijumpai pada tempat – tempat peninggalan seperti monumen, tugu, dll. Oleh karena itu, dibutuhkan sebuah sistem yang mampu mengenali sebuah tulisan tangan yang terdapat pada beberapa tempat wisata peninggalan di DIY.

*Handwriting recognition* atau pengenalan sebuah tulisan tangan pada citra diperlukan metode yang mampu mengekstraksi ciri pada citra digital khususnya citra aksara Jawa. Diantara beberapa ekstrasi fitur terdapat metode *diagonal based feature extraction* yang sering digunakan untuk mengenali sebuah tulisan tangan. Metode ekstraksi ciri *diagonal based feature extraction* mampu mengambil dan menyimpan karakteristik khusus dari suatu objek pada citra. Fitur -fitur inilah yang kemudian akan digunakan sebagai pembanding untuk mengenali objek tertentu pada suatu citra [1].

Dari uraian permasalahan di atas, maka pada penelitian ini dikembangkan sebuah sistem untuk mengkonversi aksara Jawa ke huruf latin pada citra digital. Sistem ini mampu mendeteksi tulisan tangan berupa tulisan aksara Jawa pada sebuah citra digital serta mampu mengkonversi ke huruf latin. Diharapkan sistem ini mampu menjadi media yang dapat mempermudah wisatawan DIY untuk membaca serta mengetahui informasi sejarah yang ada di beberapa tempat wisata.

## Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, rumusan masalah dalam penelitian ini adalah :

1. Bagaimana cara menentukan ciri citra aksara Jawa dengan menggunakan *diagonal based feature extraction*?
2. Bagaimana menghitung kemiripan citra berdasarkan ekstraksi citra menggunakan perhitungan *ecludian distance*?

## Tujuan

Tujuan dari penelitian ini adalah untuk pengembangan sistem konversiaksara Jawa ke huruf latin menggunakan metode *diagonal based feature extraction*.

Hasil penelitian ini diharapkan dapat bermanfaat bagi beberapa pihak, diantaranya:

1. Untuk Wisatawan, agar dapat membantu mengetahui informasi sejarah yang berada di beberapa tempat wisata DIY.
2. Untuk Mahasiswa, agar dapat mengembangkan kemampuan dalam mengembangkan perangkat lunak.
3. Untuk Bidang Ilmu Pengetahuan, agar menambah referensi penggunaan suatu metode pengembangan sistem *handwriting recognition* pada citra digital.

## Batasan Masalah

Batasan-batasan masalah pada penelitian ini adalah :

1. Citra yang digunakan adalah citra digital dengan text aksara Jawa.
2. Text aksara Jawa pada citra diambildalam posisi sejajar pada ruang terbuka.
3. Warna citra aksara Jawa dengan background harus kontras.
4. Format file masukan dalam format .jpg dari gambar yang bertuliskan aksara Jawa.
5. Text aksara Jawa yang dapat dideteksi adalah aksara pokok.
6. Konversi dilakukan setiap huruf dan *cropping* dilakukan secara manual.
7. Implemenatasi menggunakan bahasa pemrograman *Visual Basic* (VB).

## Sistematika Penulisan

Sistematika penulisan laporan skripsi ini terdiri dari 7 bab yang disusun sebagai berikut :

**BAB I PENDAHULUAN**

Pada bab ini dijelaskan mengenai hal-hal yang bersifat umum seperti latar belakang, rumusan masalah, batasan masalah, tujuan penulisan, dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Pada bab ini berisi teori-teori yang mendasari dan berkaitan dengan masalah perencanaan dan pembuatan aplikasi yang digunakan untuk memudahkan pemahaman dan pemecahan terhadap masalah yang ada.

**BAB III METODOLOGI PENELITIAN**

Pada bab ini menjelaskan langkah-langkah yang digunakan penulis untuk memilih metode, teknik, prosedur apa yang tepat, *tools* apa yang akan digunakan sehingga setiap tahap penelitian dapat dilakukan dengan tepat.

**BAB IV ANALISIS DAN PERANCANGAN**

Berisi pembahasanmengenai analisis dan perancangan terhadap sistem.

**BAB V IMPLEMENTASI**

Pada bab ini membahas mengenai pembuatan aplikasi yang telah didesain pada tahap sebelumnya ke dalam bahasa pemrograman, disertai dengan gambaran databse dan interface aplikasi.

**BAB VI PENGUJIAN DAN PEMBAHASAN**

Pada bab ini membahas mengenaiapakah hasil dari tugas akhir sudah sesuai dengan kebutuhan sistem dan berjalan sesuai lingkungan yang diinginkan. Pengujian dapat berupa pengujian fungsional. Pada pengujian dipaparkan secara detail mengenai metode pengujian, tujuan pengujian, proses pengujian serta analisa hasil pengujian.

**BAB VII KESIMPULAN**

Pada bab ini membahas mengenai kesimpulan yang diperoleh dengan melihat hasil pengujian aplikasi yang telah dibuat dan saran yang dapat digunakan untuk kepentingan pengembangan selanjutnya.

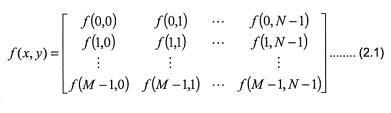
BAB II. LANDASAN TEORI

## Citra Digital

Citra digital merujuk pada pemrosesan gambar 2 dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiapdata dua dimensi. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai *real* maupun komplek yang direpresentasikan dengan deretan bit tertentu [2].

Suatu citra didefinisikan sebagai fungsi *f(x,y)* berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo *f* dan titik koordinat *(x,y)*dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x,y, dan nilai amplitudo *f* secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dikatakan bahwa citra tersebut adalah citra digital. Gambar 2.1 menunjukkan posisi koordinat citra digital.

# Gambar 2.1 Koordinat Citra Digital

Citra digital dapat ditulis dalam bentuk matrik seperti pada Gambar 2.2.

# Gambar 2.2 Bentuk Matriks Citra Digital

Nilai pada suatu irisan baris dan kolom (pada posisi x,y) disebut dengan *picture elements*, *image elements*, *pels*, atau *pixels*. Istilah terakhir (*pixel*) paling sering digunakan pada citra digital.

Operasi – operasi yang terdapat pada pengolahan citra antara lain [3]:

* + 1. *Cropping*

*Cropping* adalah proses pengambilan bagian dalam suatu citra pada bagian yang diinginkan saja. Pengambilan bagian pada suatu citra dilakukan agar citra yang akan diproses lebih fokus pada bagian yang diolah selanjutnya. Contohnya pada penelitian ini adalah ketika kita ingin melakukan pengenalan pada tulisan tangan yang mengandung aksara Jawa pada sebuah citra suatu objek peninggalan, citra yang menampilkan suatu objek dapat di *crop* sehingga menyisakana bagian tulisan aksara Jawa saja.

* + 1. *Resize*

Mengubah besarnya ukuran citra digital dalam piksel.

* + 1. *Binarization*

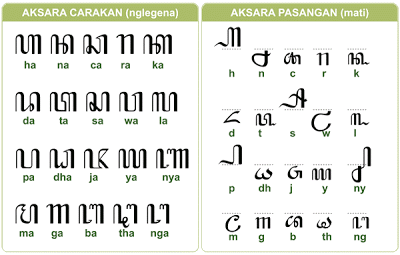
Citra biner disebut juga citra hitam dan putih. Hal ini dikarenakan hanya dibutuhkan 1 bit untuk mewakili setiap nilai citra biner.

## Aksara Jawa

Bahasa Jawa adalah salah satu bahasa daerah yang merupakan bagian dari kebudayaan nasional Indonesia, yang hidup dan tetap dipergunakan dalam masyarakat bersangkutan. *Carakan* (abjad Jawa) yang digunakan di dalam ejaan bahasa Jawa pada dasarnya terdiri dari 20 aksara pokok yang bersifat *silabik* (bersifast kesukukataan). Masing – masing aksara pokok mempunyai aksara pasangan, yakni akasara yang berfungsi untuk menghubungkan suku kata tertutup konsonan dengan suku kata berikutnya. Bahasa Jawa ternyata senantiasa terus – menerus mengalami perkembangan, sehingga ejaannya pun perlu disesuaikan dengan perkembangan tersebut, terutama dalam penulisan aksara Jawa yang makin tidak dikenal oleh masyarakat. Oleh karena itu, Dinas Pendidikan dan Kebudayaan Provinsi Daerah Istimewa Yogyakarta (DIY) pada tahun anggaran 1992/1993 memutuskan ditetapkan adanya pedoman penulisan aksara Jawa. Hal itu juga telah disepakati oleh Pemerintah Provinsi Daerah Tingkat I Jawa Tengah dan Pemerintah Provinsi Daerah Tingkat I Jawa Timur [4].

Adapaun ringkasan dari pedoman penulisan aksara Jawa, sebagai berikut :

* + 1. Aksara Pokok

Aksara Pokok atau biasa yang disebut dengan aksara carakan yaitu,aksara inti yang terdiri dari 20 aksara.

# Gambar 2.3 Aksara Pokok

## Konsep *HandwritingRecognition*

*Handwriting recognition* atau pengenalan tulisan tangan adalah suatu proses dimana komputer menerjemahkan tulisan tangan kedalam teks komputer [1]. Proses pengenalan tulisan tangan menghadapi tantangan tertentu [5] - [8]. Tantangan terbesar adalah gambar tulisan tangan memiliki berbagai dimensi yang harus dinormalisasi dan / atau diproses agar sesuai dengan persyaratan spesifik sistem. Dengan demikian tantangan pengenalan tulisan tangan tidak hanya dari cara menulisnya saja, namun juga dari beberapa kriteria lain yang harus memenuhi persyaratan dalam penulisan tangan itu sendiri [9], [10] . Selain itu, ada berbagai tingkat ketebalan tulisan tangan dan posisi penulisan yang berbeda [11] - [13]. Apalagi Akhtar dkk, [9] menunjukkan bahwa karena berbagai gaya penulisan orang akan bergantung pada usia, kualifikasi, mode, latar belakang, dan lain-lain. Pengenalan tulisan tangan adalah penelitian yang relatif rumit, jenis tulisan tangan akan memiliki dampak besar pada klasifikasi dan / atau identifikasi, karena subjek yang berbeda akan menggunakan gaya penulisan yang berbeda. Bahkan, tulisan yang telah ditulis oleh orang yang sama pada waktu yang berbeda juga ditemukan bervariasi [14], [15]. Oleh karena itu, hampir tidak mungkin untuk mengembangkan pengenal generik yang dapat mengenali jumlah tak terbatas sebuah tulisan tangan.

Proses pengenalan tulisan terdiri dari beberapa tahap, yaitu ; *image acquisition, pre-processing, feature extraction* dan *distance measurement* ditunjukkan pada gambar 2.4.

# Gambar 2.4 Diagram Skema *Recognition System*

Preprocessing adalah rangkaian operasi yang dilakukan pada citra input. Pada dasarnya mempersiapkan citra yang sesuai untuk ekstraksi fitur. Beberapa opersai yang dilakukan pada gambar pada tahap *preprocessing* ditunjukkan pada gambar 2.5 [16].

# Gambar 2.5 Diagram Proses *Preprocessing*

* + 1. *Scanned Input Image*

Merupakan proses penangkapan citra. Suatu citra harus memiliki format khusus seperti JPEG, BMP, dll. Citra digital didapatkan melalui digital kamera atau perangkat lain seperti *handphone*, dll.

* + 1. *Grayscale Image*

Citra tulisan tangan di normalisasi untuk pengenalan tulisan tangan dengan mengubah cirtra berwarna menjadi citra ke-abuan, ditunjukkan dengan persamaan sebagai berikut [17] :

(2.1)

* + 1. *Binarization*

Konversi dari citra grayscale menjadi citra biner yang memiliki nilai0/1. 0 adalah nilai untuk hitam dan 1 adalah nilai untuk putih denganmenggunakan teknik *thresholding*.

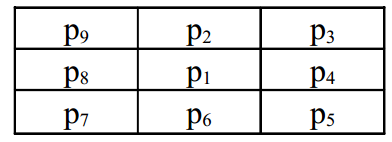
* + 1. *Thinning*

Algoritma ini adalah salah satualgoritma thinning yang cukup populardan telah digunakan sebagai suatu basisperbandingan untuk thinning. Setiap iterasi dari metode ini terdiri dari dua sub-iterasi yang berurutan yang dilakukan terhadap *contour points* dariwilayah citra.*Contour point* adalahsetiap pixel dengan nilai 1 dan memilikisetidaknya satu *8-neighbor* yangmemiliki nilai 0.

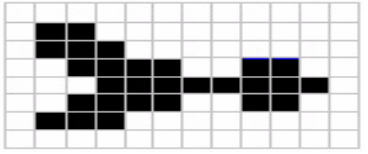
Dengan informasi ini, langkah pertamaadalah menandai contour point p untukdihapus jika semua kondisi ini dipenuhi [18]:

1. (2.2)
2. (2.3)
3. (2.4)
4. (2.5)

dimana N(p1) adalah jumlah tetanggadari p1 yang tidak 0; yaitu,  
N(p1) = p2 + p3 + ... + p8 + p9



# Gambar 2.6 Ilustrasi *8-neighbour*.

Jika salah satu dari keempat kondisi diatas tidak dipenuhi atau dilanggar makanilai piksel yang bersangkutan tidakdiubah. Sebaliknya jika semua kondisitersebut dipenuhi maka piksel tersebutditandai untuk penghapusan.

# Gambar 2.7 Citra Sebelum Proses *Thinning*.

# Gambar 2.8 Citra Setelah Proses *Thinning*.

* + 1. *Resize*

Mengubah ukuran citra digital dalam piksel. Contoh 90x60 piksel. Hal ini digunakan mempermudah dalam proses ekstarksi fitur.

## *Diagonal Based Feature Extraction*

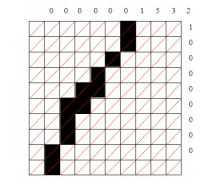
*Diagonal based feature extraction* adalah proses untuk mengenali karakter tulisan dengan metode *offline* di dalam pengerjaannya , algoritma ini mengekstraksi ciri yang membagi ukuran piksel gambar yang menjadi piksel – piksel yang lebih kecil dan sama rata. Misalnya *character image* yang berukuran 90x60 piksel dibagi menjadi 54 zona yang sama rata, setiap zona berukuran 10x10 piksel (Gambar 2.9)

# Gambar 2.9 Cara untuk Mengekstraksi Fitur dari Karakter

Fitur yang akan di ekstraksi dari setiap zona piksel dengan bergerak secara diagonal dari tiap zona yang masing – masing berukuran 10x10 piksel. Setiap zona memiliki 19 garis diagonal dan *foreground*piksel yang ada disetiap baris diagonal dan dijumlahkan untuk mendapatkan sub-fitur tunggal.

Kemudian, nilai dari 19 sub-fitur ini dibagi rata untuk mendapatkan satu nilai fitur dan ditempatkan di zona yang sesuai. Prosedur ini dilakukan berulang secara berurutan di semua zona. Apabila ada suatu zona yang garis diagonal nya bernilai kosong, maka nilai fitur yang sesuai dengan zona tersebut adalah 0.

Algoritma *diagonal based feature extraction* sebagai berikut [19]:

1. Hitung histogram diagonal setiap zona. Histogram diagonal adalah banyaknya piksel hitam setiap diagonal pada satu zona. Setiap zona memiliki 19 nilai histogram diagonal yang disebut Histd, dimana 1 ≤ d ≤ 19. Secara jelas dapat dilihat pada gambar 2.10.

# Gambar 2.10 Histogram Diagonal Zona

1. Hitung nilai fitur setiap zona, yaitu rata-rata histogram setiap zona, disebut Zn dimana 1 ≤ n ≤ 54.

(2.6)

1. Hitung rata-rata zona setiap baris, disebut Bi, dimana 1 ≤ i ≤ 9.

(2.7)

Baris 1 = (Z1+Z2+Z3+Z4+Z5+Z6) /6

Baris 2 = (Z7+Z8+Z9+Z10+Z11+Z12) /6

Baris 3 = (Z13+Z14+Z15+Z16+Z17+Z18) /6

Baris 4 = (Z19+Z20+Z21+Z22+Z23+Z24) /6

Baris 5 = (Z25+Z26+Z27+Z28+Z29+Z30) /6

Baris 6 = (Z31+Z32+Z33+Z34+Z35+Z36) /6

Baris 7 = (Z37+Z38+Z39+Z40+Z41+Z42) /6

Baris 8 = (Z43+Z44+Z45+Z46+Z47+Z48) /6

Baris 9 = (Z49+Z50+Z51+Z52+Z53+Z56) /6

1. Hitung rata-rata zona setiap kolom, disebut Kj, dimana 1 ≤ j ≤ 6

(2.8)

Kolom 1 = (Z1+Z7+Z13+Z19+Z25+Z31+Z37+Z43+Z49) / 9

Kolom 2 = (Z2+Z8+Z14+Z20+Z26+Z32+Z38+Z44+Z50) / 9

Kolom 3 = (Z3+Z9+Z15+Z21+Z27+Z33+Z39+Z45+Z51) / 9

Kolom 4 = (Z4+Z10+Z16+Z22+Z28+Z34+Z40+Z46+Z52) / 9

Kolom 5 = (Z5+Z11+Z17+Z23+Z29+Z35+Z41+Z47+Z53) / 9

Kolom 6 = (Z6+Z12+Z18+Z24+Z30+Z36+Z42+Z48+Z54) / 9

## Uji Kemiripan

Uji Kemiripan digunakan untuk menghitung jarak terdekat data training dan data testing. Pada penielitian ini menggunakan perhitungan Ecludian Distance. Jarak Euclidean didefinisikan sebagai jarak garis lurus antara dua titik, yang menguji akar perbedaan persegi antara koordinat sepasang benda [20], [21]. Jarak Euclidean dapat dihitung dengan menggunakan persamaan di bawah ini:

(2.9)

BAB III. METODOLOGI PENELITIAN

## Metode Penelitian

Untuk membantu dalam penyusunan penelitian ini, maka perlu adanya susunan kerangka kerja (*frame work*) yang jelas tahapan-tahapannya. Kerangka kerja ini merupakan langkah-langkah yang akan dilakukan dalam penyelesaian masalah yang akan dibahas. Adapun kerangka kerja penelitian yang di gunakan seperti terlihat pada Gambar 3.1 :

# Gambar 3.1 Tahapan Penelitian

Berdasarkan kerangka kerja penelitian yang telah digambarkan di atas, maka dapat diuraikan pembahasan masing-masing tahap dalam penelitian adalah sebagai berikut:

* + 1. Studi Literatur

Pada tahap ini dilakukan pencarian landasan-landasan teori yang diperoleh dari berbagai buku dan juga internet untuk melengkapi perbendaharaan konsep dan teori, sehingga memiliki landasan dan keilmuan yang baik dan sesuai.

* + 1. Pengumpulan Data

Pada tahap ini dilakukan proses pengumpulan data dengan metode observasi. Pada tahap ini dilakukan observasi terhadap citra-citra yang sesuai digunakan sebagai objek penelitian dalam pengembangan aplikasi konversi aksaa Jawa ke huruf latin pada citra digital.

* + 1. Analisis Sistem

Pada tahap ini dilakukan identifikasi masalah dan membuat desain sistem. Diantaranya alur sistem, kebutuhan *software* dan *hardware*, desain *database* dan desain antarmuka.

* + 1. Pengembangan Sistem

Pada Tahap ini dilakukan pengembangan sistem dengan menggunakan model *waterfall.* Sistem ini akan dikembangkan menjadi aplikasi berbasis *desktop* menggunakan *Visual Basic* (VB).

* + 1. Pembuatan Laporan

Pada tahapan ini dilakukan pembuatan laporan yang disusun berdasarkan hasil penelitian dengan menggunakan teknik pengumpulan data *primer* dan sekunder sehingga menjadi laporan penelitian yang dapat memberikan gambaran secara utuh tentang sistem yang sedang dibangun.

BAB IV. ANALISIS DAN PERANCANGAN

Pada bab ini akan dijelaskan analisis dan perancangankonversi aksara Jawa ke huruf latin pada citra digital.

## Analisis Sistem

Sistem yang akan dibuat merupakan salah satu percabangan dari topik *handwriting recognition* atau pengenalan tulisan tangan, yaitu dimana komputer mampu menerjemahkan tulisan tangan menjadi teks dalam komputer. Sistem pengenalan tulisan tangan pada umumnya adalah kemampuan komputer mengenali pengenalan tulisan tangan pada huruf latin, maka pada sistem konversi ini komputer mampu mengenali tulisan tangan pada aksara Jawa atau biasa dikenal dengan huruf Jawa.

Dalam konversi aksara Jawa ke huruf latin ini memiliki dua proses utama yaitu:

1. Tahap pertama adalah *preprocessing* merupakan tahap persiapan data, dimana citra yang telah diambil disiapkan untuk memudahkan proses selanjutnya, yaitu ekstrasi ciri. Adapun tahap *preprocessing* pada penelitian kali ini, yaitu ; *grayscale image*, *binarization*, *thinning*, dan *resize*. Kemudian dilanjutkan tahap ekstrasi ciri menggunakan metode *diagonal based feature extraction*yang merupakan pengambilan informasi dari citra aksara Jawa, sehingga setiap aksara Jawa memiliki karakteristik yang berbeda. Ciri-ciri setiap aksara nantinya akan di simpan sebagai data *tranning* yang digunakan untuk menguji kemiripan pada data *testing*. Nilai ekstrasi ciri inilah yang akan membedakan setiap aksara Jawa.
2. Tahap kedua adalah proses uji kemiripan pada penelitian ini menggunakan perhitungan jarak terdekat *eucledian distance*. Perhitungan tersebut membandingkan data *testing* dengan data *tranning* yang ada di dalam *database* hingga menemukan jarak yang terdekat.

Berikut adalah gambaran kerja sistem pada penelitian konversi aksara Jawa ke huruf latin pada citra digital:

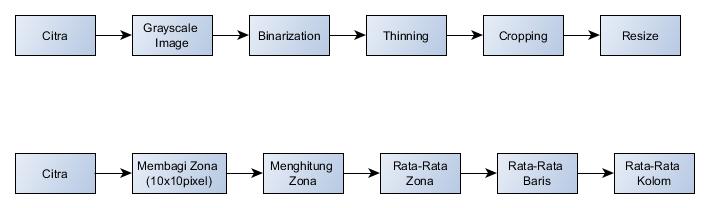
# 

# Gambar 4.1 Gambaran Kerja Sistem

## *Preprocessing*

*Preprocessing* merupakan proses persiapan data sebelum dilakukan ekstraksi ciri pada citra.

Tahapan *preprocessing* pada penilitian ini antara lain :



# Gambar 4.2 Skema *Preprocessing*

1. *Grayscale Image*

Proses merubah citra berwarna ke citra ke abuan. Pada penelitian ini perhitungan *grayscale image* menggunakan perhitungan sebagai berikut :

(4.1)

*Grayscale average* merupakan metode untuk mengubah citra RGB menjadi grayscale dengan menghitung rata – rata dari setiap piksel.

1. *Binarization*

Proses merubah dari citra *grayscale* (8-bit) menjadi citra biner (1-bit). Jadi pada tahap *binarization* citra hanya memiliki dua nila pixel yaitu nilai 0/1, dimana 0 adalah nilai untuk warna hitam dan 1 adalah nilai untuk warna putih yang direpresentasikan dengan piksel 255 (putih). Pada penelitian ini proses *binarization* menggunakan teknik *threshold* dengan menentukan nilai T sebagai tolak ukur *foreground* dan *background*. Pada proses ini, niali T = 125, dimana untuk nilai T < 125 merupakan *backgorund* dengan nilai pixel 0 (hitam) sedangkan T >=125 merupakan *foreground* dengan nilai pixel 255 (putih).

1. *Thinning*

Proses *thinning* merupakan operasi morfologi yang digunakan untuk mengurangi suatu piksel pada citra digital menjadi ukuran yang minimum. Pada penelitian ini proses *thinning* menggunakan algoritma *Zhang-Sheun.* Adapun syarat algoritma yang harus dipenuhi adalah sebagai berikut :

1. (4.2)
2. (4.3)
3. (4.4)
4. (4.5)

Jika salah satu dari keempat kondisi di atas tidak dipenuhi atau dilanggar maka nilai piksel yang bersangkutan tidak diubah. Sebaliknya jika semua kondisi tersebut dipenuhi maka piksel tersebut ditandai untuk selanjutnya piksel akan diubah menjadi 0.

1. *Cropping*

Proses untuk memotong citra secara otomatis dengan kondisi yang kita inginkan. Pada penelitian ini *cropping* digunakan untuk membuang piksel yang tidak digunakan, dengan mendeteksi batas atas batas bawah, batas kanan dan batas kiri ketika mendeteksi adanya piksel 255 pada citra digital.

1. *Resize*

Proses mengubah ukuran citra digital dalam piksel. Pada penelitian ini, dibutuhkan citra dengan ukuran 30x30 piksel sebelum dilakukan tahap ekstrasi ciri menggunakan *library* yang sudah ada di bahasa pemrograman *visual basic*. Proses *resize* merupakan ujung dari tahapan *preprocessing*.

## Ekstrasi Ciri

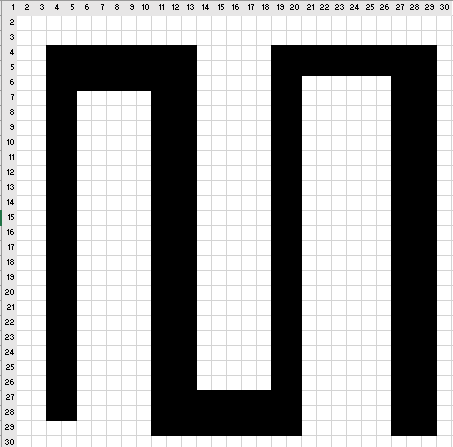
Ekstrasi ciri merupakan proses pengambilan informasi dari citra. Pada penelitian ini akan dilakukan ekstraksi untuk mendapatkan ciri – ciri dari citra berupa nilai rata – rata melalui proses berikut :



# Gambar 4.3 Tahap Ekstrasi Ciri

1. Membagi Zona

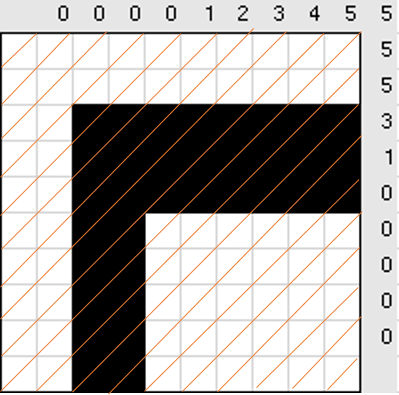
Proses membagi zona merupakan proses memotong citra menjadi beberapa zona atau bagian. Dimana citra yang diproses untuk ekstrasi ciri adalah citra dengan ukuran 30x30 piksel. Selanjutnya citra tersebut dibagi menjadi 10x10 piksel, seperti pada gambar 4.5, sehingga menjadi sembilan zona. Pada dasarnya untuk pembagian zona tidak ada penetapan yang tetap pada metode *diagonal based feature extraction* ini. Nantinya yang membedakan adalah jumlah diagonal yang akan digunakan pada tahap selanjutnya.



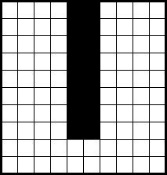
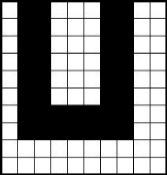
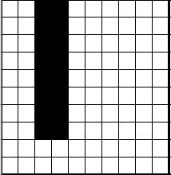
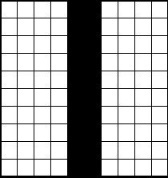
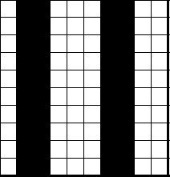
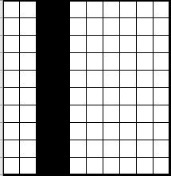
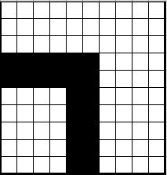
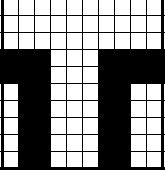
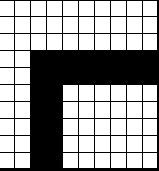
# Gambar 4.4 Citra Ukuran 30x30 piksel

# Gambar 4.5 Pembagian Zona (10x10 piksel)

1. Menghitung Nilai Zona

Setelah citra dibagi menjadi sembilan zona. Selanjutnya meghitung nilai zona secara diagonal, seperti pada Gambar 4.6. Zona yang dihitung merupakan *foreground* yaitu piksel dengan nilai 255 (putih) yang dipresentasikan pada gambar 4.6 dengan warna hitam. Gambar 4.6 merupakan cara menghitung nilai zona yang didapatkan dengan menjumlahkan jumlah *foreground*. Nilai zona pada gambar 4.6 adalah 34. Perhitungan nilai zona dilakukan untuk sembilan zona yang lainnya.

# Gambar 4.6 Contoh Menghitung Nilai Zona



# Gambar 4.7 Pembagian 9 zona.

Berikut adalah nilai masing-masing zona.

1. Menghitung Rata-Rata Zona

Setelah mendapatkan sembilan nilai zona, maka langkah selanjutnya adalah menghitung rata – rata masing – masing zonadisebut Zn dimana 1 ≤ n ≤ 9.

(4.6)

1. Menghitung Rata – Rata Baris pada Rata – Rata Zona

Setelah mendapatkan rata – rata zona dari sembilan zona selanjutnya menghitung rata – rata setiap baris pada sembilan zona.

(4.7)

Baris 1 = (Z1 + Z2 + Z3) / 3 = (1,368 + 1,789 + 1,158) / 3 = 1.44

Baris 2= (Z4 + Z5 + Z6) / 3 = (1,053 + 2,105 + 1,053) / 3 = 1.40

Baris 3= (Z7 + Z8 + Z9) / 3 = (0,842 + 2 + 0.842) / 3 = 1,23

1. Menghitung Rata – Rata Kolom pada Rata – Rata Zona

Setelah mendapatkan rata – rata zona dari sembilan zona selanjutnya menghitung rata – rata setiap baris pada sembilan zona.

(4.8)

Kolom 1 = (Z1 + Z4 + Z7) /3 = (1,368 + 1,053 + 0,842) / 3 = 1,09

Kolom 2 = (Z2 + Z5 +Z8) / 3 = (1,789 + 2,105 + 2) / 3 = 1.96

Kolom 3 = (Z3 + Z6 + Z9) / 3 = (1,158 + 1,053 + 0,842) = 1,02

## Uji Kemiripan

Perhitungan *ecludiandistance* merupakan teori pengukuran jarak dari dua buah titik. Teori ini berkaitan dengan teori teorema *phytagoras* untuk menghitung jarak terdekat dengan menggunakan rumus sebagai betikut.

(4.9)

Keterangan :

d = jarak

b = rata – rata baris

k = rata – rata kolom

## Analisa Kebutuhan Sistem

Pada sub sistem bab ini akan dijelaskan kebutuhan – kebutuhan untuk membangun sistem konversi aksara Jawa ke huruf latin.

* + 1. Kebutuhan Perangkat Lunak

Untuk dapat melakukan perancangan dan menjalankan aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction*perlu memperhatikan kebutuhan perangkat lunak yaitu diantaranya sebagai berikut:

1. *Operating System*: Windows 10
2. *Software* untuk membuat Aplikasi *desktop*: Microsoft Visual Studio 2012
3. *Software* untuk *database:*SQL Server 2017
   * 1. Kebutuhan Perangkat Keras

Spesifikasi minimum perangkat keras untuk menjalankan perangkat lunak diatas adalah:

1. CPU 2.0 GHz
2. RAM 4GB
3. HDD 150GB
4. *Camera* 8MP

## Perancangan

Pada poin ini akan dijelaskan perancangan dalam pembuatan sistem konversi aksara Jawa ke huruf latin menggunakan metode *diagonal based feature extraction*.

* + 1. Perancangan *Database*

Dari penjelasan sebelumnya, telah diutarakan sistem ini membutuhkan sistem penyimpanan untuk menyimpan data *training*. Maka dirancang *database* dengan struktur tabel sebagai berikut:

Tabel 4.1 Tabel Citra

|  |  |  |
| --- | --- | --- |
| **Nama** | **Tipe Data** | **Keterangan** |
| kodeCitra | Int | Menyimpan kode citra |
| namaCitra | Varchar | Menyimpan nama citra |
| fileCitra | Varchar | Menyimpan file cirectory citra |

Tabel 4.2 Tabel Rata-Rata

| **Nama** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| kodeRata | Int | Menyimpan kode rata |
| kodeCitra | Int | Menyimpan kode citra |
| rataBaris1 | Float | Menyimpan nilai baris 1 |
| rataBaris2 | Float | Menyimpan nilai baris 2 |
| rataBaris3 | Float | Menyimpan nilai baris 2 |
| rataKolom1 | Float | Menyimpan nilai kolom 1 |
| rataKolom2 | Float | Menyimpan nilai kolom 2 |
| rataKolom3 | Float | Menyimpan nilai kolom 3 |

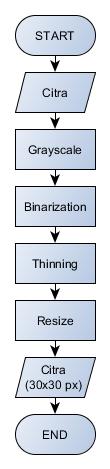
Tabel 4.3 Tabel Hasil

|  |  |  |
| --- | --- | --- |
| **Nama** | **Tipe Data** | **Keterangan** |
| kodehasil | Int | Menyimpan kode hasil |
| nilaihasil | Float | Menyimpan nilai hasil |

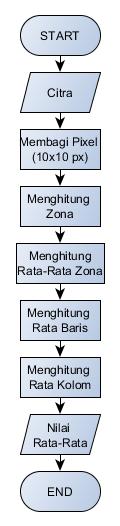
* + 1. Perancangan Proses

Ada tiga proses pada sistem konversi aksara Jawa ke huruf latin pada citra digital, yaitu *preprocessing*, ekstrasi ciri dan uji kemiripan.

* + - 1. Proses *Preprocessing*

Pada proses ekstrasi, masukan berupa citra digital yang berektensi atau berformat .jpg. Proses *preprocessing* yang harus dilewati memiliki beberapa tahap. Alur *preprocessing* digambarkan pada Gambar 4.8.

# Gambar 4.8*Flowchart Preprocessing*

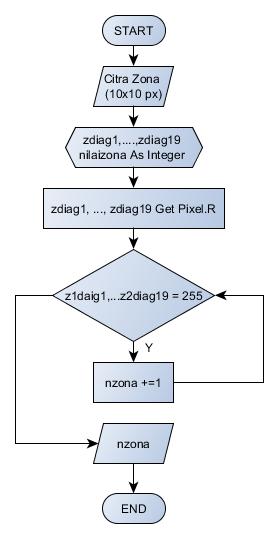
* + - 1. Proses Ekstrasi Ciri

# Gambar 4.9*Flowchart*Ekstrasi Ciri

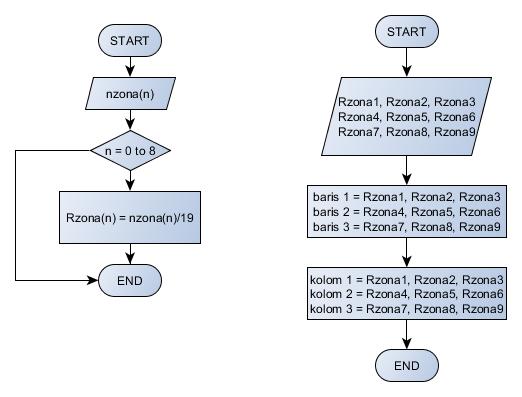
Pada proses ekstrasi ciri, masukan merupakan hasil dari citra *preprocessing* dengan ukuran piksel 30x30. Selanjutnya alur ekstrasi ciri secara umum digambarkan pada gambar 4.9

Proses membagi zona merupakan sub proses yang ada di ekstrasi ciri. Proses menghitung zona telah digambarkan pada gambar 4.5. Untuk membuatnya menjadi kode program dirancang sebuah alur proses. Alur proses membagi zona dijelaskan pada *flowchart* berikut :

# Gambar 4.10*Flowchart*Membagi Zona

Proses menghitung zona merupakan sub proses yang ada di ekstrasi ciri. Proses menghitung zona telah digambarkan pada gambar 4.6 . Untuk membuatnya menjadi kode program dirancang sebuah alur proses. Alur proses membagi zona dijelaskan pada *flowchart* berikut :

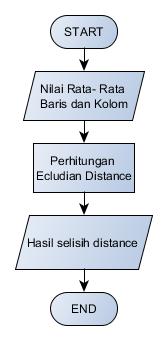
# Gambar 4.11*Flowchart*Menghitung Zona

 Sama halnya dalam sub proses yang lainnya dalam proses ekstrasi ciri yaitu menghitung rata – rata zona, menghitung rata – rata baris pada rata – rata zona dan menghitung rata – rata kolom pada rata – rata zona.

# Gambar 4.12*Flowchart*Menghitung Rata – Rata Zona

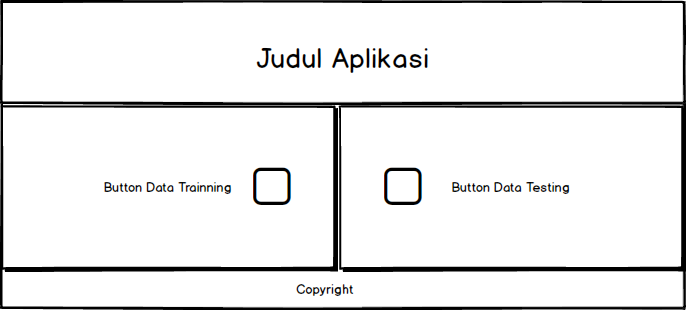
# Gambar 4.13*Flowchart*Menghitung Rata – Rata Baris dan Kolom

* + - 1. Proses Uji Kemiripan

Pada proses uji kemiripan, masukan berupa nilai dari rata – rata baris dan kolom yang kemudian dibandingkan dengan nilai dari rata – rata baris dan kolom pada citra dalam *database* menggunakan perhitungan *ecludian distance*. Perhitungan *ecludian distance* telah dijelaskan pada persamaan 4.10. Alur uji kemiripan digambarkan pada *flowchart* 4.14.

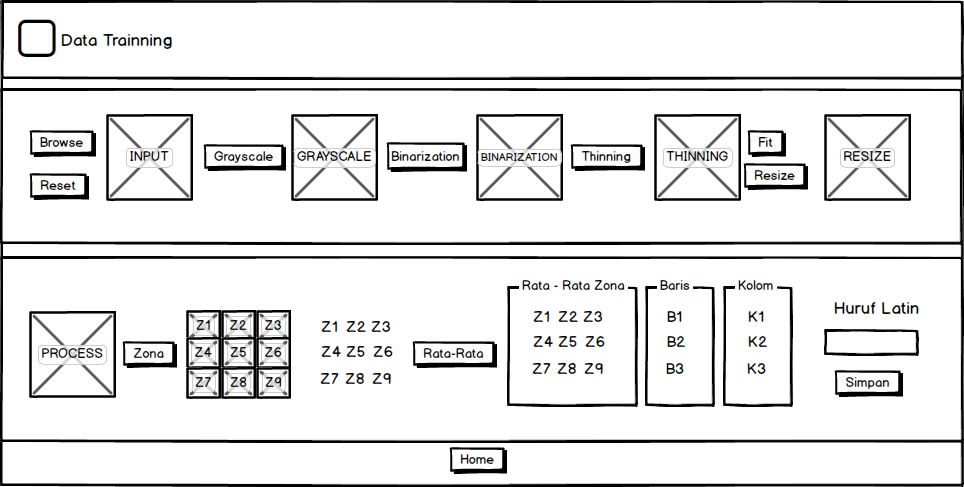
# Gambar 4.14*Flowchart* Uji Kemiripan

* + 1. Perancangan *User Interface*
       1. Rancangan Halaman Utama

Halaman utama saat memasuki aplikasi. Halaman ini berisi dua *button* yang dapat mengarahkan untuk melakukan *tranning* atau *testing*. Button ‘data*tranning*’ akan mengarahkan untuk menambah koleksi citra, sedangkan untuk button‘data *testing*’ akan mengarahkan untuk melakukan uji coba.

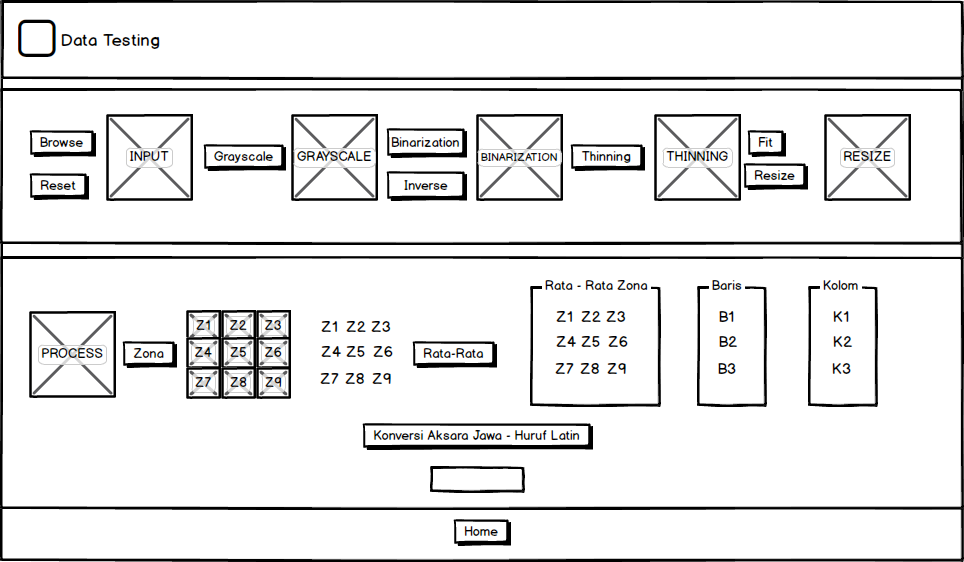
# Gambar 4.15 Rancangan Halaman Utama

* + - 1. Rancangan Tampilan Halaman*Trainning*

Merupakan rancangan halaman yang digunakan untuk menambahkan koleksi citra pada aplikasi. Citra yang ditambahkan akan disimpan ke dalam *database*.

# Gambar 4.16 Rancangan Halaman *Trainning*

* + - 1. Rancangan Tampilan Halaman*Testing*

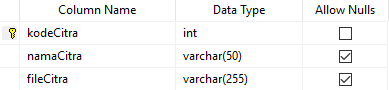
Merupakan rancangan halaman yang digunakan untuk melakuakn uji coba pada aplikasi.

# Gambar 4.17 Rancangan Halaman *Testing*

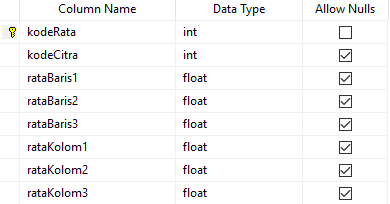
BAB V. IMPLEMENTASI

Pada bab ini akan dijelaskan implementasi aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction* dengan rancangan yang telah dijelaskan pada Bab IV Analisis dan Perancangan.

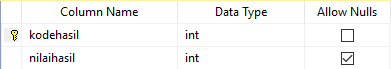
## Implementasi *Database*

Untuk menyimpan informasi citra dan hasil ekstrasi citra dibuat sebuah *database*. Fungsi dari tiap – tiap *field* ditunjukkan pada Bab 4 Sub Bab 4.6.1.

# Gambar 5.1 Implementasi Tabel Citra



# Gambar 5.2 Implementasi Tabel Rata – Rata



# Gambar 5.3 Implementasi Tabel Hasil

## Implementasi Antar Muka

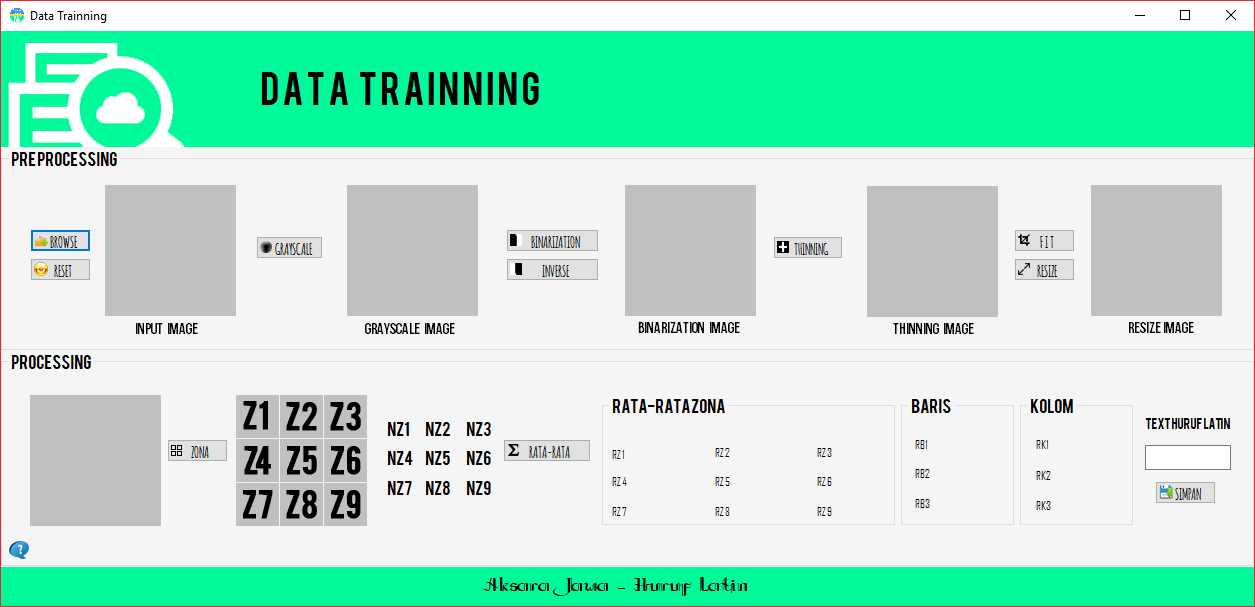
Berikut ini implementasi antar muka dari aplikasikonversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction*:

* + 1. Tampilan Halaman Utama

Pada tampilan halaman utama terdapat dua menu yaitu menu *training* dan menu *testing.* Selain kedua menu utama, pada menu utama terdapat *button* untuk mengarahkan ke halaman bantuan.

# Gambar 5.4 Tampilan Halaman Utama

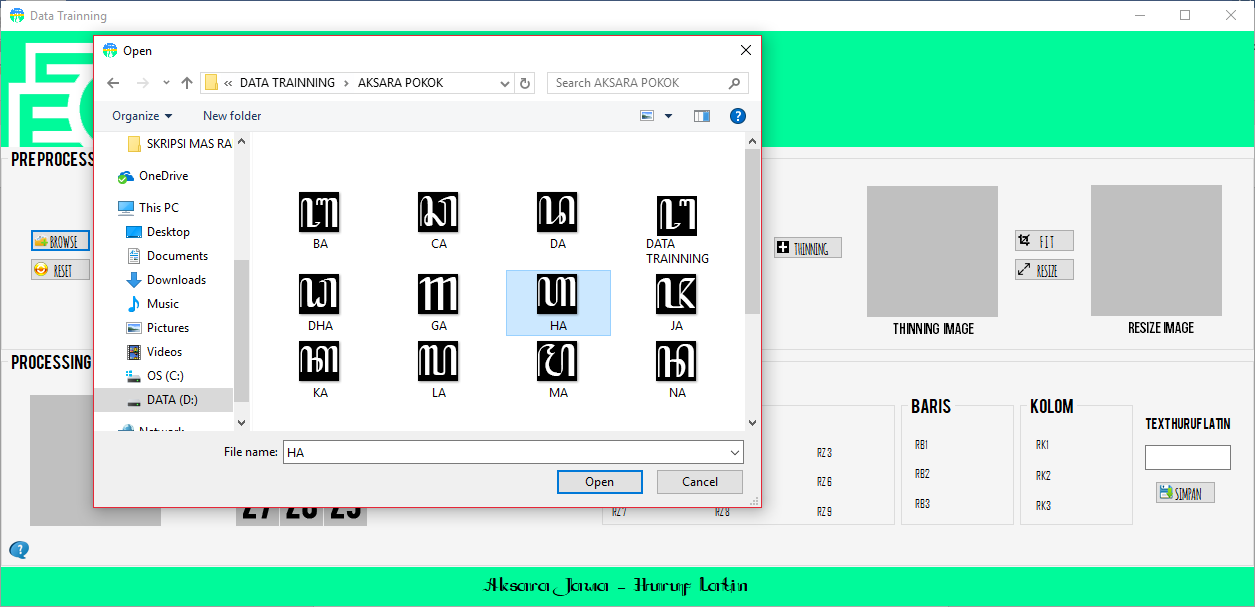
* + 1. Tampilan Halaman Data *Trainning*

Pada sub bab implementasi data *training* ini akan dijelaskan bagaimana proses *training* aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction*menggunakan bahasa pemograman *visual basic*. Dibawah ini merupakan langkah-langkah penggunaan aplikasi yang terdapat pada proses *training*.

# Gambar 5.5 Tampilan HalamanData *Trainning*

* + - 1. Proses *Input* Citra Digital

Langkah awal dari proses *training* yaitu *user* melakukan *input* citra aksara Jawa. Langkah-langkah untuk melakukan *input* citra aksara Jawa adalah sebagai berikut :

Klik *button* ‘Browse’ – selanjutnya akan diarahkan pada *file dialog* untuk memilih citra aksara Jawa yang akan di proses. Lalu pilih ‘Open’. Maka citra akan ditampilkan pada *picturebox*.

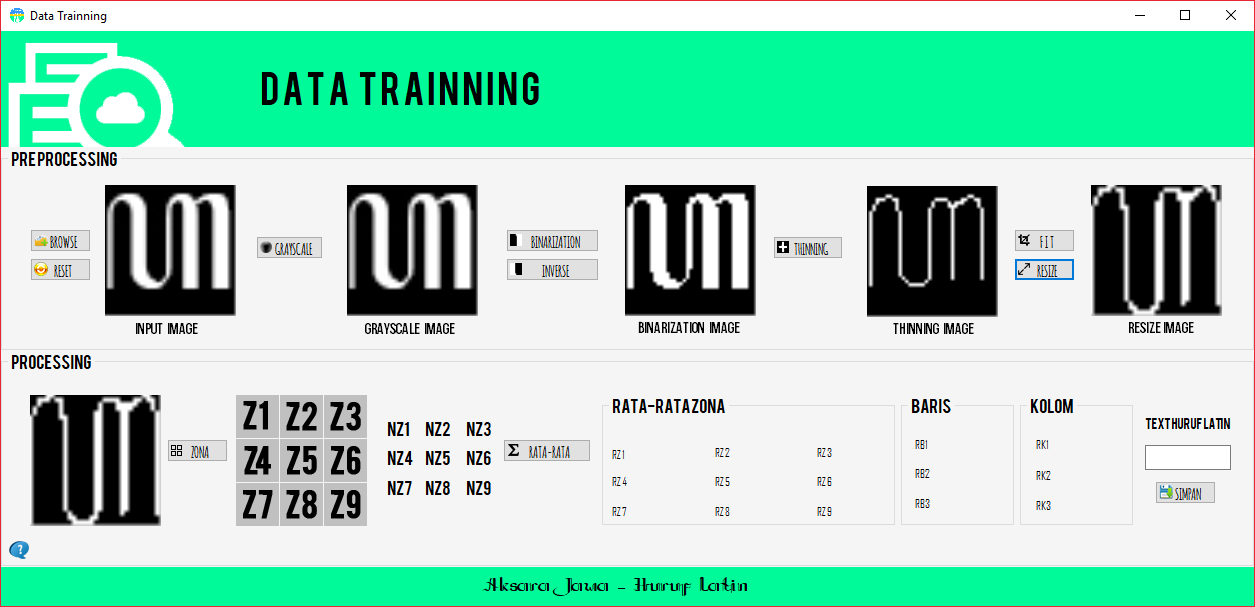
# Gambar 5.6 Tampilan *FileDialog* untuk *Input* Citra Aksar Jawa

# Gambar 5.7 Tampilan *Input* Citra Aksara Jawa

* + - 1. Proses *Preprocessing*

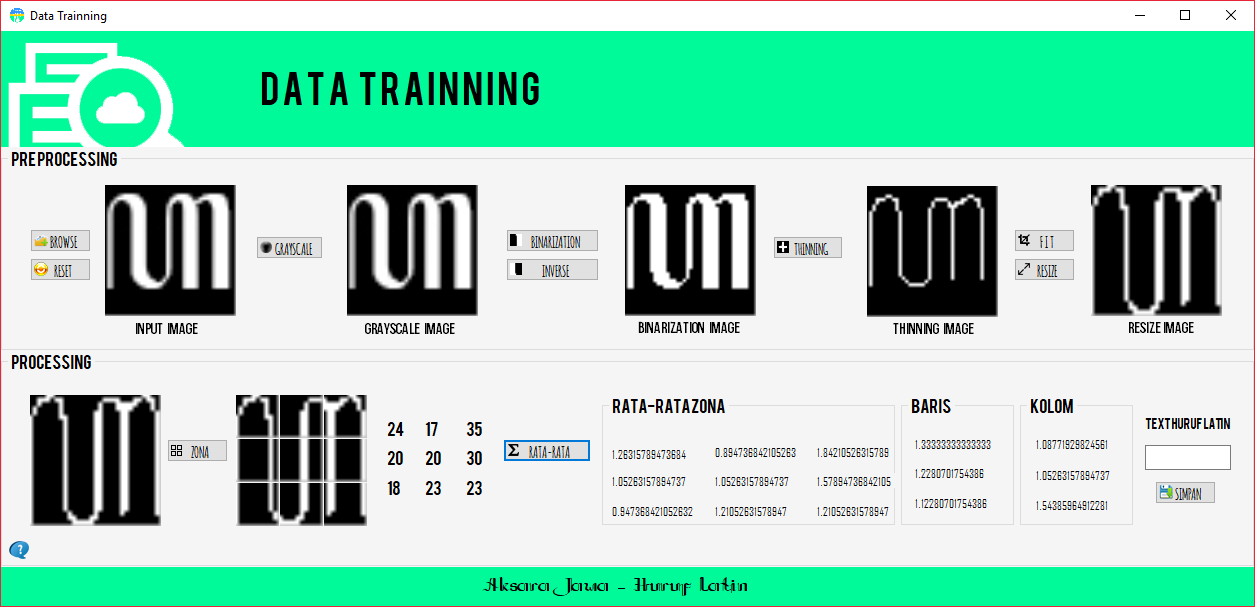
Langkah selanjutnya setelah citra aksara Jawa di *input* kan ke dalam sistem, selanjutnya adalah proses *preprocessing* yang bertujuan untuk mempersiapkan citra

sebelum proses ekstrasi ciri.

Langkah-langkah untuk melakukan *preprocessing* adalah sebagai berikut :Klik *buttonGrayscale* – *Binarization* – *Thinning* – *Fit* - *Resize*. Klik *button*‘inverse’ jika proses *binarization* menghasilkan tulisan pada citra aksara Jawa dengan nilai piksel 0 seperti pada Gambar 5.14. Proses *inverse* ini bertujuan untuk merubah piksel 0 menjadi 255 dan piksel 255 menjadi 0, hal ini perlu dilakukan karena proses selanjutnya (*Thinning*) membutuhkan nilai piksel 255 sebagai *foreground*.

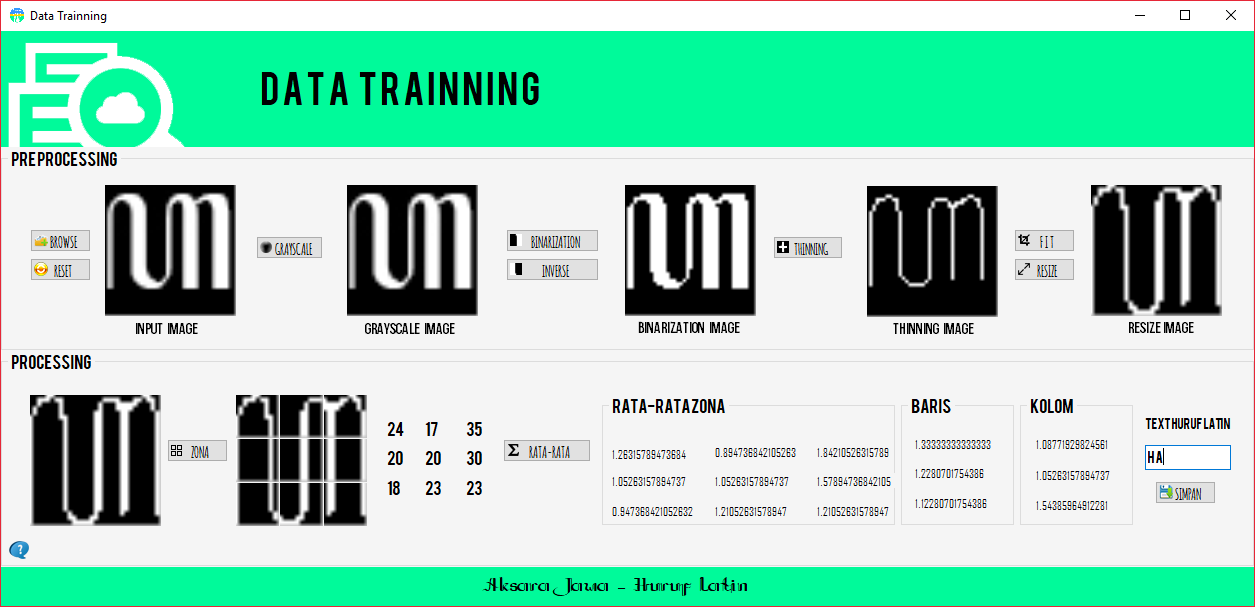
# Gambar 5.8 Tampilan Proses *Preprocesing.*

* + - 1. Proses Ekstrasi Ciri

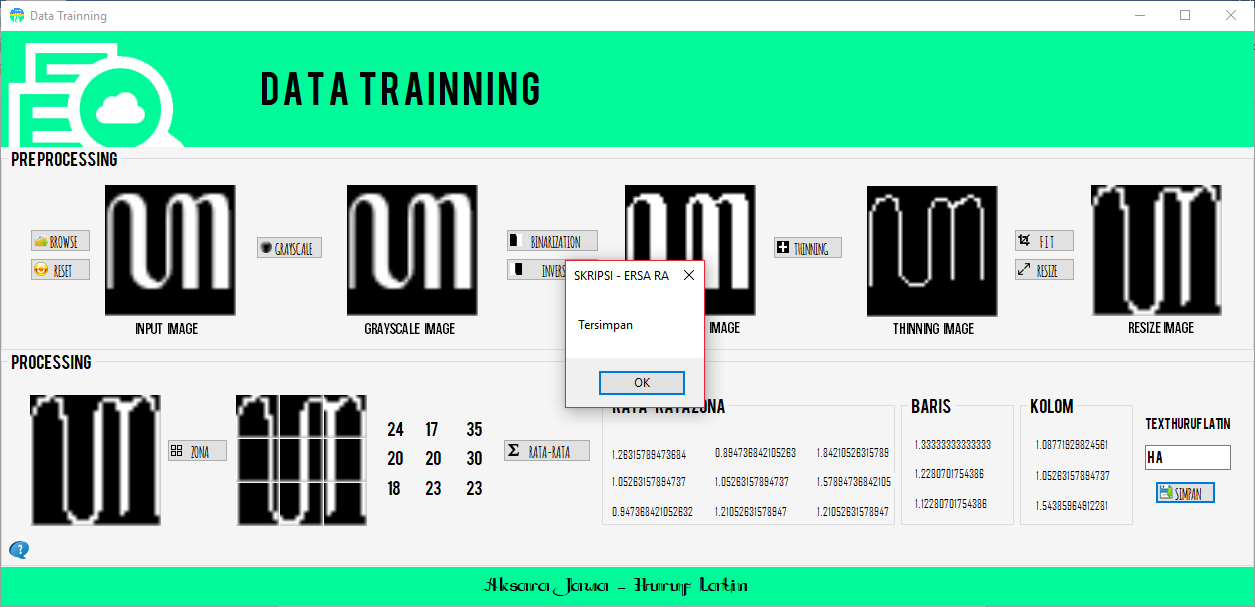
Langkah selanjutnya setelah citra aksara Jawa dilakukan *preprocessing* yaitu melakukan proses ekstraksi citradengan menggunakan metode *diagonal based feature extraction.* Adapun perhitungan tersebut telah dijelasakan pada Bab 4 Sub Bab 4.3. Untuk proses menghitung ekstrasi ciri klik *button* ‘Zona’ – lalu klik *button* ‘Rata-Rata’.

# Gambar 5.9 Tampilan Proses Perhitungan Ekstrasi Ciri*.*

* + - 1. Proses Menyimpan *Dataset*

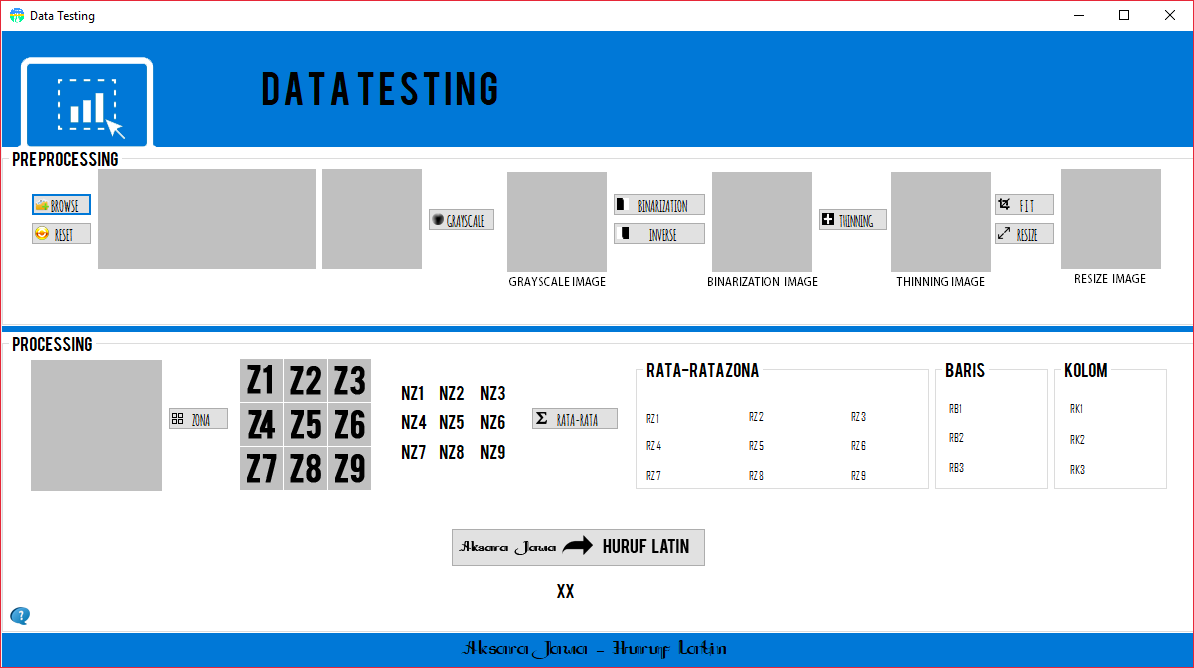
Langkah selanjutnya yaitu memasukkan nilai ekstrasi ciri ke *database* yang selanjutnya akan dijadikan *dataset,* antara lain nilai rata – rata baris 1 sampai 3 dan rata – rata kolom 1 sampai 3. Untuk proses menyimpan silahkan masukkan text huruf latin ke text box yang telah disediakan - lalu pilih *button* ‘Simpan’.

# Gambar 5.10 Tampilan Proses Menyimpan Dataset*.*

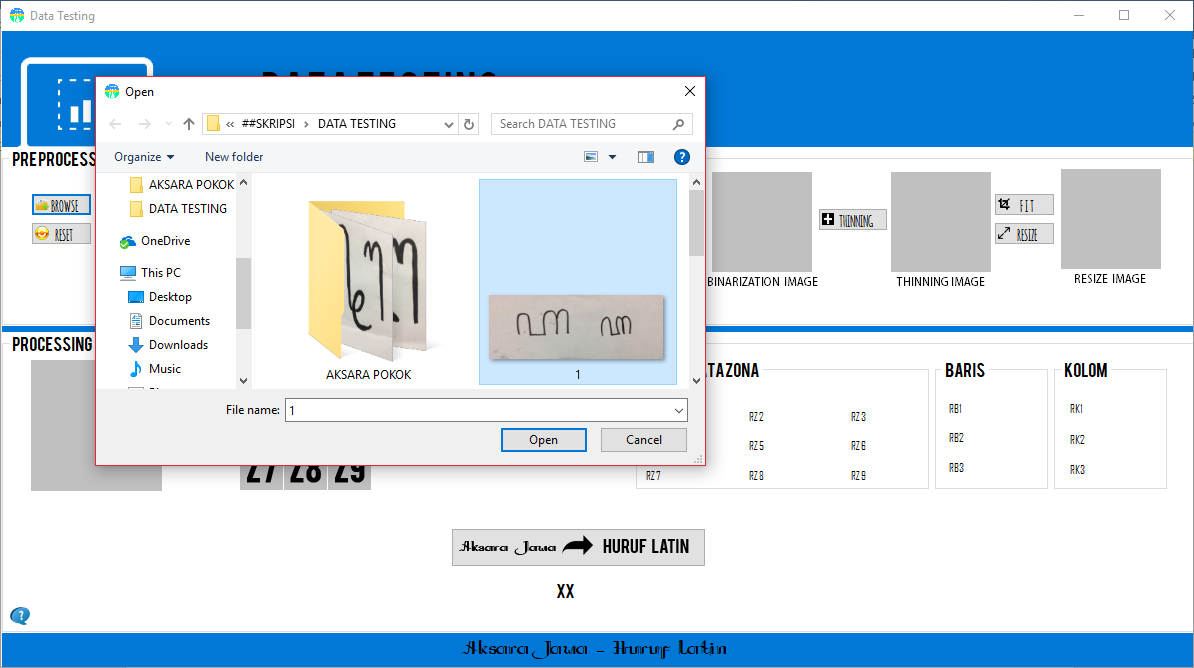


# Gambar 5.11 Tampilan Proses Berhasil Menyimpan Dataset*.*

* + 1. Tampilan Halaman Data *Testing*

Pada sub bab implementasi pada halamandata *testing*ini akan dijelaskan bagaimana proses *testing*aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction*menggunakan bahasa pemograman *visual basic*. Dibawah ini merupakan langkah-langkah penggunaan aplikasi yang terdapat pada proses *testing.*

# Gambar 5.12 Tampilan HalamanData *Testing*

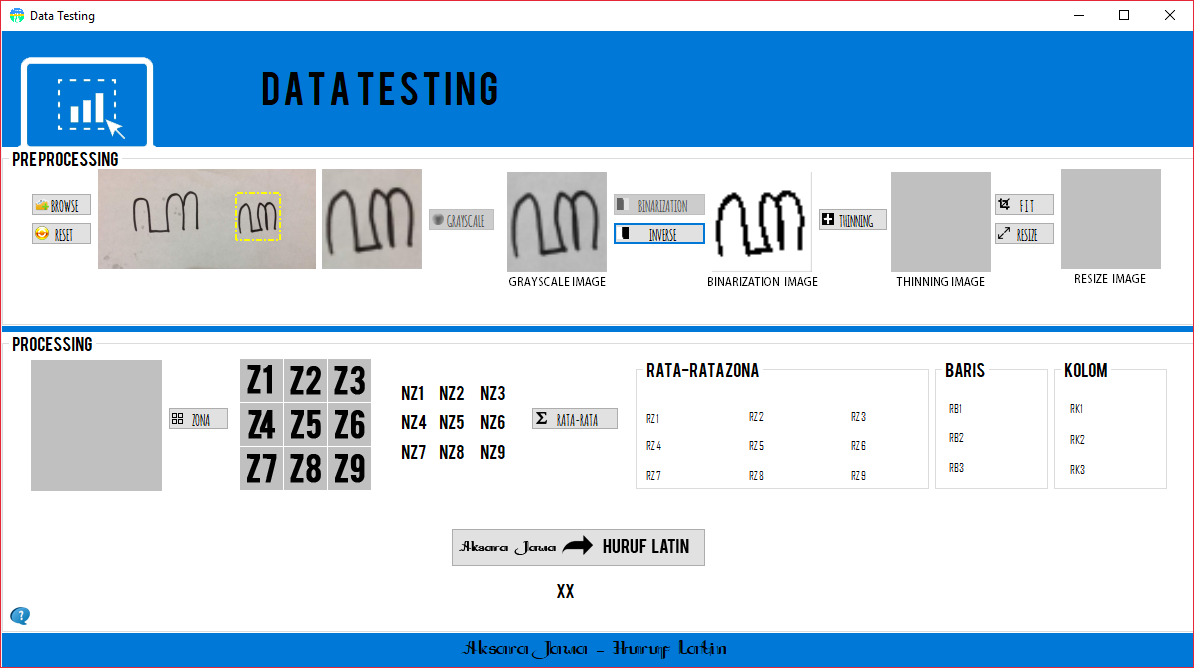
* + - 1. Proses *Input* Citra Digital

# Gambar 5.13 Tampilan *File Dialog* untuk *Input* Citra Aksara Jawa.

Langkah awal dari proses *testing*yaitu *user* melakukan *input* citra aksara Jawa. Langkah-langkah untuk melakukan *input* citra aksara Jawa adalah sebagai berikut :

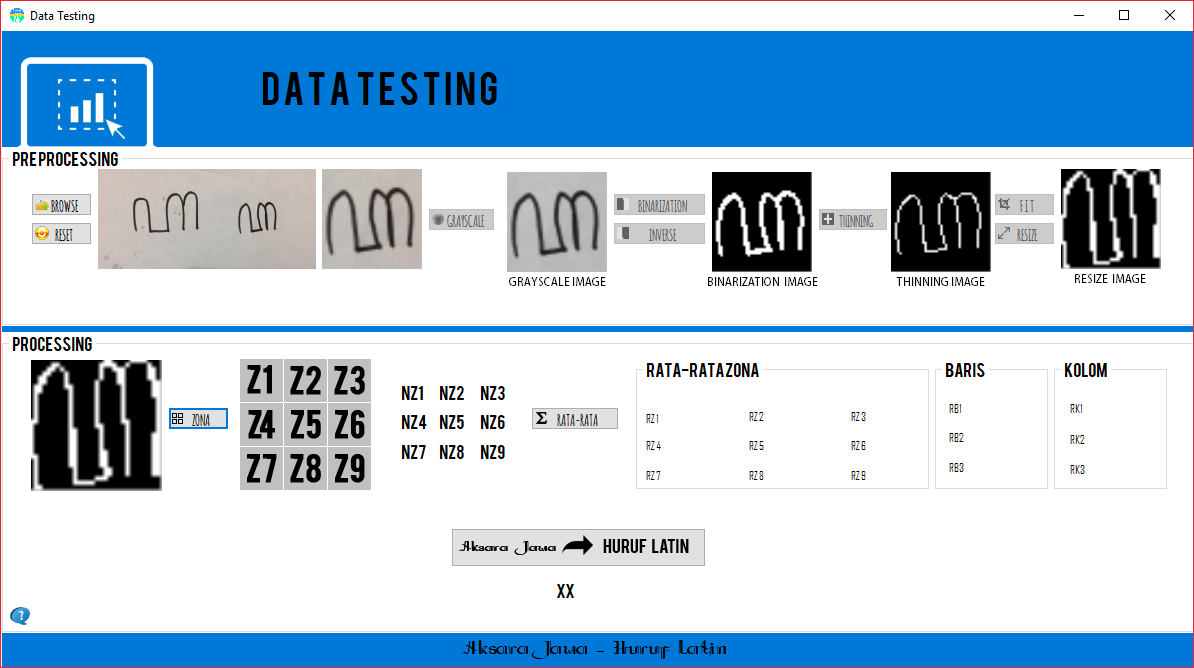
Klik *button* ‘Browse’ – selanjutnya akan diarahkan pada *file dialog* untuk memilih citra aksara Jawa yang akan di proses. Lalu pilih ‘Open’. Maka citra akan ditampilkan pada *picturebox*.

* + - 1. Proses *Preprocessing*

Langkah selanjutnya setelah citra aksara Jawa di *input* kan ke dalam sistem, selanjutnya adalah proses *preprocessing* yang bertujuan untuk mempersiapkan citra sebelum proses ekstrasi ciri.

# Gambar 5.14 Tampilan *Preprocessing* Tanpa Proses *Inverse*.

Langkah-langkah untuk melakukan *preprocessing* adalah sebagai berikut :

Pilih area huruf yang diinginkan, karena aplikasi ini hanya bisa mendeteksi setiap huruf. Menentukan area huruf hanya dilakukan jika citra lebih dari satu aksara Jawa. Selanjutnya, klik *buttonGrayscale – Binnarization – Thinning – Fit - Resize*. Klik *button inverse* jikaproses *binarization* menghasilkan tulisan pada citra aksara Jawa dengan nilai piksel 0 seperti pada Gambar 5.14. Proses *inverse* ini bertujuan untuk merubah piksel 0 menjadi 255 dan piksel 255 menjadi 0, hal ini perlu dilakukan karena proses selanjutnya (*thinning*) membutuhkan nilai piksel 255sebagai *foreground*. .

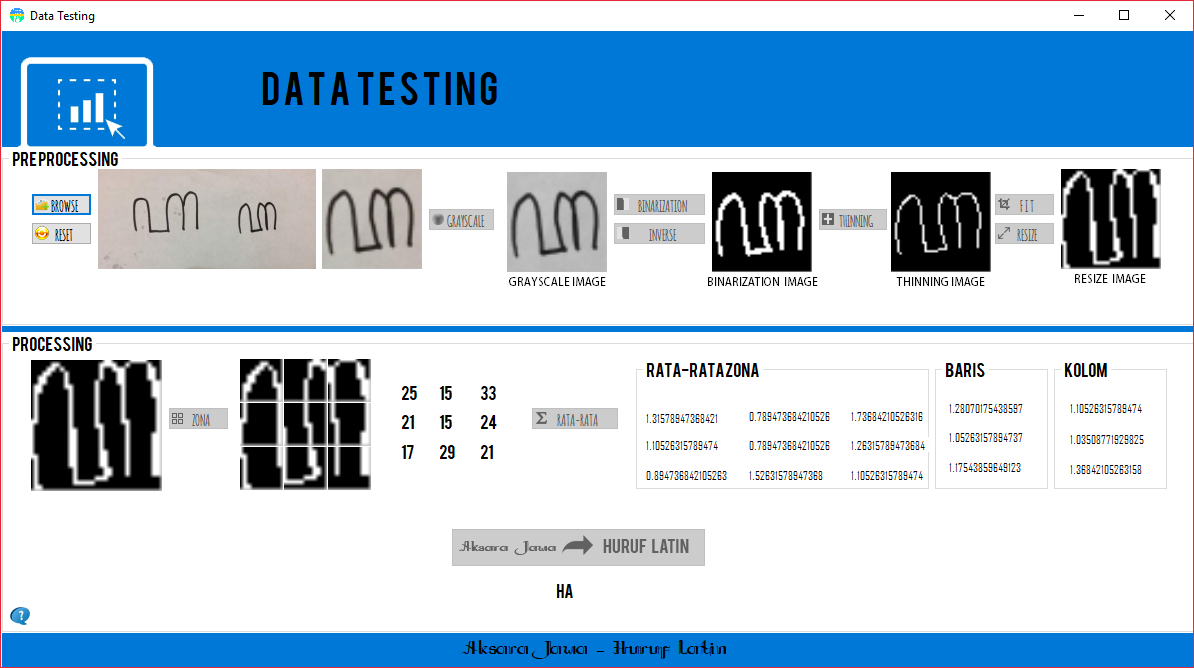
# Gambar 5.14 Tampilan *Preprocessing* dengan Proses *Inverse*.

* + - 1. Proses Ekstrasi Ciri

Langkah selanjutnya setelah citra aksara Jawa dilakukan *preprocessing* yaitumelakukan proses ekstraksi citradengan menggunaka metode *diagonal based feature extraction.* Adapun perhitungan tersebut telah dijelasakan pada Bab 4 Sub Bab 4.3. Untuk proses menghitung ekstrasi ciri klik *button* ‘Zona’ – lalu klik *button* ‘Rata-Rata’

# Gambar 5.15 Tampilan Proses Perhitungan Ekstrasi Ciri*.*

* + - 1. Proses Uji Kemiripan

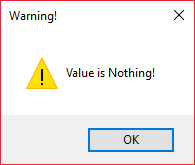
Langkah selanjutnya menghitung uji kemiripan dengan menggunakan perhitungan *ecludian distance* yang telah dijabarkan pada Bab 4 Sub Bab 4.4. Untuk melakukan uji kemiripan tekan *button* ‘Aksara Jawa – Huruf Latin’. Setelah itu huruf latin akan ditampilkan pada text box.

# Gambar 5.16 Tampilan Proses Uji Kemiripan

* + 1. Tampilan Pesan Error

Pesan *error* akan dimunculkan oleh sistem ketika *user* menekan *buttonGrayscale, Binarization, Inverse, Thinning, Fit, Resize* dan Zona jika belum menginputkan citra digital.

# Gambar 5.16 Tampilan Pesan *Error* jika Belum *Input* Citra.

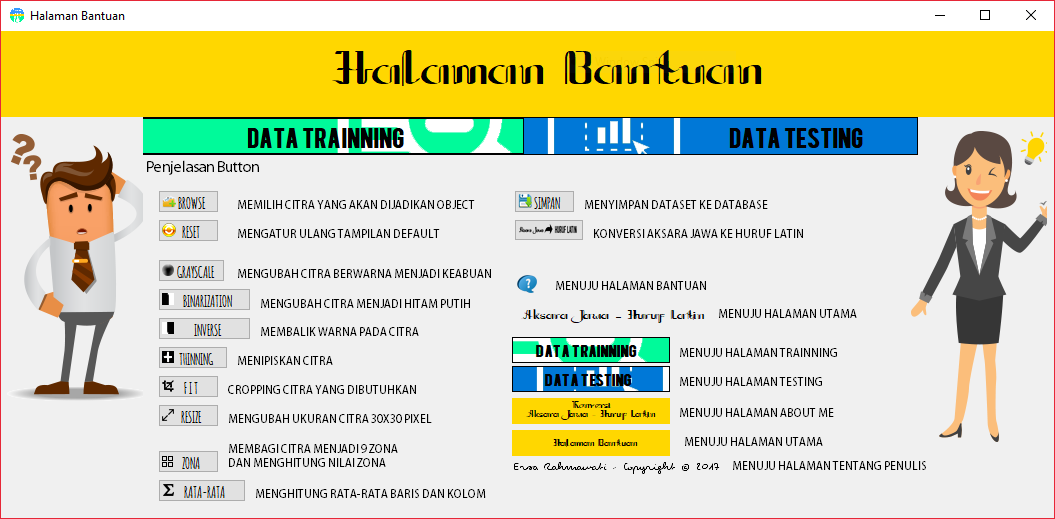
Selain itu sistem juga akan menampilkan pesan *error*  ketika belum ada nilai yang terisi di dalam label jika *user*  menekan *button* ‘Simpan’ dan ‘Rata-Rata’ pada halaman *training* dan *button* ‘Aksara Jawa – Huruf Latin’ dan ‘Rata-Rata’ pada halaman *testing*. Pesan *error*dapat dilihat pada Gambar 5.17.

# Gambar 5.17 Tampilan Pesan *Error* jika Tidak Ada Nilai pada Label

* + 1. HalamanBantuan

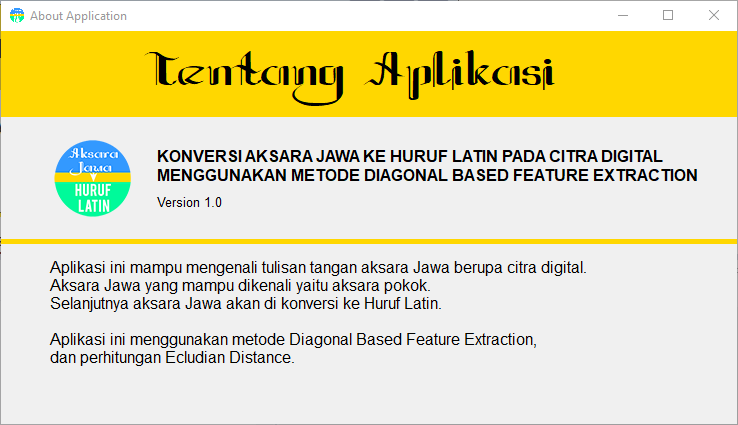
Apabila *user* tidak mengerti mengenai fungsi daripada fitur-fitur yang tersedia pada aplikasi, *user* dapat klik *button*help yang ada pada pojok kiri bawah di setiap halaman *training,* halaman *testing* dan halamna utama. Isi dari halaman bantuan dapat dilihat pada Gambar 5.18.

# Gambar 5.18*Button* Halaman Bantuan



# Gambar 5.19Tampilan Halaman Bantuan

* + 1. Halaman Penjelasan tentang Aplikasi

 Halaman tentang aplikasi menjelaskan secara singkat aplikasi konversi aksara Jawa ke huruf latin pada citra digital. Disini telah teretera batasan – batasan yang ada pada aplikasi tersebut.

# Gambar 5.20 Tampilan Halaman Tentang Aplikasi

BAB VI. PENGUJIAN DAN PEMBAHASAN

## Pengujian

Setelah melakukan implementasi sistem, maka pada bab pengujian dan pembahasan ini dilakukan tahap untuk menguji hasil implementasi sistem. Pengujian pada aplikasi ini dilakukan dengan menggunakan pengujian unit dan pengujian akurasi. Pengujian sistem dilakukan dengan maksud untuk mengetahui apakah konten dari aplikasi sudah berjalan sesuai dengan fungsinya masing-masing. Sedangkan pengujian akurasi digunakan untuk menguji tujuan utama dari aplikasi pengenalan tulisan tangan ini bahwa dengan menerapkan metode *diagonal based feature extraction*untukkonversi aksara Jawa ke huruf latin pada citra digital.Hasil pengujian akan dimanfaatkan untuk menyempurnakan kinerja program dan digunakan dalam pengembangan lebih lanjut.

* + 1. Pengujian Unit

Untuk tahap pengujian unit menggunakan metode *blackbox*. Metode ini memungkinkan adanya pengembangan untuk melatih seluruh fungsi pada sistem. Metode ini digunakan untuk mendemonstrasikan jalannya aplikasi dan menemukan kesalahan saat aplikasi dijalankan. Dengan menggunakan metode ini dapat dinilai apakah *input* yang diterima dan *output* yang dihasilkan sudah tepat atau belum. Pengujian unit dilakukan pada halaman *training*  dan halaman *testing*. Pada pengujian unit akan dijelaskan hasil yang diharapkan dan hasil pengujian sehingga mendapatkan kesimpulan keberhasilan dari pengujian tersebut.

* + - 1. Pengujian Unit *Trainning*

Pada pengujian unit *training* dilakukan secara fungsional untuk menemukan kesalahan pada saat aplikasi dijalankan. Berikut uraian tabel hasil pengujian dari pengujian sistem*training* yang telah diuraikan pada Tabel 6.1:

Tabel 6.1 Pengujian Unit Training

| **Konversi Aksara Jawa ke Huruf Latin pada Citra Digital** | | | | |
| --- | --- | --- | --- | --- |
| **No.** | **Skenario Pengujian** | **Hasil yang diharapkan** | **Hasil pengujian** | **Kesimpulan** |
|  | Pada form *train-ning* terdapat be-berapa *picture box,* jika *image* belum diinput-kan kemudian *user* klik *buttonGrayscale, Binarization, Inverse, Thin-ning, Fit, Re-size*, Zona. | Muncul *message box* “Image is Nothing!” | Sesuai Harapan | Berhasil |
|  | Pada form *train-ning* terdapat beberapa label yang digunakan untuk menam-pilkan perhi-tungan metode*,* jika label ko-song kemudian *user* klik button Rata-Rata. | Muncul *message box* “Value is Nothing!” | Sesuai Harapan | Berhasil |
|  | Pada form *training* terdapat *text box* yang digunakan untuk menginputkan huruf latin, jika *text box* akan kosong kemudian *user* klik *button* Simpan. | Muncul *message box* “Value is Nothing!” | Sesuai Harapan | Berhasil |
|  | Klik button “Browse” pilih gambar yang akan dibuat data set dan akan muncul jendela *open file dialog.* | *Picture box image input* akan menampilkan citra yang dipilih. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button re-set.* | Sistem menghapus *image* pada semua *picture box image*, menghidupkan kembali *button*  yang mati, menghapus nilai pada label. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button gray-scale.* | *Picture box image grayscale* akan merubah citra menjadi citra keabuan(8 bit). | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button binarization.* | *Picture box image binarization* akan merubah citra menjadi 1 bit (hitam/putih). | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button inverse.* | *Picture box image binarization* akan membalik warna hitam menjadi putih dan warna putih menjadi hitam | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button thinning.* | *Picture box image thinning* akan menipiskan citra | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button Fit.* | *Picture box image fit* akan memotong citra sesuai dengan batas citra dengan warna putih. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button Resize.* | *Picture box image fit* akan merubah ukuran piksel citra menjadi 30x30 pikseldan menampilkan citra 30x30 pixel ke *picture box output image* | Sesuai harapan | Berhasil |
|  | Pada *group box processing,* klik *button* zona*.* | Sistem akan memotong *picture box output image*  menjadi 9 bagian dan ditampilkan pada masing-masing *picture box* zona dan selanjutnya mampu menghitung nilai zona | Sesuai harapan | Berhasil |
|  | Pada *group box processing,* klik *button* rata-rata*.* | Sistem akan menghitung rata-rata nilai masing-masing zona dan menghitung rata-rata baris 1 sampai 3 serta rata-rata baris 1 sampai 3. | Sesuai harapan | Berhasil |
|  | Pada group box*processing*, klik *button* rata-rata. | Sistem akan menyimpan ke *database* dan muncul *message box* tersimpan sebanyak dua kali. | Sesuai harapan | Berhasil |
|  | Dibagian bawah, klik *button help.* | Menampilkan informasi halaman bantuan | Sesuai harapan | Berhasil |

* + - 1. Pengujian Unit Testing

Pada pengujian unit *testing* dilakukan secara fungsional untuk menemukan kesalahan pada saat aplikasi dijalankan. Berikut blackbox dari pengujian sistem*testing*:

Tabel 6.2 Pengujian Unit *Testing*

| **Konversi Aksara Jawa ke Huruf Latin pada Citra Digital** | | | | |
| --- | --- | --- | --- | --- |
| **No.** | **Skenario Pengujian** | **Hasil yang diharapkan** | **Hasil pengujian** | **Kesimpulan** |
|  | Pada form *testing* terdapat beberapa *picture box,* jika *image* belum diinput-kan kemudian *user* klik *button Grayscale, Binarization, Inverse, Thinning, Fit, Resize*, Zona. | Muncul *message box* “Image is Nothing!” | Sesuai Harapan | Berhasil |
|  | Pada form *testing* terdapat beberapa label yang digunakan untuk menampilkan perhitungan metode*,* jika label kosong kemudian *user* klik button Rata-Rata atau *button* konversi aksara Jawa ke huruf latin. | Muncul *message box* “Value is Nothing!” | Sesuai Harapan | Berhasil |
|  | Klik button “Browse” pilih gambar yang akan digunakan *testing* dan akan muncul jendela *open file dialog* | *Picture box image input* akan menampilkan citra yang dipilih. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button reset.* | Sistem menghapus *image* pada semua *picture box image*, menghidupkan kembali *button*  yang mati, menghapus nilai pada label. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button grayscale.* | *Picture box image grayscale* akan merubah citra menjadicitra keabuan (8 bit). | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button binarization.* | *Picture box image binarization* akan merubah citra menjadi 1 bit (hitam/putih). | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button inverse.* | *Picture box image binarization* akan membalik warna hitam menjadi putih dan warna putih menjadi hitam. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button thinning.* | *Picture box image thinning* akan menipiskan citra. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button Fit.* | *Picture box image fit* akan memotong citra.sesuai dengan batas citra dengan warna putih. | Sesuai harapan | Berhasil |
|  | Pada *group box preprocessing,* klik *button Resize.* | *Picture box image fit* akan merubah ukuran piksel citra menjadi 30x30 piksel dan menampilkan citra 30x30 pixel ke *picture box output image.* | Sesuai harapan | Berhasil |
|  | Pada *group box processing,* klik *button zona.* | Sistem akan memotong *picture box output image*  menjadi 9 bagian dan ditampilkan pada masing-masing *picture box* zona dan selanjutnya mampu menghitung nilai zona. | Sesuai harapan | Berhasil |
|  | Pada *group box processing,* klik *button rata-rata.* | Sistem akan menghitung rata-rata nilai masing-masing zona dan menghitung rata-rata baris 1 sampai 3 serta rata-rata baris 1 sampai 3. | Sesuai harapan | Berhasil |
|  | Pada bagian bawah, klik *button* konversi Aksara Jawa ke Huruf Latin. | Sistem akan menampilkan hasil konvesrsi ke *text box* yang ada dibawahnya. | Sesuai harapan | Berhasil |
|  | Dibagian bawah, klik *button help.* | Menampilkan informasi halaman bantuan. | Sesuai harapan | Berhasil |
|  | Dibagian bawah, klik *button table.* | Menampilkan informasi lengkap tentang nilai dalam *database.* | Sesuai harapan | Berhasil |
|  | Dibagian bawah, klik label “Aksara Jawa-Huruf Latin. | Menampilkan halaman utama . | Sesuai harapan | Berhasil |

* + 1. Pengujian Akurasi

Pengujian akurasi digunakan untuk menguji tujuan utama dari aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction*. Dalam pengujian ini diperlukan sebanyak 20 sampel karakter aksara Jawa. Berikut adalah hasil pengujian konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction.*

Tabel 6.3 Pengujian Konversi Aksara Jawa ke Huruf Latin

| **No.** | **Citra Input** | **Citra Output** | **Manual** | **Aplikasi** | **Hasil** |
| --- | --- | --- | --- | --- | --- |
|  |  |  | HA | YA | TIDAK SESUAI |
|  |  |  | NA | NA | SESUAI |
|  |  |  | CA | CA | SESUAI |
|  |  |  | RA | RA | SESUAI |
|  |  |  | KA | YA | TIDAK SESUAI |
|  |  |  | DA | DA | SESUAI |
|  |  |  | TA | TA | SESUAI |
|  |  |  | SA | SA | SESUAI |
|  |  |  | WA | CA | TIDAK SESUAI |
|  |  |  | LA | LA | SESUAI |
|  |  |  | PA | WA | TIDAK SESUAI |
|  |  |  | DHA | DHA | SESUAI |
|  |  |  | JA | JA | SESUAI |
|  |  |  | YA | YA | SESUAI |
|  |  |  | NYA | NYA | SESUAI |
|  |  |  | MA | MA | SESUAI |
|  |  |  | GA | GA | SESUAI |
|  |  |  | BA | BA | SESUAI |
|  |  |  | THA | THA | SESUAI |
|  |  |  | NGA | NYA | TIDAK  SESUAI |

Tabel 6.4Hasil Pengujian Tingkat Keakurasian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kategori | Jumlah Sample | Sesuai | Tidak Sesuai | Tingkat Akurasi |
| Aksara Jawa | 20 | 15 | 5 | **75%** |

Sumber : Pengujian

(6.1)

Dari hasil pengujian aksara Jawa sebanyak 20 sampel, dari 20 data tersebut teridentifikasi sesuai dengan total 15 dan teridentifikasi dengan total. Selanjutnya dari hasil pengujian di dapatkan tingkat akurasi 75% . Untuk perhitungan akrurasi   
dihitung dengan rumus pada persamaan 6.1.

Hasil pengujian di pengaruh karena beberapa faktor, untuk hasil pengujian pada tabel 6.3 poin 1 dan poin 5, pada huruf akasara Jawa ‘HA’ dan ‘KA’ teridentifikasi huruf ‘YA’, dikarenakan huruf ‘YA’ merupakan aksara Jawa dengan bentuk yang kompleks, sehingga memiliki nilai ekstrasi ciri yang dominan. Sedangkan pada tabel 6.3 pada poin 9, 11 dan 20 pada huruf aksara Jawa ‘WA’, ‘PA’, dan ‘NGA’ teridentifikasi dengan huruf yang memiliki bentuk yang hampir mirip. Hal ini terjadi dikarenakan bentuk tulisan tangan yang berbeda-beda sehingga nilai ekstrasinya memiliki kemiripan dengan aksara Jawa yang lainnya. Untuk itu, membutuhkan proses *preprocessing* yang baik.

## Pembahasan

Pengujian aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction* berjalan dengan baik. Sistem telah mampu memberikan keluaran huruf latin yang cukup sesuai jika dibandingkan pengamatan manusia secara manual.

Pada penelitian ini metode *diagonal based feature extraction* menggunakan 9 zona telah mampu mempresentasikan ekstrasi ciri. Hal ini ditunjukkan dengan hasil uji kemiripam menggunakan perhitungan *ecludian distance*mampu menunjukkan hasil yang cukup akurat jika dibandingkan pengamatan mata secara manual.

Aplikasi ini mampu mengenali tulisan tangan aksara Jawa dengan baik. Namun, proses ekstrasi ciri dipengaruhi oleh cara pengambilan foto dan bentuk tulisan tangan pada aksara Jawa itu sendiri sehingga pada kasus tertentu aplikasi ini memberikan hasil konversi aksara Jawa ke huruf latin yang salah dikarenakan sistem hanya menghitung nilai berdasarkan zona yang ada.

BAB VII. KESIMPULAN

## Kesimpulan

Dari semua uraian yang sudah dibahas dapat ditarik kesimpulan sebagai berikut:

1. Pengembangan aplikasi konversi aksara Jawa ke huruf latin pada citra digital menggunakan metode *diagonal based feature extraction* berhasil dilakukan.
2. Uji kemiripan menggunakan perhitungan *ecludian distance* mampu memberikan angka kemiripan antara citra *testing* dan citra *training*.
3. Pada proses pengujian aplikasi mampu menentukan hasil konversi aksara Jawa ke huruf latin dengan nilai akurasi 75%.

## Saran

Saran yang ditujukan untuk pengembangan penelitian ini adalah:

1. Diharapkan untuk penelitian selanjutnya ditambah lagivarian huruf aksara Jawa dengan menggunakan aksara pasangan, aksara paten dll.
2. Pada pengembangan aplikasi selanjutnya diharapkan dapat mengaplikasikan ke dalam versi android, sehingga dapat lebih memiliki nilai manfaat lebih bagi masyarakat pada umumnya.

DAFTAR PUSTAKA

[1] Meenu Alex and Smija Das, “An aprooach towards Malayalam Handwriting Recogniting Using Dissimilar Classifiers,” Global Colloquium in Recent Advancement and Effectual Researches in Engineering, Science and Technology (RAREST), 2016.

[2] Putra, Darma.2010.*Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi.

[3] Pramudana, Yustar. 2013. *Pengenalan Tulisan Tangan dengan Menggunakan Metode Diagonal Feature Extraction dan K-Nearest Neighbour*. Bandung: Telkom University.

[4] Darusuprapta, Prof. DR.,et al. 2002. *Pedoman Penulisan Aksara Jawa*. Yogyakarta: Yayasan Pustaka Nusantara.

[5] Jackson, J.E. (1991). A User‟s Guide to Principal Components (Wiley).

[6] Jollife, Ian. “Principal component analysis.” New York: SpringerVerlag., 2002.

[7] Mohebi, Ehsan, and Adil Bagirov. “A convolutional recursive modified Self Organizing Map for handwritten digits recognition.” Neural Networks 60 (2014): 104-118.

[8] Khatri, Sunil Kumar, Shivali Dutta, and Prashant Johri. “Recognizing images of handwritten digits using learning vector quantization artificial neural network.” Reliability, Infocom Technologie

[9] Akhtar, Muhammad Suhail, and Hammad A. Qureshi, “Handwritten digit recognition through wavelet decomposition and wavelet packet decomposition.” Digital Information Management (ICDIM), 2013 Eighth International Conference on. IEEE, 2013.

[10] Niu, Xiao-Xiao, and Ching Y. Suen. “A novel hybrid CNN–SVM classifier for recognizing handwritten digits.” Pattern Recognition 45.4 (2012): 1318-1325.

[11] Wu, J. (2012). “Advances in K-means clustering: a data mining thinking.” Springer Science & Business Media.

[12] Khan, M. E. (2015). “K-means clustering.”

[13] Juanying Xie, Shuai Jiang 2010, “A simple and fast algorithm for global K-means clustering”, 2010 Second International Workshop on Education Technology and Computer Science.

[14] Decong Yu, Lihong, Ma. “Digit Recognition Based on Multi-features”, IEEE Systems and Design Engineering Symposium, 2007.

[15] Ye Xu, Dezhan Liu: Clustering research of handwritten digit recognition by genetic algorithm. BIC-TA 2010: 291-297.

[16] Rafael C. Gonzalez, Richard E. woods and Steven L.Eddins, Digital Image Processing using MATLAB, Pearson Education, Dorling Kindersley, South Asia, 2004.

[17] Md. Al-Amin Bhuiyan. “Towards Face Recognition Using Eigenface”. International Journal of Advanced Computer Science and Applications (IJACSA). Vol. 7, No. 5, 2016.

[18] T. Zhang, C. Suen, A fast parallel algorithm for thinning digital patterns, Commun. ACM 27 (3) (1984) 236–239.

[19] J. Pradeep, E. Srinivasan and S.Himavathi, “Diagonal Based Feature Extraction for Handwritten Alphabets Recognition System Using Neural Network”, International Journal of Computer Science & Information Technology (IJCSIT). Vol. [20]

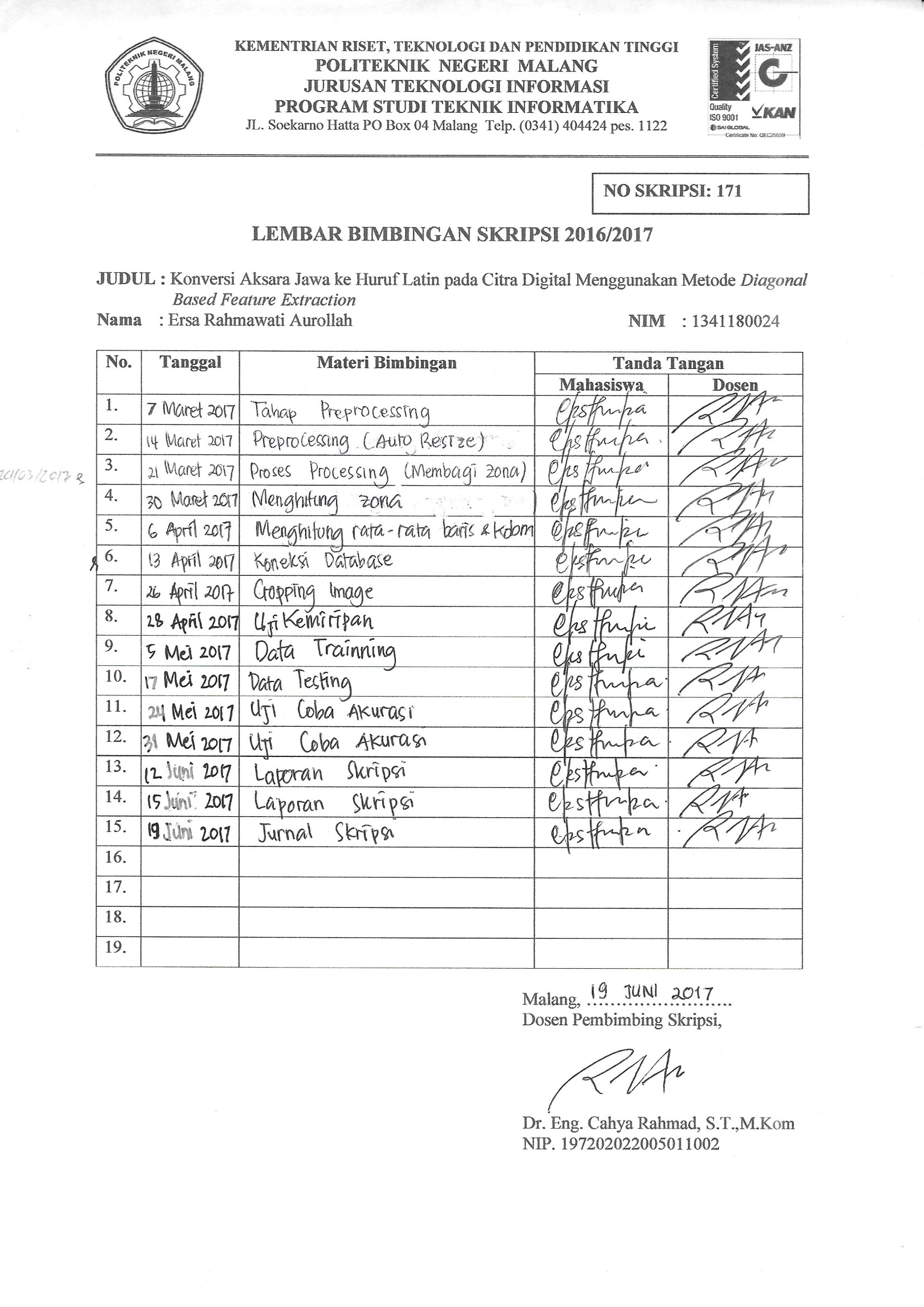
[20]Gawande, M.P. and Agrawal, D.G., Face recognition using PCA anddifferent distance classifiers. IOSR Journal of Electronics and Communication Engineering, 9(1), pp.1-5.

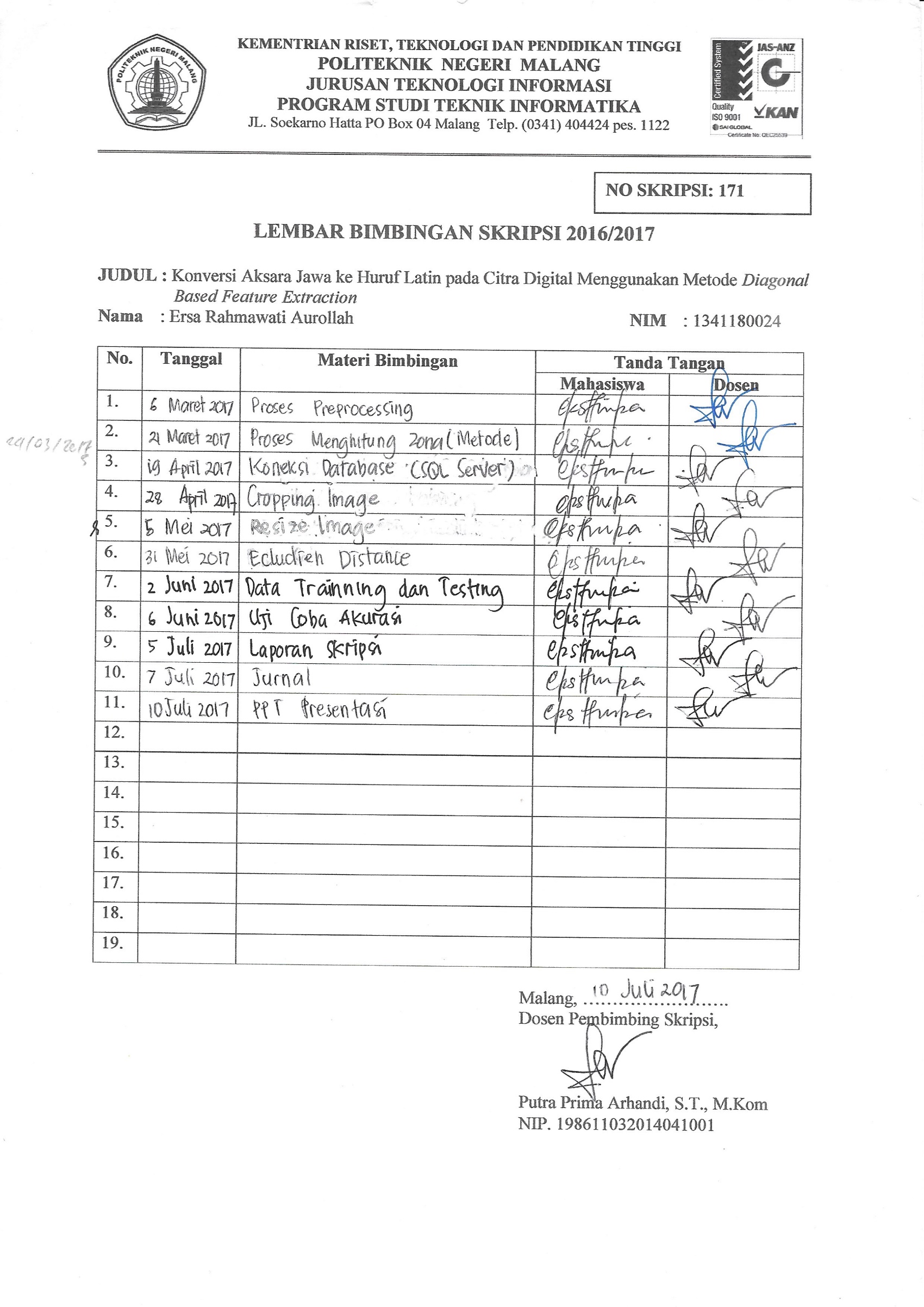
[21] Cha SH, Comprehensive Survey on Distance/Similarity Measures Between Probability Density Functions, City, 2007, 1:2, 1.3, no. 1, Feb. 2011.

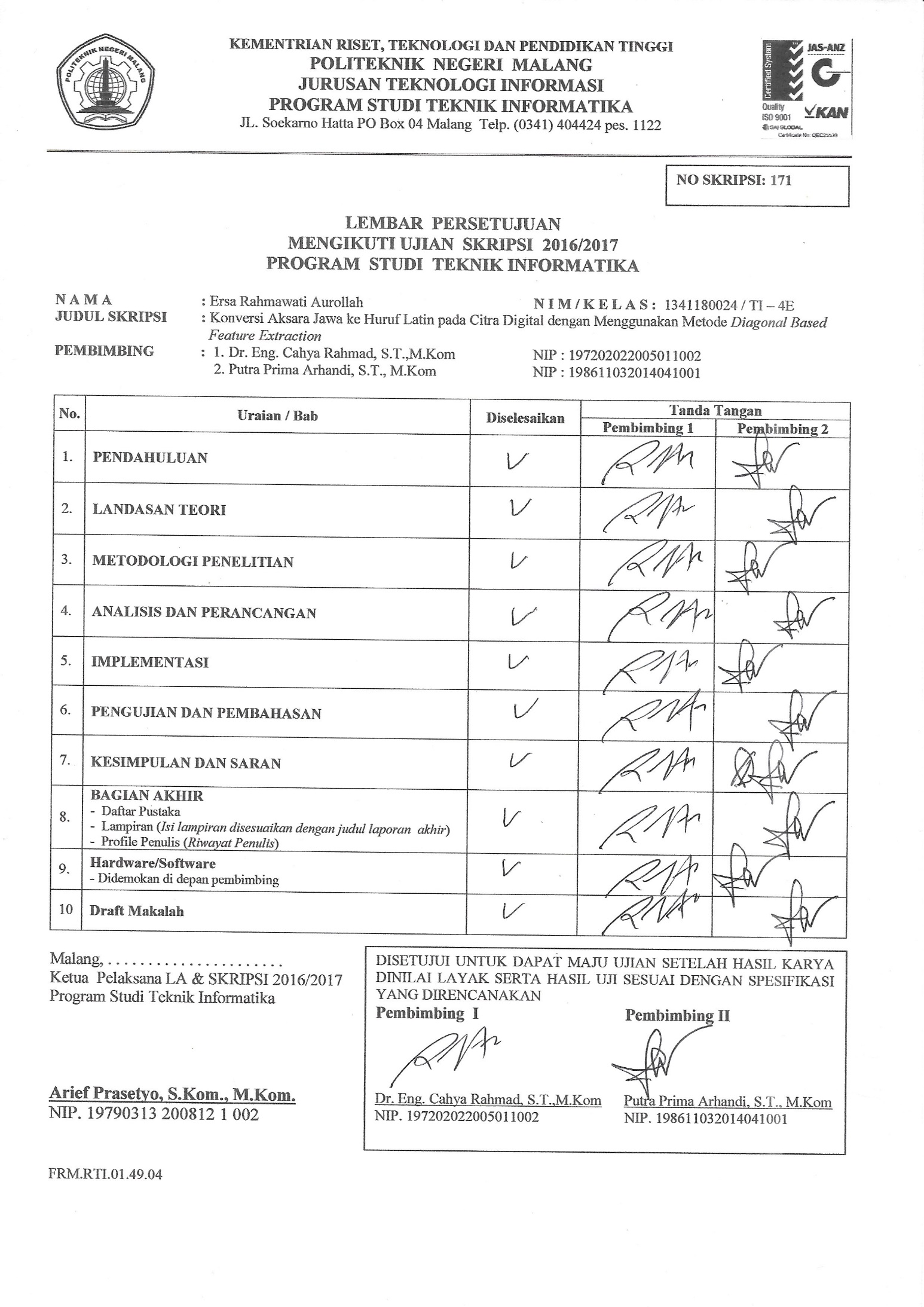
LAMPIRAN – LAMPIRAN

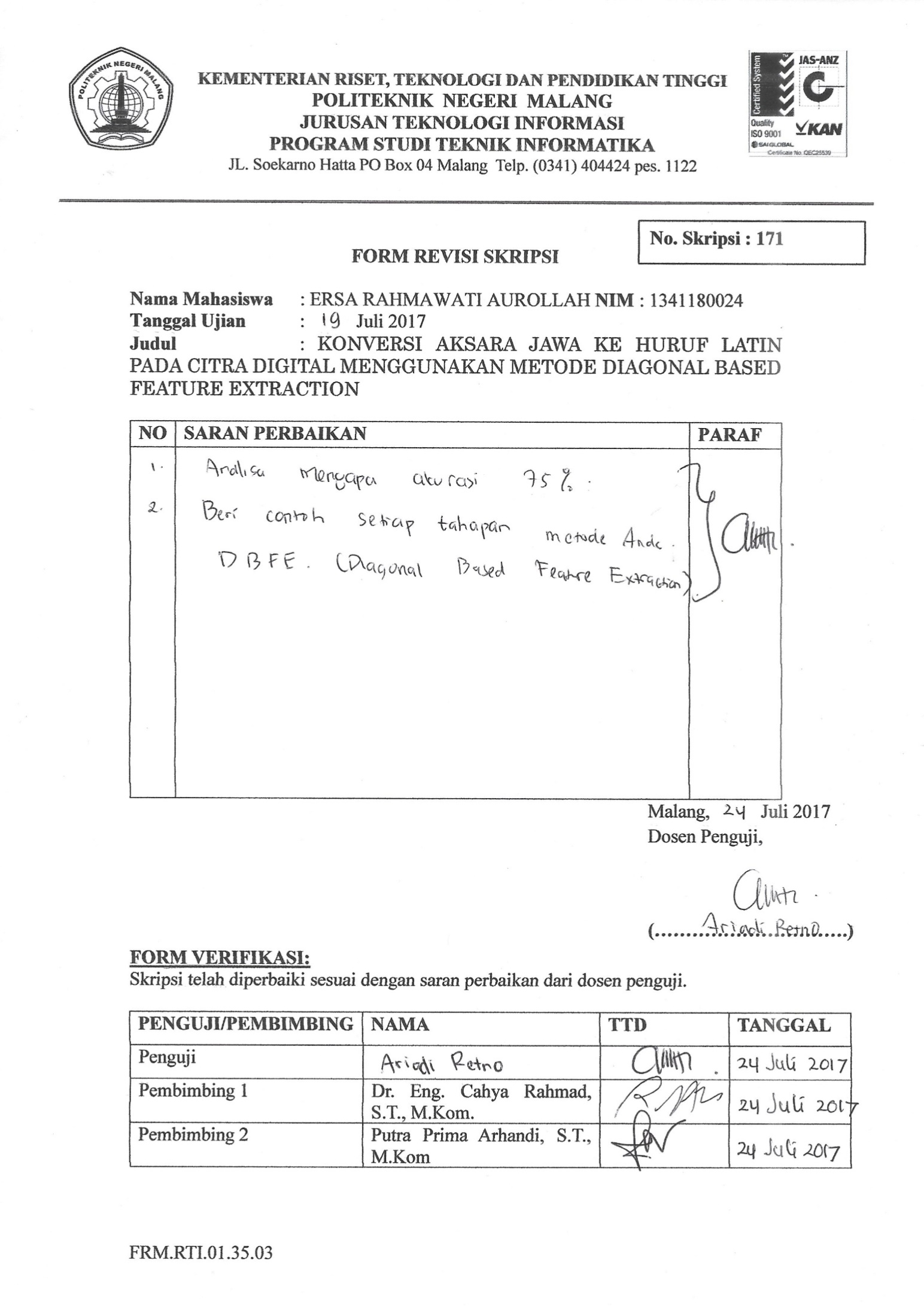
Lampiran 1. Surat Pengantar Observas

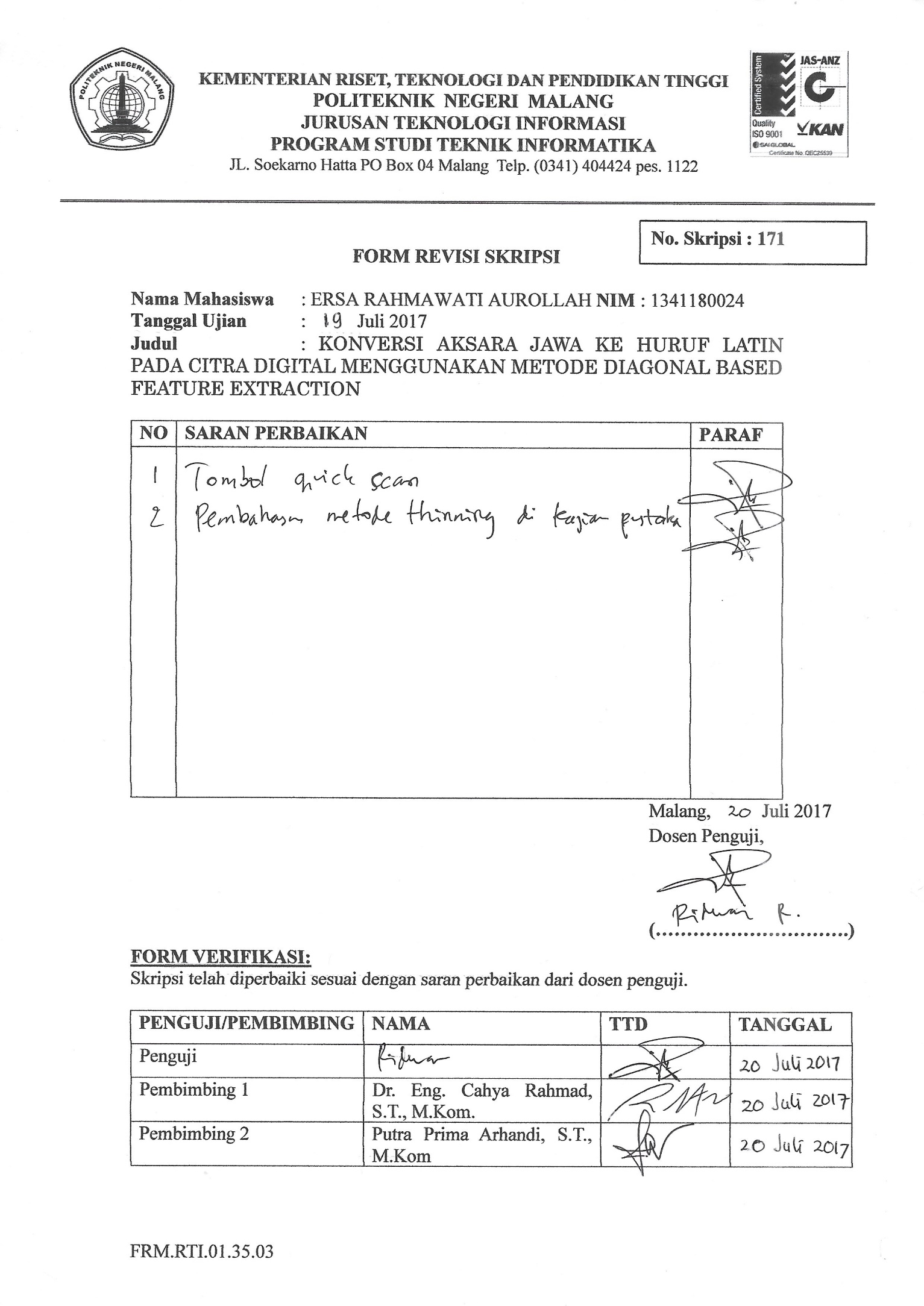
Lampiran 2. Surat Keterangan Uji Coba Sistem­

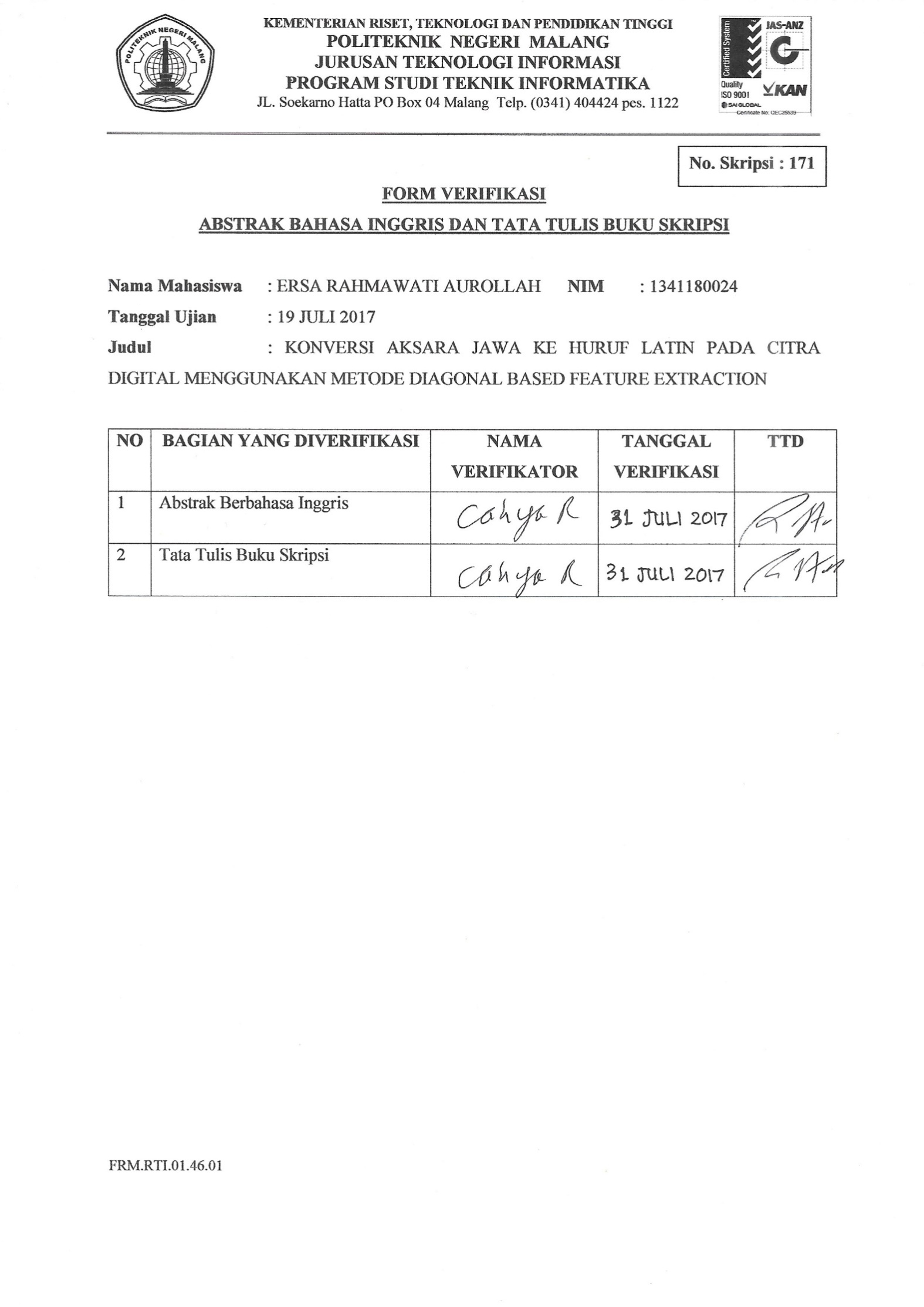
Lampiran 3. Lembar Bimbingan Skripsi (Pembimbing 1)

Lampiran 4. Lembar Bimbingan Skripsi (Pembimbing 2)

Lampiran 5. Lembar Persetujuan Mengikuti Ujian Skripsi

Lampiran 6. Form Revisi Skripsi (Penguji 1)

Lampiran 7. Form Revisi Skripsi (Penguji 2)

Lampiran 8. Form Verifikasi Abstrak Bahasa Inggris dan Tata Tulis Buku Skripsi

Lampiran 9. Kode Program *Browse*

Dim browse As OpenFileDialog = New OpenFileDialog()

browse.RestoreDirectory = True

If browse.ShowDialog = Windows.Forms.DialogResult.OK Then

inputImage.Image = Image.FromFile(browse.FileName)

inputImage.SizeMode = PictureBoxSizeMode.StretchImage

End If

filename = browse.FileName

Lampiran 10. Kode Program Reset

inputImage.Image = Nothing

resizeImage.Image = Nothing

GrayscaleImage.Image = Nothing

BinarizationImage.Image = Nothing

ProcessImage.Image = Nothing

thinningImage.Image = Nothing

Z1Image.Image = Nothing

Z2Image.Image = Nothing

Z3Image.Image = Nothing

Z4Image.Image = Nothing

Z5Image.Image = Nothing

Z6Image.Image = Nothing

Z7Image.Image = Nothing

Z8Image.Image = Nothing

Z9Image.Image = Nothing

LabelNZ1.Text = "NZ1"

LabelNZ2.Text = "NZ2"

LabelNZ3.Text = "NZ3"

LabelNZ4.Text = "NZ4"

LabelNZ5.Text = "NZ5"

LabelNZ6.Text = "NZ6"

LabelNZ7.Text = "NZ7"

LabelNZ8.Text = "NZ8"

LabelNZ9.Text = "NZ9"

LabelRZ1.Text = "RZ1"

LabelRZ2.Text = "RZ2"

LabelRZ3.Text = "RZ3"

LabelRZ4.Text = "RZ4"

LabelRZ5.Text = "RZ5"

LabelRZ6.Text = "RZ6"

LabelRZ7.Text = "RZ7"

LabelRZ8.Text = "RZ8"

LabelRZ9.Text = "RZ9"

LabelRB1.Text = "RB1"

LabelRB2.Text = "RB2"

LabelRB3.Text = "RB3"

LabelRK1.Text = "RK1"

LabelRK2.Text = "RK2"

LabelRK3.Text = "RK3"

TextBoxHuruf.Clear()

ButtonBrowse.Enabled = True

ButtonGrayscale.Enabled = True

ButtonBinerization.Enabled = True

ButtonInverse.Enabled = True

ButtonThinning.Enabled = True

ButtonFit.Enabled = True

ButtonResize.Enabled = True

ButtonZona.Enabled = True

ButtonRataRata.Enabled = True

ButtonSimpan.Enabled = True

Lampiran 11. Kode Program *Grayscale*

Dim img As New Bitmap(inputImage.Image)

GrayscaleImage.Image = img

Dim x, y As Integer

Dim r, g, b, average As Integer

For x = 0 To img.Width - 1

For y = 0 To img.Height - 1

r = img.GetPixel(x, y).R

g = img.GetPixel(x, y).G

b = img.GetPixel(x, y).B

average = CInt((r + g + b) / 3)

img.SetPixel(x, y, Color.FromArgb(average, average, average))

Next

Next

GrayscaleImage.SizeMode = PictureBoxSizeMode.StretchImage

GrayscaleImage.Refresh()

ButtonGrayscale.Enabled = False

Lampiran 12. Kode Program *Binnarization*

Dim img AsNewBitmap(GrayscaleImage.Image)

BinarizationImage.Image = img

Dim x, y AsInteger

Dim px, biner, coba AsInteger

For x = 0 To img.Width - 1

For y = 0 To img.Height - 1

px = img.GetPixel(x, y).R

If px < 128 Then

biner = 0

ElseIf px >= 128 Then

biner = 255

EndIf

img.SetPixel(x, y, Color.FromArgb(biner, biner, biner))

If px = 255 Then

coba += 1

EndIf

Next

Next

BinarizationImage.SizeMode = PictureBoxSizeMode.StretchImage

BinarizationImage.Refresh()

ButtonBinerization.Enabled = False

Lampiran 13. Kode Program Inverse

Dim img As New Bitmap(BinarizationImage.Image)

BinarizationImage.Image = img

Dim x, y As Integer ' x = baris , y = kolom

Dim px, inverse As Integer

For x = 0 To img.Width - 1

For y = 0 To img.Height - 1

px = img.GetPixel(x, y).R

If px = 0 Then

inverse = 255

ElseIf px = 255 Then

inverse = 0

End If

img.SetPixel(x, y, Color.FromArgb(inverse, inverse, inverse))

Next

Next

BinarizationImage.SizeMode = PictureBoxSizeMode.StretchImage

BinarizationImage.Refresh()

ButtonInverse.Enabled = False

Lampiran 14. Kode Program Thinning

Dim gambar1 AsNewBitmap(BinarizationImage.Image)

Dim X, Y AsInteger

Dim p2, p3, p4, p5, p6, p7, p8, p9 AsInteger

Dim kondisi1, kondisi2, kondisi3, kondisi4 AsInteger

Dim terpenuhi(gambar1.Width - 1, gambar1.Height - 1) AsInteger

For iterasi = 1 To 20

Dim count = 0

For X = 0 To gambar1.Width - 1

For Y = 0 To gambar1.Height - 1

'p2

If ((Y - 1) >= 0) Then

p2 = gambar1.GetPixel(X, Y - 1).R

Else

p2 = 0

EndIf

'p3

If (((X + 1) < gambar1.Width) And ((Y - 1) >= 0)) Then

p3 = gambar1.GetPixel(X + 1, Y - 1).R

Else

p3 = 0

EndIf

'p4

If ((X + 1) < gambar1.Width) Then

p4 = gambar1.GetPixel(X + 1, Y).R

Else

p4 = 0

EndIf

'p5

If (((X + 1) < gambar1.Width) And ((Y + 1) < gambar1.Height)) Then

p5 = gambar1.GetPixel(X + 1, Y + 1).R

Else

p5 = 0

EndIf

'p6

If ((Y + 1) < gambar1.Height) Then

p6 = gambar1.GetPixel(X, Y + 1).R

Else

p6 = 0

EndIf

'p7

If (((X - 1) >= 0) And ((Y + 1) < gambar1.Height)) Then

p7 = gambar1.GetPixel(X - 1, Y + 1).R

Else

p7 = 0

EndIf

'p8

If ((X - 1) >= 0) Then

p8 = gambar1.GetPixel(X - 1, Y).R

Else

p8 = 0

EndIf

'p9

If (((X - 1) >= 0) And ((Y - 1) >= 0)) Then

p9 = gambar1.GetPixel(X - 1, Y - 1).R

Else

p9 = 0

EndIf

kondisi1 = 0

kondisi2 = 0

kondisi3 = 0

kondisi4 = 0

If (p2 = 0) And (p3 = 255) Then

kondisi2 += 1

EndIf

If (p3 = 0) And (p4 = 255) Then

kondisi2 += 1

EndIf

If (p4 = 0) And (p5 = 255) Then

kondisi2 += 1

EndIf

If (p5 = 0) And (p6 = 255) Then

kondisi2 += 1

EndIf

If (p6 = 0) And (p7 = 255) Then

kondisi2 += 1

EndIf

If (p7 = 0) And (p8 = 255) Then

kondisi2 += 1

EndIf

If (p8 = 0) And (p9 = 255) Then

kondisi2 += 1

EndIf

If (p9 = 0) And (p2 = 255) Then

kondisi2 += 1

EndIf

kondisi1 = (p2 + p3 + p4 + p5 + p6 + p7 + p8 + p9) / 255

If iterasi Mod 2 <> 0 Then

kondisi3 = p2 \* p4 \* p6

kondisi4 = p4 \* p6 \* p8

Else

kondisi3 = p2 \* p4 \* p8

kondisi4 = p2 \* p6 \* p8

EndIf

terpenuhi(X, Y) = 0

If kondisi1 >= 2 And kondisi1 <= 6 Then

If kondisi2 = 1 Then

If kondisi3 = 0 Then'

If kondisi4 = 0 Then

terpenuhi(X, Y) = 1

EndIf

EndIf

EndIf

EndIf

Next

Next

For Y = 0 To gambar1.Height - 1

For X = 0 To gambar1.Width - 1

If terpenuhi(X, Y) = 1 Then

count += 1

gambar1.SetPixel(X, Y, Color.Black)

EndIf

Next

Next

If count = 0 Then

Exit For

EndIf

Next

thinningImage.Image = gambar1

thinningImage.SizeMode = PictureBoxSizeMode.StretchImage

ButtonThinning.Enabled = False

ButtonInverse.Enabled = False

Lampiran 15. Kode Program Fit

PrivateFunction CropBitmap(ByRef bmp AsBitmap, ByVal cropX AsInteger, ByVal cropY AsInteger, ByVal cropWidth AsInteger, ByVal cropHeight AsInteger) AsBitmap

Dim rect AsNewRectangle(cropX, cropY, cropWidth, cropHeight)

Dim cropped AsBitmap = bmp.Clone(rect, bmp.PixelFormat)

Return cropped

EndFunction

Dim img AsNewBitmap(thinningImage.Image)

Dim x, y, count AsInteger

Dim r, average, temp AsInteger

Dim meanw(img.Width - 1) AsInteger

Dim meanh(img.Height - 1) AsInteger

Dim id\_baris AsNewArrayList

Dim id\_kolom AsNewArrayList

count = 0

For x = 0 To img.Width - 1

temp = 0

For y = 0 To img.Height - 1

r = img.GetPixel(x, y).R

If (r = 255) Then

average = 1

Else

average = 0

EndIf

temp = temp + average

Next

If (temp > 0) Then

id\_kolom.Insert(count, x)

count = count + 1

EndIf

Next

count = 0

For x = 0 To img.Height - 1

temp = 0

For y = 0 To img.Width - 1

r = img.GetPixel(y, x).R

If (r = 255) Then

average = 1

Else

average = 0

EndIf

temp = temp + average

Next

If (temp > 0) Then

id\_baris.Insert(count, x)

count = count + 1

EndIf

Next

Console.WriteLine(id\_baris(0))

resizeImage.Image = CropBitmap(img, id\_kolom(0), id\_baris(0), id\_kolom.Count, id\_baris.Count)

ButtonFit.Enabled = False

Lampiran 16. Kode Program Resize

Dim resize As New Size(30, 30)

resizeImage.Image = New Bitmap(resizeImage.Image, resize)

Dim img As New Bitmap(resizeImage.Image)

resizeImage.Image = img

Dim x, y As Integer ' x = baris , y = kolom

Dim px, biner, coba As Integer 'r=red, g=Green , b=Blue

For x = 0 To img.Width - 1

For y = 0 To img.Height - 1

px = img.GetPixel(x, y).R

If px < 50 Then

biner = 0

ElseIf px >= 50 Then

biner = 255

End If

img.SetPixel(x, y, Color.FromArgb(biner, biner, biner))

If px = 255 Then

coba += 1

End If

Next

Next

ProcessImage.Image = resizeImage.Image

ProcessImage.SizeMode = PictureBoxSizeMode.StretchImage

ProcessImage.Refresh()

ButtonResize.Enabled = False

Catch

Lampiran 17. Kode Program Zona

Dim img As Bitmap = New Bitmap(ProcessImage.Image)

' make image indexed

Dim z(8) As Bitmap

For cc As Integer = 0 To 8

z(cc) = New Bitmap(10, 10, System.Drawing.Imaging.PixelFormat.Format32bppPArgb)

Next

Dim i As Integer = 0

Dim c, d, ee, f, xx, yy As Integer

c = 0

d = 9

For bb As Integer = 0 To 2

ee = 0

f = 9

For aa As Integer = 0 To 2

yy = 0

For y As Integer = c To d

xx = 0

For x = ee To f

Dim curpixcolor As Color = img.GetPixel(x, y)

Console.WriteLine(curpixcolor)

z(i).SetPixel(xx, yy, curpixcolor)

xx = xx + 1

Next

yy = yy + 1

Next

ee = ee + 10

f = f + 10

i = i + 1

Next

c = c + 10

d = d + 10

Next

Z1Image.Image = z(0)

Z2Image.Image = z(1)

Z3Image.Image = z(2)

Z4Image.Image = z(3)

Z5Image.Image = z(4)

Z6Image.Image = z(5)

Z7Image.Image = z(6)

Z8Image.Image = z(7)

Z9Image.Image = z(8)

'MENGHITUNG NILAI MASING - MASING ZONA'

Dim nzona1, nzona2, nzona3, nzona4, nzona5, nzona6, nzona7, nzona8, nzona9 As Integer

'ZONA 1'

Dim z1diag1 As Integer = z(0).GetPixel(0, 0).R

Dim z1diag2(1) As Integer

z1diag2(0) = z(0).GetPixel(1, 0).R

z1diag2(1) = z(0).GetPixel(0, 1).R

Dim z1diag3(2) As Integer

z1diag3(0) = z(0).GetPixel(2, 0).R

z1diag3(1) = z(0).GetPixel(1, 1).R

z1diag3(2) = z(0).GetPixel(0, 2).R

Dim z1diag4(3) As Integer

z1diag4(0) = z(0).GetPixel(3, 0).R

z1diag4(1) = z(0).GetPixel(2, 1).R

z1diag4(2) = z(0).GetPixel(1, 2).R

z1diag4(3) = z(0).GetPixel(0, 3).R

Dim z1diag5(4) As Integer

z1diag5(0) = z(0).GetPixel(4, 0).R

z1diag5(1) = z(0).GetPixel(3, 1).R

z1diag5(2) = z(0).GetPixel(2, 2).R

z1diag5(3) = z(0).GetPixel(1, 3).R

z1diag5(4) = z(0).GetPixel(0, 4).R

Dim z1diag6(5) As Integer

z1diag6(0) = z(0).GetPixel(5, 0).R

z1diag6(1) = z(0).GetPixel(4, 1).R

z1diag6(2) = z(0).GetPixel(3, 2).R

z1diag6(3) = z(0).GetPixel(2, 3).R

z1diag6(4) = z(0).GetPixel(1, 4).R

z1diag6(5) = z(0).GetPixel(0, 5).R

Dim z1diag7(6) As Integer

z1diag7(0) = z(0).GetPixel(6, 0).R

z1diag7(1) = z(0).GetPixel(5, 1).R

z1diag7(2) = z(0).GetPixel(4, 2).R

z1diag7(3) = z(0).GetPixel(3, 3).R

z1diag7(4) = z(0).GetPixel(2, 4).R

z1diag7(5) = z(0).GetPixel(1, 5).R

z1diag7(6) = z(0).GetPixel(0, 6).R

Dim z1diag8(7) As Integer

z1diag8(0) = z(0).GetPixel(7, 0).R

z1diag8(1) = z(0).GetPixel(6, 1).R

z1diag8(2) = z(0).GetPixel(5, 2).R

z1diag8(3) = z(0).GetPixel(4, 3).R

z1diag8(4) = z(0).GetPixel(3, 4).R

z1diag8(5) = z(0).GetPixel(2, 5).R

z1diag8(6) = z(0).GetPixel(1, 6).R

z1diag8(7) = z(0).GetPixel(0, 7).R

Dim z1diag9(8) As Integer

z1diag9(0) = z(0).GetPixel(8, 0).R

z1diag9(1) = z(0).GetPixel(7, 1).R

z1diag9(2) = z(0).GetPixel(6, 2).R

z1diag9(3) = z(0).GetPixel(5, 3).R

z1diag9(4) = z(0).GetPixel(4, 4).R

z1diag9(5) = z(0).GetPixel(3, 5).R

z1diag9(6) = z(0).GetPixel(2, 6).R

z1diag9(7) = z(0).GetPixel(1, 7).R

z1diag9(8) = z(0).GetPixel(0, 8).R

Dim z1diag10(9) As Integer

z1diag10(0) = z(0).GetPixel(9, 0).R

z1diag10(1) = z(0).GetPixel(8, 1).R

z1diag10(2) = z(0).GetPixel(7, 2).R

z1diag10(3) = z(0).GetPixel(6, 3).R

z1diag10(4) = z(0).GetPixel(5, 4).R

z1diag10(5) = z(0).GetPixel(4, 5).R

z1diag10(6) = z(0).GetPixel(3, 6).R

z1diag10(7) = z(0).GetPixel(2, 7).R

z1diag10(8) = z(0).GetPixel(1, 8).R

z1diag10(9) = z(0).GetPixel(0, 9).R

Dim z1diag11(8) As Integer

z1diag11(0) = z(0).GetPixel(9, 1).R

z1diag11(1) = z(0).GetPixel(8, 2).R

z1diag11(2) = z(0).GetPixel(7, 3).R

z1diag11(3) = z(0).GetPixel(6, 4).R

z1diag11(4) = z(0).GetPixel(5, 5).R

z1diag11(5) = z(0).GetPixel(4, 6).R

z1diag11(6) = z(0).GetPixel(3, 7).R

z1diag11(7) = z(0).GetPixel(2, 8).R

z1diag11(8) = z(0).GetPixel(1, 9).R

Dim z1diag12(7) As Integer

z1diag12(0) = z(0).GetPixel(9, 2).R

z1diag12(1) = z(0).GetPixel(8, 3).R

z1diag12(2) = z(0).GetPixel(7, 4).R

z1diag12(3) = z(0).GetPixel(6, 5).R

z1diag12(4) = z(0).GetPixel(5, 6).R

z1diag12(5) = z(0).GetPixel(4, 7).R

z1diag12(6) = z(0).GetPixel(3, 8).R

z1diag12(7) = z(0).GetPixel(2, 9).R

Dim z1diag13(6) As Integer

z1diag13(0) = z(0).GetPixel(9, 3).R

z1diag13(1) = z(0).GetPixel(8, 4).R

z1diag13(2) = z(0).GetPixel(7, 5).R

z1diag13(3) = z(0).GetPixel(6, 6).R

z1diag13(4) = z(0).GetPixel(5, 7).R

z1diag13(5) = z(0).GetPixel(4, 8).R

z1diag13(6) = z(0).GetPixel(3, 9).R

Dim z1diag14(5) As Integer

z1diag14(0) = z(0).GetPixel(9, 4).R

z1diag14(1) = z(0).GetPixel(8, 5).R

z1diag14(2) = z(0).GetPixel(7, 6).R

z1diag14(3) = z(0).GetPixel(6, 7).R

z1diag14(4) = z(0).GetPixel(5, 8).R

z1diag14(5) = z(0).GetPixel(4, 9).R

Dim z1diag15(4) As Integer

z1diag15(0) = z(0).GetPixel(9, 5).R

z1diag15(1) = z(0).GetPixel(8, 6).R

z1diag15(2) = z(0).GetPixel(7, 7).R

z1diag15(3) = z(0).GetPixel(6, 8).R

z1diag15(4) = z(0).GetPixel(5, 9).R

Dim z1diag16(3) As Integer

z1diag16(0) = z(0).GetPixel(9, 6).R

z1diag16(1) = z(0).GetPixel(8, 7).R

z1diag16(2) = z(0).GetPixel(7, 8).R

z1diag16(3) = z(0).GetPixel(6, 9).R

Dim z1diag17(2) As Integer

z1diag17(0) = z(0).GetPixel(9, 7).R

z1diag17(1) = z(0).GetPixel(8, 8).R

z1diag17(2) = z(0).GetPixel(7, 9).R

Dim z1diag18(1) As Integer

z1diag18(0) = z(0).GetPixel(9, 8).R

z1diag18(1) = z(0).GetPixel(8, 9).R

Dim z1diag19 As Integer = z(0).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z1diag1 = 255 Then

nzona1 += 1

End If

For mm As Integer = 0 To z1diag2.Length - 1

If z1diag2(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag3.Length - 1

If z1diag3(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag4.Length - 1

If z1diag4(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag5.Length - 1

If z1diag5(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag6.Length - 1

If z1diag6(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag7.Length - 1

If z1diag7(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag8.Length - 1

If z1diag8(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag9.Length - 1

If z1diag9(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag10.Length - 1

If z1diag10(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag11.Length - 1

If z1diag11(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag12.Length - 1

If z1diag12(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag13.Length - 1

If z1diag13(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag14.Length - 1

If z1diag14(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag15.Length - 1

If z1diag15(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag16.Length - 1

If z1diag16(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag17.Length - 1

If z1diag17(mm) = 255 Then

nzona1 += 1

End If

Next

For mm As Integer = 0 To z1diag18.Length - 1

If z1diag18(mm) = 255 Then

nzona1 += 1

End If

Next

If z1diag19 = 255 Then

nzona1 += 1

End If

LabelNZ1.Text = nzona1

'ZONA 2'

Dim z2diag1 As Integer = z(1).GetPixel(0, 0).R

Dim z2diag2(1) As Integer

z2diag2(0) = z(1).GetPixel(1, 0).R

z2diag2(1) = z(1).GetPixel(0, 1).R

Dim z2diag3(2) As Integer

z2diag3(0) = z(1).GetPixel(2, 0).R

z2diag3(1) = z(1).GetPixel(1, 1).R

z2diag3(2) = z(1).GetPixel(0, 2).R

Dim z2diag4(3) As Integer

z2diag4(0) = z(1).GetPixel(3, 0).R

z2diag4(1) = z(1).GetPixel(2, 1).R

z2diag4(2) = z(1).GetPixel(1, 2).R

z2diag4(3) = z(1).GetPixel(0, 3).R

Dim z2diag5(4) As Integer

z2diag5(0) = z(1).GetPixel(4, 0).R

z2diag5(1) = z(1).GetPixel(3, 1).R

z2diag5(2) = z(1).GetPixel(2, 2).R

z2diag5(3) = z(1).GetPixel(1, 3).R

z2diag5(4) = z(1).GetPixel(0, 4).R

Dim z2diag6(5) As Integer

z2diag6(0) = z(1).GetPixel(5, 0).R

z2diag6(1) = z(1).GetPixel(4, 1).R

z2diag6(2) = z(1).GetPixel(3, 2).R

z2diag6(3) = z(1).GetPixel(2, 3).R

z2diag6(4) = z(1).GetPixel(1, 4).R

z2diag6(5) = z(1).GetPixel(0, 5).R

Dim z2diag7(6) As Integer

z2diag7(0) = z(1).GetPixel(6, 0).R

z2diag7(1) = z(1).GetPixel(5, 1).R

z2diag7(2) = z(1).GetPixel(4, 2).R

z2diag7(3) = z(1).GetPixel(3, 3).R

z2diag7(4) = z(1).GetPixel(2, 4).R

z2diag7(5) = z(1).GetPixel(1, 5).R

z2diag7(6) = z(1).GetPixel(0, 6).R

Dim z2diag8(7) As Integer

z2diag8(0) = z(1).GetPixel(7, 0).R

z2diag8(1) = z(1).GetPixel(6, 1).R

z2diag8(2) = z(1).GetPixel(5, 2).R

z2diag8(3) = z(1).GetPixel(4, 3).R

z2diag8(4) = z(1).GetPixel(3, 4).R

z2diag8(5) = z(1).GetPixel(2, 5).R

z2diag8(6) = z(1).GetPixel(1, 6).R

z2diag8(7) = z(1).GetPixel(0, 7).R

Dim z2diag9(8) As Integer

z2diag9(0) = z(1).GetPixel(8, 0).R

z2diag9(1) = z(1).GetPixel(7, 1).R

z2diag9(2) = z(1).GetPixel(6, 2).R

z2diag9(3) = z(1).GetPixel(5, 3).R

z2diag9(4) = z(1).GetPixel(4, 4).R

z2diag9(5) = z(1).GetPixel(3, 5).R

z2diag9(6) = z(1).GetPixel(2, 6).R

z2diag9(7) = z(1).GetPixel(1, 7).R

z2diag9(8) = z(1).GetPixel(0, 8).R

Dim z2diag10(9) As Integer

z2diag10(0) = z(1).GetPixel(9, 0).R

z2diag10(1) = z(1).GetPixel(8, 1).R

z2diag10(2) = z(1).GetPixel(7, 2).R

z2diag10(3) = z(1).GetPixel(6, 3).R

z2diag10(4) = z(1).GetPixel(5, 4).R

z2diag10(5) = z(1).GetPixel(4, 5).R

z2diag10(6) = z(1).GetPixel(3, 6).R

z2diag10(7) = z(1).GetPixel(2, 7).R

z2diag10(8) = z(1).GetPixel(1, 8).R

z2diag10(9) = z(1).GetPixel(0, 9).R

Dim z2diag11(8) As Integer

z2diag11(0) = z(1).GetPixel(9, 1).R

z2diag11(1) = z(1).GetPixel(8, 2).R

z2diag11(2) = z(1).GetPixel(7, 3).R

z2diag11(3) = z(1).GetPixel(6, 4).R

z2diag11(4) = z(1).GetPixel(5, 5).R

z2diag11(5) = z(1).GetPixel(4, 6).R

z2diag11(6) = z(1).GetPixel(3, 7).R

z2diag11(7) = z(1).GetPixel(2, 8).R

z2diag11(8) = z(1).GetPixel(1, 9).R

Dim z2diag12(7) As Integer

z2diag12(0) = z(1).GetPixel(9, 2).R

z2diag12(1) = z(1).GetPixel(8, 3).R

z2diag12(2) = z(1).GetPixel(7, 4).R

z2diag12(3) = z(1).GetPixel(6, 5).R

z2diag12(4) = z(1).GetPixel(5, 6).R

z2diag12(5) = z(1).GetPixel(4, 7).R

z2diag12(6) = z(1).GetPixel(3, 8).R

z2diag12(7) = z(1).GetPixel(2, 9).R

Dim z2diag13(6) As Integer

z2diag13(0) = z(1).GetPixel(9, 3).R

z2diag13(1) = z(1).GetPixel(8, 4).R

z2diag13(2) = z(1).GetPixel(7, 5).R

z2diag13(3) = z(1).GetPixel(6, 6).R

z2diag13(4) = z(1).GetPixel(5, 7).R

z2diag13(5) = z(1).GetPixel(4, 8).R

z2diag13(6) = z(1).GetPixel(3, 9).R

Dim z2diag14(5) As Integer

z2diag14(0) = z(1).GetPixel(9, 4).R

z2diag14(1) = z(1).GetPixel(8, 5).R

z2diag14(2) = z(1).GetPixel(7, 6).R

z2diag14(3) = z(1).GetPixel(6, 7).R

z2diag14(4) = z(1).GetPixel(5, 8).R

z2diag14(5) = z(1).GetPixel(4, 9).R

Dim z2diag15(4) As Integer

z2diag15(0) = z(1).GetPixel(9, 5).R

z2diag15(1) = z(1).GetPixel(8, 6).R

z2diag15(2) = z(1).GetPixel(7, 7).R

z2diag15(3) = z(1).GetPixel(6, 8).R

z2diag15(4) = z(1).GetPixel(5, 9).R

Dim z2diag16(3) As Integer

z2diag16(0) = z(1).GetPixel(9, 6).R

z2diag16(1) = z(1).GetPixel(8, 7).R

z2diag16(2) = z(1).GetPixel(7, 8).R

z2diag16(3) = z(1).GetPixel(6, 9).R

Dim z2diag17(2) As Integer

z2diag17(0) = z(1).GetPixel(9, 7).R

z2diag17(1) = z(1).GetPixel(8, 8).R

z2diag17(2) = z(1).GetPixel(7, 9).R

Dim z2diag18(1) As Integer

z2diag18(0) = z(1).GetPixel(9, 8).R

z2diag18(1) = z(1).GetPixel(8, 9).R

Dim z2diag19 As Integer = z(1).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z2diag1 = 255 Then

nzona2 += 1

End If

For mm As Integer = 0 To z2diag2.Length - 1

If z2diag2(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag3.Length - 1

If z2diag3(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag4.Length - 1

If z2diag4(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag5.Length - 1

If z2diag5(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag6.Length - 1

If z2diag6(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag7.Length - 1

If z2diag7(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag8.Length - 1

If z2diag8(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag9.Length - 1

If z2diag9(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag10.Length - 1

If z2diag10(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag11.Length - 1

If z2diag11(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag12.Length - 1

If z2diag12(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag13.Length - 1

If z2diag13(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag14.Length - 1

If z2diag14(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag15.Length - 1

If z2diag15(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag16.Length - 1

If z2diag16(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag17.Length - 1

If z2diag17(mm) = 255 Then

nzona2 += 1

End If

Next

For mm As Integer = 0 To z2diag18.Length - 1

If z2diag18(mm) = 255 Then

nzona2 += 1

End If

Next

If z2diag19 = 255 Then

nzona2 += 1

End If

LabelNZ2.Text = nzona2

'ZONA 3'

Dim z3diag1 As Integer = z(2).GetPixel(0, 0).R

Dim z3diag2(1) As Integer

z3diag2(0) = z(2).GetPixel(1, 0).R

z3diag2(1) = z(2).GetPixel(0, 1).R

Dim z3diag3(2) As Integer

z3diag3(0) = z(2).GetPixel(2, 0).R

z3diag3(1) = z(2).GetPixel(1, 1).R

z3diag3(2) = z(2).GetPixel(0, 2).R

Dim z3diag4(3) As Integer

z3diag4(0) = z(2).GetPixel(3, 0).R

z3diag4(1) = z(2).GetPixel(2, 1).R

z3diag4(2) = z(2).GetPixel(1, 2).R

z3diag4(3) = z(2).GetPixel(0, 3).R

Dim z3diag5(4) As Integer

z3diag5(0) = z(2).GetPixel(4, 0).R

z3diag5(1) = z(2).GetPixel(3, 1).R

z3diag5(2) = z(2).GetPixel(2, 2).R

z3diag5(3) = z(2).GetPixel(1, 3).R

z3diag5(4) = z(2).GetPixel(0, 4).R

Dim z3diag6(5) As Integer

z3diag6(0) = z(2).GetPixel(5, 0).R

z3diag6(1) = z(2).GetPixel(4, 1).R

z3diag6(2) = z(2).GetPixel(3, 2).R

z3diag6(3) = z(2).GetPixel(2, 3).R

z3diag6(4) = z(2).GetPixel(1, 4).R

z3diag6(5) = z(2).GetPixel(0, 5).R

Dim z3diag7(6) As Integer

z3diag7(0) = z(2).GetPixel(6, 0).R

z3diag7(1) = z(2).GetPixel(5, 1).R

z3diag7(2) = z(2).GetPixel(4, 2).R

z3diag7(3) = z(2).GetPixel(3, 3).R

z3diag7(4) = z(2).GetPixel(2, 4).R

z3diag7(5) = z(2).GetPixel(1, 5).R

z3diag7(6) = z(2).GetPixel(0, 6).R

Dim z3diag8(7) As Integer

z3diag8(0) = z(2).GetPixel(7, 0).R

z3diag8(1) = z(2).GetPixel(6, 1).R

z3diag8(2) = z(2).GetPixel(5, 2).R

z3diag8(3) = z(2).GetPixel(4, 3).R

z3diag8(4) = z(2).GetPixel(3, 4).R

z3diag8(5) = z(2).GetPixel(2, 5).R

z3diag8(6) = z(2).GetPixel(1, 6).R

z3diag8(7) = z(2).GetPixel(0, 7).R

Dim z3diag9(8) As Integer

z3diag9(0) = z(2).GetPixel(8, 0).R

z3diag9(1) = z(2).GetPixel(7, 1).R

z3diag9(2) = z(2).GetPixel(6, 2).R

z3diag9(3) = z(2).GetPixel(5, 3).R

z3diag9(4) = z(2).GetPixel(4, 4).R

z3diag9(5) = z(2).GetPixel(3, 5).R

z3diag9(6) = z(2).GetPixel(2, 6).R

z3diag9(7) = z(2).GetPixel(1, 7).R

z3diag9(8) = z(2).GetPixel(0, 8).R

Dim z3diag10(9) As Integer

z3diag10(0) = z(2).GetPixel(9, 0).R

z3diag10(1) = z(2).GetPixel(8, 1).R

z3diag10(2) = z(2).GetPixel(7, 2).R

z3diag10(3) = z(2).GetPixel(6, 3).R

z3diag10(4) = z(2).GetPixel(5, 4).R

z3diag10(5) = z(2).GetPixel(4, 5).R

z3diag10(6) = z(2).GetPixel(3, 6).R

z3diag10(7) = z(2).GetPixel(2, 7).R

z3diag10(8) = z(2).GetPixel(1, 8).R

z3diag10(9) = z(2).GetPixel(0, 9).R

Dim z3diag11(8) As Integer

z3diag11(0) = z(2).GetPixel(9, 1).R

z3diag11(1) = z(2).GetPixel(8, 2).R

z3diag11(2) = z(2).GetPixel(7, 3).R

z3diag11(3) = z(2).GetPixel(6, 4).R

z3diag11(4) = z(2).GetPixel(5, 5).R

z3diag11(5) = z(2).GetPixel(4, 6).R

z3diag11(6) = z(2).GetPixel(3, 7).R

z3diag11(7) = z(2).GetPixel(2, 8).R

z3diag11(8) = z(2).GetPixel(1, 9).R

Dim z3diag12(7) As Integer

z3diag12(0) = z(2).GetPixel(9, 2).R

z3diag12(1) = z(2).GetPixel(8, 3).R

z3diag12(2) = z(2).GetPixel(7, 4).R

z3diag12(3) = z(2).GetPixel(6, 5).R

z3diag12(4) = z(2).GetPixel(5, 6).R

z3diag12(5) = z(2).GetPixel(4, 7).R

z3diag12(6) = z(2).GetPixel(3, 8).R

z3diag12(7) = z(2).GetPixel(2, 9).R

Dim z3diag13(6) As Integer

z3diag13(0) = z(2).GetPixel(9, 3).R

z3diag13(1) = z(2).GetPixel(8, 4).R

z3diag13(2) = z(2).GetPixel(7, 5).R

z3diag13(3) = z(2).GetPixel(6, 6).R

z3diag13(4) = z(2).GetPixel(5, 7).R

z3diag13(5) = z(2).GetPixel(4, 8).R

z3diag13(6) = z(2).GetPixel(3, 9).R

Dim z3diag14(5) As Integer

z3diag14(0) = z(2).GetPixel(9, 4).R

z3diag14(1) = z(2).GetPixel(8, 5).R

z3diag14(2) = z(2).GetPixel(7, 6).R

z3diag14(3) = z(2).GetPixel(6, 7).R

z3diag14(4) = z(2).GetPixel(5, 8).R

z3diag14(5) = z(2).GetPixel(4, 9).R

Dim z3diag15(4) As Integer

z3diag15(0) = z(2).GetPixel(9, 5).R

z3diag15(1) = z(2).GetPixel(8, 6).R

z3diag15(2) = z(2).GetPixel(7, 7).R

z3diag15(3) = z(2).GetPixel(6, 8).R

z3diag15(4) = z(2).GetPixel(5, 9).R

Dim z3diag16(3) As Integer

z3diag16(0) = z(2).GetPixel(9, 6).R

z3diag16(1) = z(2).GetPixel(8, 7).R

z3diag16(2) = z(2).GetPixel(7, 8).R

z3diag16(3) = z(2).GetPixel(6, 9).R

Dim z3diag17(2) As Integer

z3diag17(0) = z(2).GetPixel(9, 7).R

z3diag17(1) = z(2).GetPixel(8, 8).R

z3diag17(2) = z(2).GetPixel(7, 9).R

Dim z3diag18(1) As Integer

z3diag18(0) = z(2).GetPixel(9, 8).R

z3diag18(1) = z(2).GetPixel(8, 9).R

Dim z3diag19 As Integer = z(2).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z3diag1 = 255 Then

nzona3 += 1

End If

For mm As Integer = 0 To z3diag2.Length - 1

If z3diag3(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag3.Length - 1

If z3diag3(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag4.Length - 1

If z3diag4(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag5.Length - 1

If z3diag5(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag6.Length - 1

If z3diag6(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag7.Length - 1

If z3diag7(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag8.Length - 1

If z3diag8(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag9.Length - 1

If z3diag9(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag10.Length - 1

If z3diag10(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag11.Length - 1

If z3diag11(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag12.Length - 1

If z3diag12(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag13.Length - 1

If z3diag13(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag14.Length - 1

If z3diag14(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag15.Length - 1

If z3diag15(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag16.Length - 1

If z3diag16(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag17.Length - 1

If z3diag17(mm) = 255 Then

nzona3 += 1

End If

Next

For mm As Integer = 0 To z3diag18.Length - 1

If z3diag18(mm) = 255 Then

nzona3 += 1

End If

Next

If z3diag19 = 255 Then

nzona3 += 1

End If

LabelNZ3.Text = nzona3

'ZONA 4'

Dim z4diag1 As Integer = z(3).GetPixel(0, 0).R

Dim z4diag2(1) As Integer

z4diag2(0) = z(3).GetPixel(1, 0).R

z4diag2(1) = z(3).GetPixel(0, 1).R

Dim z4diag3(2) As Integer

z4diag3(0) = z(3).GetPixel(2, 0).R

z4diag3(1) = z(3).GetPixel(1, 1).R

z4diag3(2) = z(3).GetPixel(0, 2).R

Dim z4diag4(3) As Integer

z4diag4(0) = z(3).GetPixel(3, 0).R

z4diag4(1) = z(3).GetPixel(2, 1).R

z4diag4(2) = z(3).GetPixel(1, 2).R

z4diag4(3) = z(3).GetPixel(0, 3).R

Dim z4diag5(4) As Integer

z4diag5(0) = z(3).GetPixel(4, 0).R

z4diag5(1) = z(3).GetPixel(3, 1).R

z4diag5(2) = z(3).GetPixel(2, 2).R

z4diag5(3) = z(3).GetPixel(1, 3).R

z4diag5(4) = z(3).GetPixel(0, 4).R

Dim z4diag6(5) As Integer

z4diag6(0) = z(3).GetPixel(5, 0).R

z4diag6(1) = z(3).GetPixel(4, 1).R

z4diag6(2) = z(3).GetPixel(3, 2).R

z4diag6(3) = z(3).GetPixel(2, 3).R

z4diag6(4) = z(3).GetPixel(1, 4).R

z4diag6(5) = z(3).GetPixel(0, 5).R

Dim z4diag7(6) As Integer

z4diag7(0) = z(3).GetPixel(6, 0).R

z4diag7(1) = z(3).GetPixel(5, 1).R

z4diag7(2) = z(3).GetPixel(4, 2).R

z4diag7(3) = z(3).GetPixel(3, 3).R

z4diag7(4) = z(3).GetPixel(2, 4).R

z4diag7(5) = z(3).GetPixel(1, 5).R

z4diag7(6) = z(3).GetPixel(0, 6).R

Dim z4diag8(7) As Integer

z4diag8(0) = z(3).GetPixel(7, 0).R

z4diag8(1) = z(3).GetPixel(6, 1).R

z4diag8(2) = z(3).GetPixel(5, 2).R

z4diag8(3) = z(3).GetPixel(4, 3).R

z4diag8(4) = z(3).GetPixel(3, 4).R

z4diag8(5) = z(3).GetPixel(2, 5).R

z4diag8(6) = z(3).GetPixel(1, 6).R

z4diag8(7) = z(3).GetPixel(0, 7).R

Dim z4diag9(8) As Integer

z4diag9(0) = z(3).GetPixel(8, 0).R

z4diag9(1) = z(3).GetPixel(7, 1).R

z4diag9(2) = z(3).GetPixel(6, 2).R

z4diag9(3) = z(3).GetPixel(5, 3).R

z4diag9(4) = z(3).GetPixel(4, 4).R

z4diag9(5) = z(3).GetPixel(3, 5).R

z4diag9(6) = z(3).GetPixel(2, 6).R

z4diag9(7) = z(3).GetPixel(1, 7).R

z4diag9(8) = z(3).GetPixel(0, 8).R

Dim z4diag10(9) As Integer

z4diag10(0) = z(3).GetPixel(9, 0).R

z4diag10(1) = z(3).GetPixel(8, 1).R

z4diag10(2) = z(3).GetPixel(7, 2).R

z4diag10(3) = z(3).GetPixel(6, 3).R

z4diag10(4) = z(3).GetPixel(5, 4).R

z4diag10(5) = z(3).GetPixel(4, 5).R

z4diag10(6) = z(3).GetPixel(3, 6).R

z4diag10(7) = z(3).GetPixel(2, 7).R

z4diag10(8) = z(3).GetPixel(1, 8).R

z4diag10(9) = z(3).GetPixel(0, 9).R

Dim z4diag11(8) As Integer

z4diag11(0) = z(3).GetPixel(9, 1).R

z4diag11(1) = z(3).GetPixel(8, 2).R

z4diag11(2) = z(3).GetPixel(7, 3).R

z4diag11(3) = z(3).GetPixel(6, 4).R

z4diag11(4) = z(3).GetPixel(5, 5).R

z4diag11(5) = z(3).GetPixel(4, 6).R

z4diag11(6) = z(3).GetPixel(3, 7).R

z4diag11(7) = z(3).GetPixel(2, 8).R

z4diag11(8) = z(3).GetPixel(1, 9).R

Dim z4diag12(7) As Integer

z4diag12(0) = z(3).GetPixel(9, 2).R

z4diag12(1) = z(3).GetPixel(8, 3).R

z4diag12(2) = z(3).GetPixel(7, 4).R

z4diag12(3) = z(3).GetPixel(6, 5).R

z4diag12(4) = z(3).GetPixel(5, 6).R

z4diag12(5) = z(3).GetPixel(4, 7).R

z4diag12(6) = z(3).GetPixel(3, 8).R

z4diag12(7) = z(3).GetPixel(2, 9).R

Dim z4diag13(6) As Integer

z4diag13(0) = z(3).GetPixel(9, 3).R

z4diag13(1) = z(3).GetPixel(8, 4).R

z4diag13(2) = z(3).GetPixel(7, 5).R

z4diag13(3) = z(3).GetPixel(6, 6).R

z4diag13(4) = z(3).GetPixel(5, 7).R

z4diag13(5) = z(3).GetPixel(4, 8).R

z4diag13(6) = z(3).GetPixel(3, 9).R

Dim z4diag14(5) As Integer

z4diag14(0) = z(3).GetPixel(9, 4).R

z4diag14(1) = z(3).GetPixel(8, 5).R

z4diag14(2) = z(3).GetPixel(7, 6).R

z4diag14(3) = z(3).GetPixel(6, 7).R

z4diag14(4) = z(3).GetPixel(5, 8).R

z4diag14(5) = z(3).GetPixel(4, 9).R

Dim z4diag15(4) As Integer

z4diag15(0) = z(3).GetPixel(9, 5).R

z4diag15(1) = z(3).GetPixel(8, 6).R

z4diag15(2) = z(3).GetPixel(7, 7).R

z4diag15(3) = z(3).GetPixel(6, 8).R

z4diag15(4) = z(3).GetPixel(5, 9).R

Dim z4diag16(3) As Integer

z4diag16(0) = z(3).GetPixel(9, 6).R

z4diag16(1) = z(3).GetPixel(8, 7).R

z4diag16(2) = z(3).GetPixel(7, 8).R

z4diag16(3) = z(3).GetPixel(6, 9).R

Dim z4diag17(2) As Integer

z4diag17(0) = z(3).GetPixel(9, 7).R

z4diag17(1) = z(3).GetPixel(8, 8).R

z4diag17(2) = z(3).GetPixel(7, 9).R

Dim z4diag18(1) As Integer

z4diag18(0) = z(3).GetPixel(9, 8).R

z4diag18(1) = z(3).GetPixel(8, 9).R

Dim z4diag19 As Integer = z(3).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z4diag1 = 255 Then

nzona4 += 1

End If

For mm As Integer = 0 To z4diag2.Length - 1

If z4diag2(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag3.Length - 1

If z4diag3(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag4.Length - 1

If z4diag4(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag5.Length - 1

If z4diag5(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag6.Length - 1

If z4diag6(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag7.Length - 1

If z4diag7(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag8.Length - 1

If z4diag8(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag9.Length - 1

If z4diag9(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag10.Length - 1

If z4diag10(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag11.Length - 1

If z4diag11(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag12.Length - 1

If z4diag12(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag13.Length - 1

If z4diag13(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag14.Length - 1

If z4diag14(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag15.Length - 1

If z4diag15(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag16.Length - 1

If z4diag16(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag17.Length - 1

If z4diag17(mm) = 255 Then

nzona4 += 1

End If

Next

For mm As Integer = 0 To z4diag18.Length - 1

If z4diag18(mm) = 255 Then

nzona4 += 1

End If

Next

If z4diag19 = 255 Then

nzona4 += 1

End If

LabelNZ4.Text = nzona4

'ZONA 5'

Dim z5diag1 As Integer = z(4).GetPixel(0, 0).R

Dim z5diag2(1) As Integer

z5diag2(0) = z(4).GetPixel(1, 0).R

z5diag2(1) = z(4).GetPixel(0, 1).R

Dim z5diag3(2) As Integer

z5diag3(0) = z(4).GetPixel(2, 0).R

z5diag3(1) = z(4).GetPixel(1, 1).R

z5diag3(2) = z(4).GetPixel(0, 2).R

Dim z5diag4(3) As Integer

z5diag4(0) = z(4).GetPixel(3, 0).R

z5diag4(1) = z(4).GetPixel(2, 1).R

z5diag4(2) = z(4).GetPixel(1, 2).R

z5diag4(3) = z(4).GetPixel(0, 3).R

Dim z5diag5(4) As Integer

z5diag5(0) = z(4).GetPixel(4, 0).R

z5diag5(1) = z(4).GetPixel(3, 1).R

z5diag5(2) = z(4).GetPixel(2, 2).R

z5diag5(3) = z(4).GetPixel(1, 3).R

z5diag5(4) = z(4).GetPixel(0, 4).R

Dim z5diag6(5) As Integer

z5diag6(0) = z(4).GetPixel(5, 0).R

z5diag6(1) = z(4).GetPixel(4, 1).R

z5diag6(2) = z(4).GetPixel(3, 2).R

z5diag6(3) = z(4).GetPixel(2, 3).R

z5diag6(4) = z(4).GetPixel(1, 4).R

z5diag6(5) = z(4).GetPixel(0, 5).R

Dim z5diag7(6) As Integer

z5diag7(0) = z(4).GetPixel(6, 0).R

z5diag7(1) = z(4).GetPixel(5, 1).R

z5diag7(2) = z(4).GetPixel(4, 2).R

z5diag7(3) = z(4).GetPixel(3, 3).R

z5diag7(4) = z(4).GetPixel(2, 4).R

z5diag7(5) = z(4).GetPixel(1, 5).R

z5diag7(6) = z(4).GetPixel(0, 6).R

Dim z5diag8(7) As Integer

z5diag8(0) = z(4).GetPixel(7, 0).R

z5diag8(1) = z(4).GetPixel(6, 1).R

z5diag8(2) = z(4).GetPixel(5, 2).R

z5diag8(3) = z(4).GetPixel(4, 3).R

z5diag8(4) = z(4).GetPixel(3, 4).R

z5diag8(5) = z(4).GetPixel(2, 5).R

z5diag8(6) = z(4).GetPixel(1, 6).R

z5diag8(7) = z(4).GetPixel(0, 7).R

Dim z5diag9(8) As Integer

z5diag9(0) = z(4).GetPixel(8, 0).R

z5diag9(1) = z(4).GetPixel(7, 1).R

z5diag9(2) = z(4).GetPixel(6, 2).R

z5diag9(3) = z(4).GetPixel(5, 3).R

z5diag9(4) = z(4).GetPixel(4, 4).R

z5diag9(5) = z(4).GetPixel(3, 5).R

z5diag9(6) = z(4).GetPixel(2, 6).R

z5diag9(7) = z(4).GetPixel(1, 7).R

z5diag9(8) = z(4).GetPixel(0, 8).R

Dim z5diag10(9) As Integer

z5diag10(0) = z(4).GetPixel(9, 0).R

z5diag10(1) = z(4).GetPixel(8, 1).R

z5diag10(2) = z(4).GetPixel(7, 2).R

z5diag10(3) = z(4).GetPixel(6, 3).R

z5diag10(4) = z(4).GetPixel(5, 4).R

z5diag10(5) = z(4).GetPixel(4, 5).R

z5diag10(6) = z(4).GetPixel(3, 6).R

z5diag10(7) = z(4).GetPixel(2, 7).R

z5diag10(8) = z(4).GetPixel(1, 8).R

z5diag10(9) = z(4).GetPixel(0, 9).R

Dim z5diag11(8) As Integer

z5diag11(0) = z(4).GetPixel(9, 1).R

z5diag11(1) = z(4).GetPixel(8, 2).R

z5diag11(2) = z(4).GetPixel(7, 3).R

z5diag11(3) = z(4).GetPixel(6, 4).R

z5diag11(4) = z(4).GetPixel(5, 5).R

z5diag11(5) = z(4).GetPixel(4, 6).R

z5diag11(6) = z(4).GetPixel(3, 7).R

z5diag11(7) = z(4).GetPixel(2, 8).R

z5diag11(8) = z(4).GetPixel(1, 9).R

Dim z5diag12(7) As Integer

z5diag12(0) = z(4).GetPixel(9, 2).R

z5diag12(1) = z(4).GetPixel(8, 3).R

z5diag12(2) = z(4).GetPixel(7, 4).R

z5diag12(3) = z(4).GetPixel(6, 5).R

z5diag12(4) = z(4).GetPixel(5, 6).R

z5diag12(5) = z(4).GetPixel(4, 7).R

z5diag12(6) = z(4).GetPixel(3, 8).R

z5diag12(7) = z(4).GetPixel(2, 9).R

Dim z5diag13(6) As Integer

z5diag13(0) = z(4).GetPixel(9, 3).R

z5diag13(1) = z(4).GetPixel(8, 4).R

z5diag13(2) = z(4).GetPixel(7, 5).R

z5diag13(3) = z(4).GetPixel(6, 6).R

z5diag13(4) = z(4).GetPixel(5, 7).R

z5diag13(5) = z(4).GetPixel(4, 8).R

z5diag13(6) = z(4).GetPixel(3, 9).R

Dim z5diag14(5) As Integer

z5diag14(0) = z(4).GetPixel(9, 4).R

z5diag14(1) = z(4).GetPixel(8, 5).R

z5diag14(2) = z(4).GetPixel(7, 6).R

z5diag14(3) = z(4).GetPixel(6, 7).R

z5diag14(4) = z(4).GetPixel(5, 8).R

z5diag14(5) = z(4).GetPixel(4, 9).R

Dim z5diag15(4) As Integer

z5diag15(0) = z(4).GetPixel(9, 5).R

z5diag15(1) = z(4).GetPixel(8, 6).R

z5diag15(2) = z(4).GetPixel(7, 7).R

z5diag15(3) = z(4).GetPixel(6, 8).R

z5diag15(4) = z(4).GetPixel(5, 9).R

Dim z5diag16(3) As Integer

z5diag16(0) = z(4).GetPixel(9, 6).R

z5diag16(1) = z(4).GetPixel(8, 7).R

z5diag16(2) = z(4).GetPixel(7, 8).R

z5diag16(3) = z(4).GetPixel(6, 9).R

Dim z5diag17(2) As Integer

z5diag17(0) = z(4).GetPixel(9, 7).R

z5diag17(1) = z(4).GetPixel(8, 8).R

z5diag17(2) = z(4).GetPixel(7, 9).R

Dim z5diag18(1) As Integer

z5diag18(0) = z(4).GetPixel(9, 8).R

z5diag18(1) = z(4).GetPixel(8, 9).R

Dim z5diag19 As Integer = z(4).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z5diag1 = 255 Then

nzona5 += 1

End If

For mm As Integer = 0 To z5diag2.Length - 1

If z5diag2(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag3.Length - 1

If z5diag3(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag4.Length - 1

If z5diag4(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag5.Length - 1

If z5diag5(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag6.Length - 1

If z5diag6(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag7.Length - 1

If z5diag7(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag8.Length - 1

If z5diag8(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag9.Length - 1

If z5diag9(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag10.Length - 1

If z5diag10(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag11.Length - 1

If z5diag11(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag12.Length - 1

If z5diag12(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag13.Length - 1

If z5diag13(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag14.Length - 1

If z5diag14(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag15.Length - 1

If z5diag15(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag16.Length - 1

If z5diag16(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag17.Length - 1

If z5diag17(mm) = 255 Then

nzona5 += 1

End If

Next

For mm As Integer = 0 To z5diag18.Length - 1

If z5diag18(mm) = 255 Then

nzona5 += 1

End If

Next

If z5diag19 = 255 Then

nzona5 += 1

End If

LabelNZ5.Text = nzona5

'ZONA 6'

Dim z6diag1 As Integer = z(5).GetPixel(0, 0).R

Dim z6diag2(1) As Integer

z6diag2(0) = z(5).GetPixel(1, 0).R

z6diag2(1) = z(5).GetPixel(0, 1).R

Dim z6diag3(2) As Integer

z6diag3(0) = z(5).GetPixel(2, 0).R

z6diag3(1) = z(5).GetPixel(1, 1).R

z6diag3(2) = z(5).GetPixel(0, 2).R

Dim z6diag4(3) As Integer

z6diag4(0) = z(5).GetPixel(3, 0).R

z6diag4(1) = z(5).GetPixel(2, 1).R

z6diag4(2) = z(5).GetPixel(1, 2).R

z6diag4(3) = z(5).GetPixel(0, 3).R

Dim z6diag5(4) As Integer

z6diag5(0) = z(5).GetPixel(4, 0).R

z6diag5(1) = z(5).GetPixel(3, 1).R

z6diag5(2) = z(5).GetPixel(2, 2).R

z6diag5(3) = z(5).GetPixel(1, 3).R

z6diag5(4) = z(5).GetPixel(0, 4).R

Dim z6diag6(5) As Integer

z6diag6(0) = z(5).GetPixel(5, 0).R

z6diag6(1) = z(5).GetPixel(4, 1).R

z6diag6(2) = z(5).GetPixel(3, 2).R

z6diag6(3) = z(5).GetPixel(2, 3).R

z6diag6(4) = z(5).GetPixel(1, 4).R

z6diag6(5) = z(5).GetPixel(0, 5).R

Dim z6diag7(6) As Integer

z6diag7(0) = z(5).GetPixel(6, 0).R

z6diag7(1) = z(5).GetPixel(5, 1).R

z6diag7(2) = z(5).GetPixel(4, 2).R

z6diag7(3) = z(5).GetPixel(3, 3).R

z6diag7(4) = z(5).GetPixel(2, 4).R

z6diag7(5) = z(5).GetPixel(1, 5).R

z6diag7(6) = z(5).GetPixel(0, 6).R

Dim z6diag8(7) As Integer

z6diag8(0) = z(5).GetPixel(7, 0).R

z6diag8(1) = z(5).GetPixel(6, 1).R

z6diag8(2) = z(5).GetPixel(5, 2).R

z6diag8(3) = z(5).GetPixel(4, 3).R

z6diag8(4) = z(5).GetPixel(3, 4).R

z6diag8(5) = z(5).GetPixel(2, 5).R

z6diag8(6) = z(5).GetPixel(1, 6).R

z6diag8(7) = z(5).GetPixel(0, 7).R

Dim z6diag9(8) As Integer

z6diag9(0) = z(5).GetPixel(8, 0).R

z6diag9(1) = z(5).GetPixel(7, 1).R

z6diag9(2) = z(5).GetPixel(6, 2).R

z6diag9(3) = z(5).GetPixel(5, 3).R

z6diag9(4) = z(5).GetPixel(4, 4).R

z6diag9(5) = z(5).GetPixel(3, 5).R

z6diag9(6) = z(5).GetPixel(2, 6).R

z6diag9(7) = z(5).GetPixel(1, 7).R

z6diag9(8) = z(5).GetPixel(0, 8).R

Dim z6diag10(9) As Integer

z6diag10(0) = z(5).GetPixel(9, 0).R

z6diag10(1) = z(5).GetPixel(8, 1).R

z6diag10(2) = z(5).GetPixel(7, 2).R

z6diag10(3) = z(5).GetPixel(6, 3).R

z6diag10(4) = z(5).GetPixel(5, 4).R

z6diag10(5) = z(5).GetPixel(4, 5).R

z6diag10(6) = z(5).GetPixel(3, 6).R

z6diag10(7) = z(5).GetPixel(2, 7).R

z6diag10(8) = z(5).GetPixel(1, 8).R

z6diag10(9) = z(5).GetPixel(0, 9).R

Dim z6diag11(8) As Integer

z6diag11(0) = z(5).GetPixel(9, 1).R

z6diag11(1) = z(5).GetPixel(8, 2).R

z6diag11(2) = z(5).GetPixel(7, 3).R

z6diag11(3) = z(5).GetPixel(6, 4).R

z6diag11(4) = z(5).GetPixel(5, 5).R

z6diag11(5) = z(5).GetPixel(4, 6).R

z6diag11(6) = z(5).GetPixel(3, 7).R

z6diag11(7) = z(5).GetPixel(2, 8).R

z6diag11(8) = z(5).GetPixel(1, 9).R

Dim z6diag12(7) As Integer

z6diag12(0) = z(5).GetPixel(9, 2).R

z6diag12(1) = z(5).GetPixel(8, 3).R

z6diag12(2) = z(5).GetPixel(7, 4).R

z6diag12(3) = z(5).GetPixel(6, 5).R

z6diag12(4) = z(5).GetPixel(5, 6).R

z6diag12(5) = z(5).GetPixel(4, 7).R

z6diag12(6) = z(5).GetPixel(3, 8).R

z6diag12(7) = z(5).GetPixel(2, 9).R

Dim z6diag13(6) As Integer

z6diag13(0) = z(5).GetPixel(9, 3).R

z6diag13(1) = z(5).GetPixel(8, 4).R

z6diag13(2) = z(5).GetPixel(7, 5).R

z6diag13(3) = z(5).GetPixel(6, 6).R

z6diag13(4) = z(5).GetPixel(5, 7).R

z6diag13(5) = z(5).GetPixel(4, 8).R

z6diag13(6) = z(5).GetPixel(3, 9).R

Dim z6diag14(5) As Integer

z6diag14(0) = z(5).GetPixel(9, 4).R

z6diag14(1) = z(5).GetPixel(8, 5).R

z6diag14(2) = z(5).GetPixel(7, 6).R

z6diag14(3) = z(5).GetPixel(6, 7).R

z6diag14(4) = z(5).GetPixel(5, 8).R

z6diag14(5) = z(5).GetPixel(4, 9).R

Dim z6diag15(4) As Integer

z6diag15(0) = z(5).GetPixel(9, 5).R

z6diag15(1) = z(5).GetPixel(8, 6).R

z6diag15(2) = z(5).GetPixel(7, 7).R

z6diag15(3) = z(5).GetPixel(6, 8).R

z6diag15(4) = z(5).GetPixel(5, 9).R

Dim z6diag16(3) As Integer

z6diag16(0) = z(5).GetPixel(9, 6).R

z6diag16(1) = z(5).GetPixel(8, 7).R

z6diag16(2) = z(5).GetPixel(7, 8).R

z6diag16(3) = z(5).GetPixel(6, 9).R

Dim z6diag17(2) As Integer

z6diag17(0) = z(5).GetPixel(9, 7).R

z6diag17(1) = z(5).GetPixel(8, 8).R

z6diag17(2) = z(5).GetPixel(7, 9).R

Dim z6diag18(1) As Integer

z6diag18(0) = z(5).GetPixel(9, 8).R

z6diag18(1) = z(5).GetPixel(8, 9).R

Dim z6diag19 As Integer = z(5).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z6diag1 = 255 Then

nzona6 += 1

End If

If z6diag19 = 255 Then

nzona6 += 1

End If

For mm As Integer = 0 To z6diag2.Length - 1

If z6diag2(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag3.Length - 1

If z6diag3(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag4.Length - 1

If z6diag4(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag5.Length - 1

If z6diag5(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag6.Length - 1

If z6diag6(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag7.Length - 1

If z6diag7(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag8.Length - 1

If z6diag8(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag9.Length - 1

If z6diag9(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag10.Length - 1

If z6diag10(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag11.Length - 1

If z6diag11(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag12.Length - 1

If z6diag12(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag13.Length - 1

If z6diag13(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag14.Length - 1

If z6diag14(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag15.Length - 1

If z6diag15(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag16.Length - 1

If z6diag16(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag17.Length - 1

If z6diag17(mm) = 255 Then

nzona6 += 1

End If

Next

For mm As Integer = 0 To z6diag18.Length - 1

If z6diag18(mm) = 255 Then

nzona6 += 1

End If

Next

LabelNZ6.Text = nzona6

'ZONA 7'

Dim z7diag1 As Integer = z(6).GetPixel(0, 0).R

Dim z7diag2(1) As Integer

z7diag2(0) = z(6).GetPixel(1, 0).R

z7diag2(1) = z(6).GetPixel(0, 1).R

Dim z7diag3(2) As Integer

z7diag3(0) = z(6).GetPixel(2, 0).R

z7diag3(1) = z(6).GetPixel(1, 1).R

z7diag3(2) = z(6).GetPixel(0, 2).R

Dim z7diag4(3) As Integer

z7diag4(0) = z(6).GetPixel(3, 0).R

z7diag4(1) = z(6).GetPixel(2, 1).R

z7diag4(2) = z(6).GetPixel(1, 2).R

z7diag4(3) = z(6).GetPixel(0, 3).R

Dim z7diag5(4) As Integer

z7diag5(0) = z(6).GetPixel(4, 0).R

z7diag5(1) = z(6).GetPixel(3, 1).R

z7diag5(2) = z(6).GetPixel(2, 2).R

z7diag5(3) = z(6).GetPixel(1, 3).R

z7diag5(4) = z(6).GetPixel(0, 4).R

Dim z7diag6(5) As Integer

z7diag6(0) = z(6).GetPixel(5, 0).R

z7diag6(1) = z(6).GetPixel(4, 1).R

z7diag6(2) = z(6).GetPixel(3, 2).R

z7diag6(3) = z(6).GetPixel(2, 3).R

z7diag6(4) = z(6).GetPixel(1, 4).R

z7diag6(5) = z(6).GetPixel(0, 5).R

Dim z7diag7(6) As Integer

z7diag7(0) = z(6).GetPixel(6, 0).R

z7diag7(1) = z(6).GetPixel(5, 1).R

z7diag7(2) = z(6).GetPixel(4, 2).R

z7diag7(3) = z(6).GetPixel(3, 3).R

z7diag7(4) = z(6).GetPixel(2, 4).R

z7diag7(5) = z(6).GetPixel(1, 5).R

z7diag7(6) = z(6).GetPixel(0, 6).R

Dim z7diag8(7) As Integer

z7diag8(0) = z(6).GetPixel(7, 0).R

z7diag8(1) = z(6).GetPixel(6, 1).R

z7diag8(2) = z(6).GetPixel(5, 2).R

z7diag8(3) = z(6).GetPixel(4, 3).R

z7diag8(4) = z(6).GetPixel(3, 4).R

z7diag8(5) = z(6).GetPixel(2, 5).R

z7diag8(6) = z(6).GetPixel(1, 6).R

z7diag8(7) = z(6).GetPixel(0, 7).R

Dim z7diag9(8) As Integer

z7diag9(0) = z(6).GetPixel(8, 0).R

z7diag9(1) = z(6).GetPixel(7, 1).R

z7diag9(2) = z(6).GetPixel(6, 2).R

z7diag9(3) = z(6).GetPixel(5, 3).R

z7diag9(4) = z(6).GetPixel(4, 4).R

z7diag9(5) = z(6).GetPixel(3, 5).R

z7diag9(6) = z(6).GetPixel(2, 6).R

z7diag9(7) = z(6).GetPixel(1, 7).R

z7diag9(8) = z(6).GetPixel(0, 8).R

Dim z7diag10(9) As Integer

z7diag10(0) = z(6).GetPixel(9, 0).R

z7diag10(1) = z(6).GetPixel(8, 1).R

z7diag10(2) = z(6).GetPixel(7, 2).R

z7diag10(3) = z(6).GetPixel(6, 3).R

z7diag10(4) = z(6).GetPixel(5, 4).R

z7diag10(5) = z(6).GetPixel(4, 5).R

z7diag10(6) = z(6).GetPixel(3, 6).R

z7diag10(7) = z(6).GetPixel(2, 7).R

z7diag10(8) = z(6).GetPixel(1, 8).R

z7diag10(9) = z(6).GetPixel(0, 9).R

Dim z7diag11(8) As Integer

z7diag11(0) = z(6).GetPixel(9, 1).R

z7diag11(1) = z(6).GetPixel(8, 2).R

z7diag11(2) = z(6).GetPixel(7, 3).R

z7diag11(3) = z(6).GetPixel(6, 4).R

z7diag11(4) = z(6).GetPixel(5, 5).R

z7diag11(5) = z(6).GetPixel(4, 6).R

z7diag11(6) = z(6).GetPixel(3, 7).R

z7diag11(7) = z(6).GetPixel(2, 8).R

z7diag11(8) = z(6).GetPixel(1, 9).R

Dim z7diag12(7) As Integer

z7diag12(0) = z(6).GetPixel(9, 2).R

z7diag12(1) = z(6).GetPixel(8, 3).R

z7diag12(2) = z(6).GetPixel(7, 4).R

z7diag12(3) = z(6).GetPixel(6, 5).R

z7diag12(4) = z(6).GetPixel(5, 6).R

z7diag12(5) = z(6).GetPixel(4, 7).R

z7diag12(6) = z(6).GetPixel(3, 8).R

z7diag12(7) = z(6).GetPixel(2, 9).R

Dim z7diag13(6) As Integer

z7diag13(0) = z(6).GetPixel(9, 3).R

z7diag13(1) = z(6).GetPixel(8, 4).R

z7diag13(2) = z(6).GetPixel(7, 5).R

z7diag13(3) = z(6).GetPixel(6, 6).R

z7diag13(4) = z(6).GetPixel(5, 7).R

z7diag13(5) = z(6).GetPixel(4, 8).R

z7diag13(6) = z(6).GetPixel(3, 9).R

Dim z7diag14(5) As Integer

z7diag14(0) = z(6).GetPixel(9, 4).R

z7diag14(1) = z(6).GetPixel(8, 5).R

z7diag14(2) = z(6).GetPixel(7, 6).R

z7diag14(3) = z(6).GetPixel(6, 7).R

z7diag14(4) = z(6).GetPixel(5, 8).R

z7diag14(5) = z(6).GetPixel(4, 9).R

Dim z7diag15(4) As Integer

z7diag15(0) = z(6).GetPixel(9, 5).R

z7diag15(1) = z(6).GetPixel(8, 6).R

z7diag15(2) = z(6).GetPixel(7, 7).R

z7diag15(3) = z(6).GetPixel(6, 8).R

z7diag15(4) = z(6).GetPixel(5, 9).R

Dim z7diag16(3) As Integer

z7diag16(0) = z(6).GetPixel(9, 6).R

z7diag16(1) = z(6).GetPixel(8, 7).R

z7diag16(2) = z(6).GetPixel(7, 8).R

z7diag16(3) = z(6).GetPixel(6, 9).R

Dim z7diag17(2) As Integer

z1diag17(0) = z(6).GetPixel(9, 7).R

z1diag17(1) = z(6).GetPixel(8, 8).R

z1diag17(2) = z(6).GetPixel(7, 9).R

Dim z7diag18(1) As Integer

z7diag18(0) = z(6).GetPixel(9, 8).R

z7diag18(1) = z(6).GetPixel(8, 9).R

Dim z7diag19 As Integer = z(6).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z7diag1 = 255 Then

nzona7 += 1

End If

For mm As Integer = 0 To z7diag2.Length - 1

If z7diag2(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag3.Length - 1

If z7diag3(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag4.Length - 1

If z7diag4(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag5.Length - 1

If z7diag5(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag6.Length - 1

If z7diag6(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag7.Length - 1

If z7diag7(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag8.Length - 1

If z7diag8(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag9.Length - 1

If z7diag9(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag10.Length - 1

If z7diag10(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag11.Length - 1

If z7diag11(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag12.Length - 1

If z7diag12(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag13.Length - 1

If z7diag13(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag14.Length - 1

If z7diag14(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag15.Length - 1

If z7diag15(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag16.Length - 1

If z7diag16(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag17.Length - 1

If z7diag17(mm) = 255 Then

nzona7 += 1

End If

Next

For mm As Integer = 0 To z7diag18.Length - 1

If z7diag18(mm) = 255 Then

nzona7 += 1

End If

Next

If z7diag19 = 255 Then

nzona7 += 1

End If

LabelNZ7.Text = nzona7

'ZONA 8'

Dim z8diag1 As Integer = z(7).GetPixel(0, 0).R

Dim z8diag2(1) As Integer

z8diag2(0) = z(7).GetPixel(1, 0).R

z8diag2(1) = z(7).GetPixel(0, 1).R

Dim z8diag3(2) As Integer

z8diag3(0) = z(7).GetPixel(2, 0).R

z8diag3(1) = z(7).GetPixel(1, 1).R

z8diag3(2) = z(7).GetPixel(0, 2).R

Dim z8diag4(3) As Integer

z8diag4(0) = z(7).GetPixel(3, 0).R

z8diag4(1) = z(7).GetPixel(2, 1).R

z8diag4(2) = z(7).GetPixel(1, 2).R

z8diag4(3) = z(7).GetPixel(0, 3).R

Dim z8diag5(4) As Integer

z8diag5(0) = z(7).GetPixel(4, 0).R

z8diag5(1) = z(7).GetPixel(3, 1).R

z8diag5(2) = z(7).GetPixel(2, 2).R

z8diag5(3) = z(7).GetPixel(1, 3).R

z8diag5(4) = z(7).GetPixel(0, 4).R

Dim z8diag6(5) As Integer

z8diag6(0) = z(7).GetPixel(5, 0).R

z8diag6(1) = z(7).GetPixel(4, 1).R

z8diag6(2) = z(7).GetPixel(3, 2).R

z8diag6(3) = z(7).GetPixel(2, 3).R

z8diag6(4) = z(7).GetPixel(1, 4).R

z8diag6(5) = z(7).GetPixel(0, 5).R

Dim z8diag7(6) As Integer

z8diag7(0) = z(7).GetPixel(6, 0).R

z8diag7(1) = z(7).GetPixel(5, 1).R

z8diag7(2) = z(7).GetPixel(4, 2).R

z8diag7(3) = z(7).GetPixel(3, 3).R

z8diag7(4) = z(7).GetPixel(2, 4).R

z8diag7(5) = z(7).GetPixel(1, 5).R

z8diag7(6) = z(7).GetPixel(0, 6).R

Dim z8diag8(7) As Integer

z8diag8(0) = z(7).GetPixel(7, 0).R

z8diag8(1) = z(7).GetPixel(6, 1).R

z8diag8(2) = z(7).GetPixel(5, 2).R

z8diag8(3) = z(7).GetPixel(4, 3).R

z8diag8(4) = z(7).GetPixel(3, 4).R

z8diag8(5) = z(7).GetPixel(2, 5).R

z8diag8(6) = z(7).GetPixel(1, 6).R

z8diag8(7) = z(7).GetPixel(0, 7).R

Dim z8diag9(8) As Integer

z8diag9(0) = z(7).GetPixel(8, 0).R

z8diag9(1) = z(7).GetPixel(7, 1).R

z8diag9(2) = z(7).GetPixel(6, 2).R

z8diag9(3) = z(7).GetPixel(5, 3).R

z8diag9(4) = z(7).GetPixel(4, 4).R

z8diag9(5) = z(7).GetPixel(3, 5).R

z8diag9(6) = z(7).GetPixel(2, 6).R

z8diag9(7) = z(7).GetPixel(1, 7).R

z8diag9(8) = z(7).GetPixel(0, 8).R

Dim z8diag10(9) As Integer

z8diag10(0) = z(7).GetPixel(9, 0).R

z8diag10(1) = z(7).GetPixel(8, 1).R

z8diag10(2) = z(7).GetPixel(7, 2).R

z8diag10(3) = z(7).GetPixel(6, 3).R

z8diag10(4) = z(7).GetPixel(5, 4).R

z8diag10(5) = z(7).GetPixel(4, 5).R

z8diag10(6) = z(7).GetPixel(3, 6).R

z8diag10(7) = z(7).GetPixel(2, 7).R

z8diag10(8) = z(7).GetPixel(1, 8).R

z8diag10(9) = z(7).GetPixel(0, 9).R

Dim z8diag11(8) As Integer

z8diag11(0) = z(7).GetPixel(9, 1).R

z8diag11(1) = z(7).GetPixel(8, 2).R

z8diag11(2) = z(7).GetPixel(7, 3).R

z8diag11(3) = z(7).GetPixel(6, 4).R

z8diag11(4) = z(7).GetPixel(5, 5).R

z8diag11(5) = z(7).GetPixel(4, 6).R

z8diag11(6) = z(7).GetPixel(3, 7).R

z8diag11(7) = z(7).GetPixel(2, 8).R

z8diag11(8) = z(7).GetPixel(1, 9).R

Dim z8diag12(7) As Integer

z8diag12(0) = z(7).GetPixel(9, 2).R

z8diag12(1) = z(7).GetPixel(8, 3).R

z8diag12(2) = z(7).GetPixel(7, 4).R

z8diag12(3) = z(7).GetPixel(6, 5).R

z8diag12(4) = z(7).GetPixel(5, 6).R

z8diag12(5) = z(7).GetPixel(4, 7).R

z8diag12(6) = z(7).GetPixel(3, 8).R

z8diag12(7) = z(7).GetPixel(2, 9).R

Dim z8diag13(6) As Integer

z8diag13(0) = z(7).GetPixel(9, 3).R

z8diag13(1) = z(7).GetPixel(8, 4).R

z8diag13(2) = z(7).GetPixel(7, 5).R

z8diag13(3) = z(7).GetPixel(6, 6).R

z8diag13(4) = z(7).GetPixel(5, 7).R

z8diag13(5) = z(7).GetPixel(4, 8).R

z8diag13(6) = z(7).GetPixel(3, 9).R

Dim z8diag14(5) As Integer

z8diag14(0) = z(7).GetPixel(9, 4).R

z8diag14(1) = z(7).GetPixel(8, 5).R

z8diag14(2) = z(7).GetPixel(7, 6).R

z8diag14(3) = z(7).GetPixel(6, 7).R

z8diag14(4) = z(7).GetPixel(5, 8).R

z8diag14(5) = z(7).GetPixel(4, 9).R

Dim z8diag15(4) As Integer

z8diag15(0) = z(7).GetPixel(9, 5).R

z8diag15(1) = z(7).GetPixel(8, 6).R

z8diag15(2) = z(7).GetPixel(7, 7).R

z8diag15(3) = z(7).GetPixel(6, 8).R

z8diag15(4) = z(7).GetPixel(5, 9).R

Dim z8diag16(3) As Integer

z8diag16(0) = z(7).GetPixel(9, 6).R

z8diag16(1) = z(7).GetPixel(8, 7).R

z8diag16(2) = z(7).GetPixel(7, 8).R

z8diag16(3) = z(7).GetPixel(6, 9).R

Dim z8diag17(2) As Integer

z8diag17(0) = z(7).GetPixel(9, 7).R

z8diag17(1) = z(7).GetPixel(8, 8).R

z8diag17(2) = z(7).GetPixel(7, 9).R

Dim z8diag18(1) As Integer

z8diag18(0) = z(7).GetPixel(9, 8).R

z8diag18(1) = z(7).GetPixel(8, 9).R

Dim z8diag19 As Integer = z(7).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z8diag1 = 255 Then

nzona8 += 1

End If

For mm As Integer = 0 To z8diag2.Length - 1

If z8diag2(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag3.Length - 1

If z8diag3(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag4.Length - 1

If z8diag4(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag5.Length - 1

If z8diag5(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag6.Length - 1

If z8diag6(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag7.Length - 1

If z8diag7(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag8.Length - 1

If z8diag8(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag9.Length - 1

If z8diag9(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag10.Length - 1

If z8diag10(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag11.Length - 1

If z8diag11(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag12.Length - 1

If z8diag12(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag13.Length - 1

If z8diag13(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag14.Length - 1

If z8diag14(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag15.Length - 1

If z8diag15(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag16.Length - 1

If z8diag16(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag17.Length - 1

If z8diag17(mm) = 255 Then

nzona8 += 1

End If

Next

For mm As Integer = 0 To z8diag18.Length - 1

If z8diag18(mm) = 255 Then

nzona8 += 1

End If

Next

If z8diag19 = 255 Then

nzona8 += 1

End If

LabelNZ8.Text = nzona8

'ZONA 9'

Dim z9diag1 As Integer = z(8).GetPixel(0, 0).R

Dim z9diag2(1) As Integer

z9diag2(0) = z(8).GetPixel(0, 1).R

z9diag2(1) = z(8).GetPixel(1, 0).R

Dim z9diag3(2) As Integer

z9diag3(0) = z(8).GetPixel(2, 0).R

z9diag3(1) = z(8).GetPixel(1, 1).R

z9diag3(

2) = z(8).GetPixel(0, 2).R

Dim z9diag4(3) As Integer

z9diag4(0) = z(8).GetPixel(3, 0).R

z9diag4(1) = z(8).GetPixel(2, 1).R

z9diag4(2) = z(8).GetPixel(1, 2).R

z9diag4(3) = z(8).GetPixel(0, 3).R

Dim z9diag5(4) As Integer

z9diag5(0) = z(8).GetPixel(4, 0).R

z9diag5(1) = z(8).GetPixel(3, 1).R

z9diag5(2) = z(8).GetPixel(2, 2).R

z9diag5(3) = z(8).GetPixel(1, 3).R

z9diag5(4) = z(8).GetPixel(0, 4).R

Dim z9diag6(5) As Integer

z9diag6(0) = z(8).GetPixel(5, 0).R

z9diag6(1) = z(8).GetPixel(4, 1).R

z9diag6(2) = z(8).GetPixel(3, 2).R

z9diag6(3) = z(8).GetPixel(2, 3).R

z9diag6(4) = z(8).GetPixel(1, 4).R

z9diag6(5) = z(8).GetPixel(0, 5).R

Dim z9diag7(6) As Integer

z9diag7(0) = z(8).GetPixel(6, 0).R

z9diag7(1) = z(8).GetPixel(5, 1).R

z9diag7(2) = z(8).GetPixel(4, 2).R

z9diag7(3) = z(8).GetPixel(3, 3).R

z9diag7(4) = z(8).GetPixel(2, 4).R

z9diag7(5) = z(8).GetPixel(1, 5).R

z9diag7(6) = z(8).GetPixel(0, 6).R

Dim z9diag8(7) As Integer

z9diag8(0) = z(8).GetPixel(7, 0).R

z9diag8(1) = z(8).GetPixel(6, 1).R

z9diag8(2) = z(8).GetPixel(5, 2).R

z9diag8(3) = z(8).GetPixel(4, 3).R

z9diag8(4) = z(8).GetPixel(3, 4).R

z9diag8(5) = z(8).GetPixel(2, 5).R

z9diag8(6) = z(8).GetPixel(1, 6).R

z9diag8(7) = z(8).GetPixel(0, 7).R

Dim z9diag9(8) As Integer

z9diag9(0) = z(8).GetPixel(8, 0).R

z9diag9(1) = z(8).GetPixel(7, 1).R

z9diag9(2) = z(8).GetPixel(6, 2).R

z9diag9(3) = z(8).GetPixel(5, 3).R

z9diag9(4) = z(8).GetPixel(4, 4).R

z9diag9(5) = z(8).GetPixel(3, 5).R

z9diag9(6) = z(8).GetPixel(2, 6).R

z9diag9(7) = z(8).GetPixel(1, 7).R

z9diag9(8) = z(8).GetPixel(0, 8).R

Dim z9diag10(9) As Integer

z9diag10(0) = z(8).GetPixel(9, 0).R

z9diag10(1) = z(8).GetPixel(8, 1).R

z9diag10(2) = z(8).GetPixel(7, 2).R

z9diag10(3) = z(8).GetPixel(6, 3).R

z9diag10(4) = z(8).GetPixel(5, 4).R

z9diag10(5) = z(8).GetPixel(4, 5).R

z9diag10(6) = z(8).GetPixel(3, 6).R

z9diag10(7) = z(8).GetPixel(2, 7).R

z9diag10(8) = z(8).GetPixel(1, 8).R

z9diag10(9) = z(8).GetPixel(0, 9).R

Dim z9diag11(8) As Integer

z9diag11(0) = z(8).GetPixel(9, 1).R

z9diag11(1) = z(8).GetPixel(8, 2).R

z9diag11(2) = z(8).GetPixel(7, 3).R

z9diag11(3) = z(8).GetPixel(6, 4).R

z9diag11(4) = z(8).GetPixel(5, 5).R

z9diag11(5) = z(8).GetPixel(4, 6).R

z9diag11(6) = z(8).GetPixel(3, 7).R

z9diag11(7) = z(8).GetPixel(2, 8).R

z9diag11(8) = z(8).GetPixel(1, 9).R

Dim z9diag12(7) As Integer

z9diag12(0) = z(8).GetPixel(9, 2).R

z9diag12(1) = z(8).GetPixel(8, 3).R

z9diag12(2) = z(8).GetPixel(7, 4).R

z9diag12(3) = z(8).GetPixel(6, 5).R

z9diag12(4) = z(8).GetPixel(5, 6).R

z9diag12(5) = z(8).GetPixel(4, 7).R

z9diag12(6) = z(8).GetPixel(3, 8).R

z9diag12(7) = z(8).GetPixel(2, 9).R

Dim z9diag13(6) As Integer

z9diag13(0) = z(8).GetPixel(9, 3).R

z9diag13(1) = z(8).GetPixel(8, 4).R

z9diag13(2) = z(8).GetPixel(7, 5).R

z9diag13(3) = z(8).GetPixel(6, 6).R

z9diag13(4) = z(8).GetPixel(5, 7).R

z9diag13(5) = z(8).GetPixel(4, 8).R

z9diag13(6) = z(8).GetPixel(3, 9).R

Dim z9diag14(5) As Integer

z9diag14(0) = z(8).GetPixel(9, 4).R

z9diag14(1) = z(8).GetPixel(8, 5).R

z9diag14(2) = z(8).GetPixel(7, 6).R

z9diag14(3) = z(8).GetPixel(6, 7).R

z9diag14(4) = z(8).GetPixel(5, 8).R

z9diag14(5) = z(8).GetPixel(4, 9).R

Dim z9diag15(4) As Integer

z9diag15(0) = z(8).GetPixel(9, 5).R

z9diag15(1) = z(8).GetPixel(8, 6).R

z9diag15(2) = z(8).GetPixel(7, 7).R

z9diag15(3) = z(8).GetPixel(6, 8).R

z9diag15(4) = z(8).GetPixel(5, 9).R

Dim z9diag16(3) As Integer

z9diag16(0) = z(8).GetPixel(9, 6).R

z9diag16(1) = z(8).GetPixel(8, 7).R

z9diag16(2) = z(8).GetPixel(7, 8).R

z9diag16(3) = z(8).GetPixel(6, 9).R

Dim z9diag17(2) As Integer

z9diag17(0) = z(8).GetPixel(9, 7).R

z9diag17(1) = z(8).GetPixel(8, 8).R

z9diag17(2) = z(8).GetPixel(7, 9).R

Dim z9diag18(1) As Integer

z9diag18(0) = z(8).GetPixel(9, 8).R

z9diag18(1) = z(8).GetPixel(8, 9).R

Dim z9diag19 As Integer = z(8).GetPixel(9, 9).R

'HIUNG NILAI ZONA PIXEL 255

If z9diag1 = 255 Then

nzona9 += 1

End If

For mm As Integer = 0 To z9diag2.Length - 1

If z9diag2(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag3.Length - 1

If z9diag3(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag4.Length - 1

If z9diag4(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag5.Length - 1

If z9diag5(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag6.Length - 1

If z9diag6(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag7.Length - 1

If z9diag7(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag8.Length - 1

If z9diag8(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag9.Length - 1

If z9diag9(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag10.Length - 1

If z9diag10(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag11.Length - 1

If z9diag11(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag12.Length - 1

If z9diag12(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag13.Length - 1

If z9diag13(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag14.Length - 1

If z9diag14(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag15.Length - 1

If z9diag15(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag16.Length - 1

If z9diag16(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag17.Length - 1

If z9diag17(mm) = 255 Then

nzona9 += 1

End If

Next

For mm As Integer = 0 To z9diag18.Length - 1

If z9diag18(mm) = 255 Then

nzona9 += 1

End If

Next

If z9diag19 = 255 Then

nzona9 += 1

End If

LabelNZ9.Text = nzona9

ButtonZona.Enabled = False

Lampiran 18. Kode Program Rata – Rata

Dim rz1, rz2, rz3, rz4, rz5, rz6, rz7, rz8, rz9 As Double

rz1 = LabelNZ1.Text / 19

LabelRZ1.Text = rz1

rz2 = LabelNZ2.Text / 19

LabelRZ2.Text = rz2

rz3 = LabelNZ3.Text / 19

LabelRZ3.Text = rz3

rz4 = LabelNZ4.Text / 19

LabelRZ4.Text = rz4

rz5 = LabelNZ5.Text / 19

LabelRZ5.Text = rz5

rz6 = LabelNZ6.Text / 19

LabelRZ6.Text = rz6

rz7 = LabelNZ7.Text / 19

LabelRZ7.Text = rz7

rz8 = LabelNZ8.Text / 19

LabelRZ8.Text = rz8

rz9 = LabelNZ9.Text / 19

LabelRZ9.Text = rz9

'Menghitung Rata Rata Baris

Dim rb1, rb2, rb3 As Double

rb1 = (rz1 + rz2 + rz3) / 3

LabelRB1.Text = rb1

rb2 = (rz4 + rz5 + rz6) / 3

LabelRB2.Text = rb2

rb3 = (rz7 + rz8 + rz9) / 3

LabelRB3.Text = rb3

'Menghitung Rata Rata Kolom

Dim rk1, rk2, rk3 As Double

rk1 = (rz1 + rz4 + rz7) / 3

LabelRK1.Text = rk1

rk2 = (rz2 + rz5 + rz8) / 3

LabelRK2.Text = rk2

rk3 = (rz3 + rz6 + rz9) / 3

LabelRK3.Text = rk3

ButtonRataRata.Enabled = False

Lampiran 19. Kode Program Simpan

Public Sub koneksidatabase()

Dim con As String

con = "server=DESKTOP-0R5DLIR\SQLEXPRESS;integrated security = true; " \_

& "database=skripsi\_db" 'integer security diganti username:;password

Try

conn = New SqlConnection(con)

conn.Open() 'membuka koneksi -- kalau ada buat aplikasi harus ad

conn.Close() 'menutup koneksi

Catch ex As Exception

MsgBox("Database tidak bisa dibuka")

End Try

End Sub

Private Sub saveCitra()

sql = "INSERT INTO citra (namaCitra,fileCitra) VALUES ('" + TextBoxHuruf.Text + "','" + filename + "')"

conn.Open()

cmd = New SqlClient.SqlCommand(sql, conn)

cmd.CommandType = CommandType.Text

cmd.ExecuteNonQuery()

MsgBox("Tersimpan")

conn.Close()

End Sub

Private Function topCitra()

koneksidatabase()

Try

sql = "SELECT TOP 1 \* FROM citra ORDER BY kodeCitra desc"

conn.Open()

cmd = New SqlClient.SqlCommand(sql, conn)

dtReader = cmd.ExecuteReader

dtReader.Read()

hasil = dtReader(0).ToString()

conn.Close()

Return hasil

Catch ex As Exception

Return "gagal"

End Try

End Function

Private Sub saveZona()

Hasil2 = topCitra()

sql = "INSERT INTO rataRata (kodeCitra,rataBaris1,rataBaris2,rataBaris3,rataKolom1,rataKolom2,rataKolom3) VALUES ('" + Hasil2 + "','" + LabelRB1.Text + "','" + LabelRB2.Text + "','" + LabelRB3.Text + "','" + LabelRK1.Text + "','" + LabelRK2.Text + "','" + LabelRK3.Text + "')"

conn.Open()

cmd = New SqlClient.SqlCommand(sql, conn)

cmd.CommandType = CommandType.Text

cmd.ExecuteNonQuery()

MsgBox("Tersimpan")

conn.Close()

End Sub

koneksidatabase()

saveCitra()

saveZona()

ButtonSimpan.Enabled = False

ButtonBrowse.Enabled = False

Lampiran 20. Kode Program Konversi Aksara jawa ke Huruf Latin

Private Function topCitra()

koneksidatabase()

Try

sql = "SELECT TOP 1 \* FROM hasil as h INNER JOIN citra as c ON h.kodehasil = c.kodeCitra ORDER By h.nilaihasil asc"

conn.Open()

cmd = New SqlClient.SqlCommand(sql, conn)

dtReader = cmd.ExecuteReader

dtReader.Read()

hasil = dtReader(3).ToString()

conn.Close()

Return hasil

Catch ex As Exception

Return "gagal"

End Try

End Function

koneksidatabase()

Dim temp As Double

Dim hasil2 As Double

Dim kode As Integer

Dim rata\_baris(2) As Double

Dim rata\_kolom(2) As Double

Dim kurang(5) As Double

rata\_baris(0) = LabelRB1.Text

rata\_baris(1) = LabelRB2.Text

rata\_baris(2) = LabelRB3.Text

rata\_kolom(0) = LabelRK1.Text

rata\_kolom(1) = LabelRK2.Text

rata\_kolom(2) = LabelRK3.Text

For i As Integer = 0 To dgv\_rata.Rows.Count - 1

temp = 0

kode = dgv\_rata.Rows(i).Cells(1).Value

kurang(0) = rata\_baris(0) - dgv\_rata.Rows(i).Cells(2).Value

kurang(1) = rata\_baris(1) - dgv\_rata.Rows(i).Cells(3).Value

kurang(2) = rata\_baris(2) - dgv\_rata.Rows(i).Cells(4).Value

kurang(3) = rata\_kolom(0) - dgv\_rata.Rows(i).Cells(5).Value

kurang(4) = rata\_kolom(1) - dgv\_rata.Rows(i).Cells(6).Value

kurang(5) = rata\_kolom(2) - dgv\_rata.Rows(i).Cells(7).Value

For j As Integer = 0 To kurang.Count - 1

temp = temp + (kurang(j) \* kurang(j))

Next

hasil2 = Math.Sqrt(temp)

sql = "INSERT INTO hasil (kodehasil,nilaihasil) VALUES ('" + kode.ToString + "','" + hasil2.ToString + "')"

conn.Open()

cmd = New SqlClient.SqlCommand(sql, conn)

cmd.CommandType = CommandType.Text

cmd.ExecuteNonQuery()

conn.Close()

Next

LabelHasil.Text = topCitra()

ButtonUjiKemiripan.Enabled = False

ButtonBrowse.Enabled = False

*PROFIL*PENULIS

|  |  |
| --- | --- |
|  | ERSA RAHMAWATI A.  Lumajang, 12 Desember 1994 |
|
| Jalan Mayjen Sukertiyo 108 Lumajang  +6282232595688  [ersarahmawati.a@gmail.com](mailto:ersarahmawati.a@gmail.com)  ersa.rahmawati@polinema.ac.id |

* **Riwayat Pendidikan Formal**

|  |  |  |
| --- | --- | --- |
| Instansi | Masuk | Lulus |
| SD Negeri Ditotrunan 01 Lumajang | 2001 | 2007 |
| SMP Negeri 1 Lumajang | 2007 | 2010 |
| SMA Negeri 2 Lumajang | 2010 | 2013 |
| Politeknik Negeri Malang  Program Studi Teknik Informatika | 2013 | 2017 |

* **Riwayat Organisasi**

|  |  |  |
| --- | --- | --- |
| Organisasi | Periode | Posisi |
| Lembaga Pers Mahasiswa KOMPEN Politeknik Negeri Malang | 2013-2014 | Anggota Penelitian dan Pengembangan |
| Lembaga Pers Mahasiswa KOMPEN Politeknik Negeri Malang | 2014-2015 | Pemimpin Perusahaan |
| Himpunan Mahasiswa Teknologi Informasi | 2014-2015 | Sekretaris 1 |
|  | | |