

**SEGMENTASI CITRA PADA CITRA NASKAH KUNO  
DENGAN MENGGUNAKAN ALGORITMA  
*LOCAL ADAPTIVE THRESHOLDING***

**SKRIPSI**

**VICKY APRIYANI  
71153020**



**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA  
MEDAN  
2020**

**SEGMENTASI CITRA PADA CITRA NASKAH KUNO  
DENGAN MENGGUNAKAN ALGORITMA  
LOCAL ADAPTIVE THRESHOLDING**

**SKRIPSI**

*Diajukan Untuk Memenuhi Syarat Mencapai Gelar Sarjana Komputer*

**VICKY APRIYANI  
NIM. 71153020**



**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA  
MEDAN  
2020**

## PERSETUJUAN SKRIPSI

Hal : Surat Persetujuan Skripsi

Lamp : -

Kepada Yth.,  
Dekan Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sumatera Utara Medan

*Assalamu'alaikumWr. Wb*

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan, maka kami selaku pembimbing berpendapat bahwa skripsi saudara,

Nama : Vicky Apriyani  
NIM : 71153020  
Program Studi : Ilmu Komputer  
Judul : Segmentasi Citra Pada Citra Naskah Kuno Dengan  
Menggunakan *Algoritma Local Adaptive  
Thresholding*

Dapat disetujui untuk segera di *munaqasyahkan*. Atas perhatiannya kami ucapkan terimakasih

Medan, 13 Agustus 2020

23 Zulhijah 1441

Pembimbing Skripsi I, Komisi Pembimbing,  
Pembimbing Skripsi II,

Dr. Mhd. Furqan, S.Si, M. Comp. Sc  
NIP. 198008062006041003

Rakhmat Kurniawan R, ST, M.Kom  
NIP. 198503162015031003

## SURAT PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini;

Nama : Vicky Apriyani  
NIM : 71153020  
Program Studi : Ilmu Komputer  
Judul : Segmentasi Citra Pada Citra Naskah Kuno Dengan  
Menggunakan *Algoritma Local Adaptive  
Thresholding*

Menyatakan bahwa skripsi ini hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya. Apabila dikemudian hari ditemukan plagiat dalam skripsi ini maka saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi lainnya sesuai dengan peraturan yang berlaku.

Medan, 13 Agustus 2020

Vicky Apriyani  
NIM. 71153020



KEMENTERIAN AGAMA REPUBLIK INDONESIA  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA MEDAN  
FAKULTAS SAINS DAN TEKNOLOGI  
Jl. IAIN No. 1 Medan 20235  
Telp. (061) 6615683-6622925, Fax. (061) 6615683  
Url: <http://saintek.uinsu.ac.id>, E-mail: [saintek@uinsu.ac.id](mailto:saintek@uinsu.ac.id)

---

### PENGESAHAN SKRIPSI

Nomor: B219/ST/ST.V.2/PP.01.1/12/2021

Judul : Segmentasi Citra Pada Citra Naskah Kuno Dengan  
*Menggunakan Algoritma Local Adaptive Thresholding*  
Nama : Vicky Apriyani  
Nomor Induk Mahasiswa : 71153020  
Program Studi : Ilmu Komputer  
Fakultas : Sains dan Teknologi

Telah dipertahankan di Hadapan Dewan Penguji Skripsi Program Studi Ilmu Komputer  
Fakultas Sain dan Teknologi UIN Sumatera Utara Medan dan Dinyatakan LULUS.  
Pada hari/tanggal : Kamis, 13 Agustus 2020  
Tempat/media : Via Zoom Meeting

Tim Ujian Munaqasyah  
Ketua,

Dr. Mhd. Furqan, S.Si, M.Comp.Sc  
NIP. 198008062006041003

Dewan Penguji	
Penguji I	Penguji II
<u>Dr. Mhd. Furqan, S.Si, M.Comp.Sc</u> NIP. 198008062006041003	<u>Rakhmat Kurniawan R, ST, M.Kom</u> NIP. 198503162015031003
Penguji III	Penguji IV
<u>Sriani, S.Kom., M.Kom</u> NIB. 1100000108	<u>Abdul Halim Hasugian, M.Kom</u> NIB. 1100000113

Mengesahkan  
Dekan Fakultas Sains dan Teknologi  
UIN Sumatera Utara Medan

Dr. H. M. Jamil, M.A  
NIP.196609101999031002

## ABSTRAK

Sebagian dari naskah kuno di Museum Negeri Provinsi Sumatera Utara mengalami kerusakan seperti robek, tulisan tidak terlihat jelas sehingga susah untuk dibaca dan terdapat noda di naskah tersebut karena sudah puluhan bahkan ratusan tahun. Naskah kuno tak akan bertahan lama apabila tidak dirawat dengan baik. Oleh sebab itu dibutuhkan sebuah perawat yang dapat melestarikan naskah kuno dengan jangka yang lebih lama. Salah satu cara untuk melestarikan naskah kuno dan membuat naskah kuno lebih dapat dibaca serta jelas, maka dapat dilakukan dengan cara segmentasi citra. Dengan segmentasi citra, naskah kuno dapat diubah kedalam bentuk naskah kuno citra digital yang bisa disimpan kedalam media penyimpanan.

Segmentasi citra yang diterapkan adalah melakukan pemisahan *background* citra naskah kuno dengan teks naskah kuno kedalam bentuk citra hitam dan putih, sehingga citra naskah kuno yang tidak jelas dapat dibaca lebih baik. Segmentasi citra memerlukan metode, pada penelitian ini metode yang digunakan ialah *Local Adaptive Thresholding*, tahapan metode *Local Adaptive Thresholding* adalah mengubah citra naskah kuno kedalam citra *luminosity grayscale*, kemudian menghitung nilai *mean* dan *threshold*, sehingga dilakukan pemisahan berdasarkan nilai *threshold*. Pada penelitian ini, metode *Local Adaptive Thresholding* berhasil melakukan segmentasi citra naskah kuno dengan baik dan jelas, klemahanya adalah ketika citra naskah kuno yang terlalu banyak mengandung karakter dan jarak yang terlalu jauh, maka hasilnya sedikit mengalami pengelapan.

**Kata Kunci:** Naskah Kuno, Segmentasi, Citra, Adaptif Thresholding

## ABSTRACT

Some of the ancient manuscripts in the North Sumatra Province State Museum suffered damage such as being torn, the writing was not seen clearly so it was difficult to read and there were stains on the manuscript because of his age that was tens or even hundreds of years old. Ancient manuscripts will not last long if not handled or cared for properly. That is why a nurse is needed who can preserve ancient texts with a longer term. One way to preserve ancient manuscripts and make ancient manuscripts more readable and clear, can be done by image segmentation. With image segmentation, ancient manuscripts can be converted into digital manuscripts that can be stored on storage media.

The image segmentation that is applied is to separate the background image of the ancient manuscript with the text of the ancient manuscript into the form of black and white images, so that the unclear image of the ancient manuscript can be read better. Image segmentation requires a method, in this study the method used is Local Adaptive Thresholding, the stage of the Adaptive Thresholding method is to change the image of an ancient manuscript into a grayscale luminosity image, then calculate the mens and threshold values, so that separation is done based on the threshold value. In this study, the Local Adaptive Thresholding method succeeded in segmenting the image of the ancient manuscript well and clearly, the friendliness was when the image of an ancient manuscript contained too much character and the distance was too far, then the results were slightly darkening.

**Keywords:** Ancient Manuscript, Segmentation, Image, Adaptive Thresholding

## KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah SWT penguasa alam semesta, pemberi nikmat kepada segenap makhluk-Nya, yaitu berupa nikmat islam, kesehatan, dan keselamatan. Dan tak lupa juga shalawat dan salam semoga dilimpahkan kepada junjungan kita Nabi Muhammad SAW beserta keluarga dan para sahabatnya yang telah membimbing kita ke jalan yang benar dan diridhoin oleh Allah SWT.

Skripsi ini disusun untuk melengkapi salah satu syarat kelulusan strata satu (S1) pada Program Studi Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara. Skripsi ini berjudul ***“Segmentasi Citra Pada Citra Naskah Kuno Dengan Menggunakan Algoritma Local Adaptive Thresholding”***.

Dalam menyusun skripsi ini banyak rintangan dan kesulitan yang penulis hadapin, tetapi dengan adanya bantuan, bimbingan, serta dukungan dan do’a dari berbagai pihak yang diberikan kepada penulis, maka semua kesulitan dapat teratasi. Pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. H. Saidurrahman, M. Ag, sebagai Rektor Universitas Islam Negeri Sumatera Utara.
2. Bapak Dr. H. M. Jamil, MA, sebagai Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara.
3. Bapak Dr. Mhd. Furqan, S.Si., M.Comp.Sc. sebagai Ketua Jurusan Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara sekaligus dosen pembimbing I yang telah bersedia meluangkan waktunya untuk memberikan bimbingan serta arahan kepada penulis sehingga penulis dapat menyelesaikan skripsi ini dengan baik.
4. Bapak Rakhmat Kurniawan R, ST, M.Kom sebagai Kepala Laboraturium Fakultas Sains da Teknologi Universitas Islam Negeri Sumatera Utara sekaligus dosen pembimbing II yang telah membimbing dan membantu penulis dalam memberikan ide, kritik, dan saran kepada penulis.



5. Kepada seluruh Dosen Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara yang telah memberikan ilmu selama dibangku perkuliahan
6. Kedua orang tua yang saya sayangi. Terimakasih atas do'a, nasehat, serta dukungan yang luar biasa kepada penulis .
7. Kepada sahabat seperjuangan, Sri Ramadhani, Irma Suryani, serta Meri Sinaga yang turut menunjang serta membagikan semangat terhadap penulis.
8. Serta seluruh pihak yang turut membantu dengan tulus.

Dalam menyusun skripsi ini penulis menyadari bahwa skripsi ini masih banyak kekurangan, untuk itu penulis mengharapkan kritik dan saran dari pembaca. Semoga skripsi ini dapat bermanfaat.

Medan, Agustus 2020  
Penulis,

Vicky Apriyani

## DAFTAR ISI

	Halaman
<b>ABSTRAK .....</b>	<b>i</b>
<b>ABSTRACT .....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR.....</b>	<b>vii</b>
<b>DAFTAR TABEL .....</b>	<b>ix</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>x</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	4
<b>BAB II TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1 Pengertian Naskah Kuno.....	5
2.2 Pengertian Citra.....	5
2.2.1 Citra Analog.....	6
2.2.2 Citra Digital .....	6
2.3 Jenis-Jenis Citra .....	7
2.3.1 Citra Warna.....	7
2.3.2 Citra <i>Grayscale</i> .....	9
2.3.3 Citra Biner .....	10
2.4 Pengolahan Citra Digital .....	10
2.4.1 Akuisi Citra.....	11
2.4.2 <i>Sampling</i> .....	12
2.4.3 Kuantisasi.....	12
2.5 Format File Citra .....	14
2.6 Segmentasi Citra .....	15

2.7 Algoritma <i>Local Adaptive Thresholding</i> .....	16
2.8 <i>Flowchart</i> .....	17
2.9 Visual Studio 2002 .....	18
2.9.1 Bahasa Pemrograman C#.....	19
2.10 Penelitian Terkait .....	19
<b>BAB III METODE PENELITIAN .....</b>	<b>21</b>
3.1 Tempat dan Waktu Penelitian.....	21
3.1.1 Tempat Penelitian .....	21
3.1.2 Waktu dan Jadwal Penelitian.....	21
3.2 Bahan dan Alat Penelitian.....	22
3.2.1 Perangkat Keras .....	22
3.2.2 Perangkat Lunak .....	22
3.3 Cara Kerja Penelitian .....	22
3.3.1 Metode Penelitian.....	22
3.3.2 Desain Penelitian.....	24
3.3.3 Teknik Pengumpulan Data .....	24
3.3.4 Analisa Kebutuhan .....	25
3.3.5 Perancangan Sistem.....	26
3.3.6 Rancangan Flowchart Penelitian .....	29
<b>BAB IV PERANCANGAN DAN HASIL .....</b>	<b>32</b>
4.1 Rancangan Interface.....	32
4.2 Pengujian <i>Local Algoritma Adaptive Thresholding</i> .....	35
4.3 Hasil Rancangan Interface Pada Aplikasi .....	41
4.3.1 Proses Segmentasi Citra Pada Aplikasi.....	44
4.4 Hasil dan Pembahasan .....	54
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>56</b>
5.1 Kesimpulan .....	56
5.2 Saran .....	56
<b>DAFTAR PUSTAKA .....</b>	<b>58</b>
<b>LAMPIRAN-LAMPIRAN</b>	

## DAFTAR GAMBAR

<b>Gambar</b>	<b>Judul Gambar</b>	<b>Halaman</b>
Gambar 2.1	Koordinat Citra Digital.....	6
Gambar 2.2	Contoh Penyimpanan Citra Warna Didalam Memor .....	8
Gambar 2.3	Citra <i>Grayscale</i> .....	9
Gambar 2.4	Perbandingan Gradasi Warna 1 <i>bit</i> , 2 <i>bit</i> , 5 <i>bit</i> , 6 <i>bit</i> , 7 <i>bit</i> , 8 <i>bit</i> .....	9
Gambar 2.5	Citra Biner .....	10
Gambar 2.6	Contoh Proses Akuisi Citra Digital .....	11
Gambar 2.7	(a) Citra Analog, (b) Citra Analog Disampling Menjadi 14 Baris dan 12 kolom, (c) Citra Digital Hasil Sampling Berukuran 14x12 Piksel.....	12
Gambar 2.8	Kuantisasi .....	13
Gambar 2.9	Perbedaan Global Dengan Adaptif <i>Thresholding</i> .....	16
Gambar 3.1	Bagan Langkah Penelitian Tindakan .....	23
Gambar 3.2	Bagan Rancangan Penelitian.....	27
Gambar 3.3	<i>Flowchart</i> Sistem .....	29
Gambar 3.4	<i>Flowchart</i> Segmentasi.....	30
Gambar 3.5	<i>Flowchart Form</i> Bantuan .....	31
Gambar 3.6	<i>Flowchart</i> Tentang Penulis .....	31
Gambar 4.1	Rancangan Menu Utama .....	32
Gambar 4.2	Rancangan <i>Form</i> Segmentasi Citra.....	33
Gambar 4.3	Rancangan <i>Form</i> Bantuan .....	34
Gambar 4.4	Rancangan <i>Form</i> Tentang Penulis .....	35
Gambar 4.5	Citra Naskah Kuno Sampel .....	36
Gambar 4.6	Hasil <i>Grayscale</i> Sampel Citra Naskah Kuno .....	37
Gambar 4.7	Hasil <i>Adaptive Thresholding</i> Sampel Citra Naskah Kuno.....	41
Gambar 4.8	Menu Utama Aplikasi .....	42
Gambar 4.9	Menu Segmentasi Citra .....	42

Gambar 4.10 Menu Bantuan .....	43
Gambar 4.11 Menu Tentang Penulis.....	44
Gambar 4.12 Proses Pemilihan Citra Naskah Kuno .....	46
Gambar 4.13 Pemilihan Citra Naskah Kuno.....	46
Gambar 4.14 Pemilihan Citra Naskah Kuno Sampel1 .....	47
Gambar 4.15 Hasil Proses Segmentasi Citra Naskah Kuno Sampel1 .....	48
Gambar 4.16 Hasil Proses Segmentasi Citra Naskah Kuno Sampel2.....	49
Gambar 4.17 Hasil Proses Segmentasi Citra Naskah Kuno Sampel3.....	49
Gambar 4.18 Hasil Proses Segmentasi Citra Naskah Kuno Sampel4.....	50
Gambar 4.19 Hasil Proses Segmentasi Citra Naskah Kuno Sampel5.....	51
Gambar 4.20 Proses Penyimpanan Citra Hasil Segmentasi .....	51
Gambar 4.21 Citra Naskah Kuno Hasil Segmentasi Sampel .....	52
Gambar 4.22 Citra Naskah Kuno Hasil Segmentasi Sampel2 .....	52
Gambar 4.23 Citra Naskah Kuno Hasil Segmentasi Sampel3 .....	53
Gambar 4.24 Citra Naskah Kuno Hasil Segmentasi Sampel4 .....	53
Gambar 4.25 Citra Naskah Kuno Hasil Segmentasi Sampel5 .....	54

## DAFTAR TABEL

<b>Tabel</b>	<b>Judul Tabel</b>	<b>Halaman</b>
Tabel 2.1	Warna Dan Nilai Penyusun Warna .....	8
Tabel 2.2	Simbol-Simbol <i>Flowchart</i> .....	17
Tabel 3.1	Jadwal Pelaksanaan Penelitian .....	21
Tabel 3.1	Kebutuhan <i>Hardware</i> .....	22
Tabel 4.1	Nilai pixel 3 x 3 Sampel Naskah Kuno.....	36
Tabel 4.2	Hasil Nilai Pixel 3 x 3 <i>Grayscale</i> Sampel Naskah Kuno Citra.....	37
Tabel 4.3	Hasil Nilai Batas <i>Threshold</i> .....	40
Tabel 4.4	Hasil Nilai Biner Pixel 3 x 3 Naskah Kuno .....	40
Tabel 4.5	Citra Naskah Kuno.....	45
Tabel 4.6	Hasil Perbandingan Ukuran Citra Naskah Kuno .....	54

## DAFTAR LAMPIRAN

<b>Lampiran</b>	<b>Judul Lampiran</b>
1.	Listing Program
2.	Perhitungan Manual
3.	Daftar Riwayat Hidup
4.	Kartu Bimbingan Skripsi

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Naskah Kuno adalah buku yang menyimpan informasi atau hasil pemikiran masyarakat seperti nilai sejarah, adat istiadat, kebiasaan, ilmu pengetahuan, serta kebudayaan dimasa lampau pada suatu daerah yang dibuat dalam bentuk tulisan tangan, berusia puluhan hingga ratusan tahun dan harus dilestarikan keberadaannya (Bahar & Mathar, 2015). Salah satu tempat di Kota Medan yang menyimpan naskah kuno yaitu Museum Negeri Provinsi Sumatera Utara .

Di dalam Al Qur'an surah Yusuf : 111 dijelaskan tentang sejarah orang-orang terdahulu, seperti berikut ini:

لَقَدْ كَانَ فِي قَصَصِهِمْ عِبْرَةٌ لِأُولَى الْأَلْبَابِ ۗ مَا كَانَ حَدِيثًا يُفْتَرَىٰ وَلَكِن تَصَدِيقَ الَّذِي بَيْنَ يَدَيْهِ وَتَفْصِيلَ كُلِّ شَيْءٍ وَهُدًى وَرَحْمَةً لِّقَوْمٍ يُؤْمِنُونَ

Artinya: “Sesungguhnya pada kisah-kisah mereka itu terdapat pengajaran bagi orang-orang yang mempunyai akal. Al Qur'an itu bukanlah cerita yang dibuat-buat, akan tetapi membenarkan (kitab-kitab) yang sebelumnya dan menjelaskan segala sesuatu, dan sebagai petunjuk dan rahmat bagi kaum yang beriman”.

Naskah kuno di Museum Negeri Provinsi Sumatera Utara hanya diletakkan dan disimpan di dalam lemari kaca. Sebagian dari naskah tersebut mengalami kerusakan seperti warna kertas yang sudah kecoklatan dan menghitam, robek, tulisan tidak terlihat jelas sehingga susah untuk dibaca dan terdapat noda di naskah tersebut karena sudah puluhan bahkan ratusan tahun. Jika tidak dirawat dengan baik, maka naskah tersebut tidak akan bertahan lama. Karena itu, perawatan naskah kuno merupakan hal yang sangat penting. Mengingat banyak naskah kuno yang secara fisik kondisinya sudah sangat memprihatinkan dan ditangani dengan kurang baik.

Maka berdasarkan masalah tersebut untuk mempertahankan kelestarian naskah kuno dari bentuk fisiknya yang mulai hancur dan tidak terbaca, dibutuhkan



sebuah pengubahan bentuk naskah kuno kedalam citra digital yang dapat disimpan kedalam media penyimpanan. Perubahan bentuk fisik naskah kuno dapat dilakukan dengan cara difoto menggunakan kamera digital. Akan tetapi hasil foto menggunakan kamera digital tentu tidak dapat merubah naskah kuno yang sudah tidak dapat dibaca dengan jelas. Oleh sebab itu, hasil digital dari naskah kuno diperlukan proses segmentasi sehingga naskah kuno berbentuk digital dapat dibaca dengan lebih baik.

Segmentasi adalah proses mempartisi citra menjadi beberapa daerah atau objek (Sinaga, 2017). Tujuan dari segmentasi citra yaitu untuk memisahkan daerah objek dengan latar belakang sehingga objek mudah untuk di analisis (Kumaseh, dkk, 2013). Proses segmentasi dengan objek citra digital yang dimulai dengan mengubah citra digital kedalam bentuk *grayscale* (abu-abu) kemudian dilakukan pemisahan objek citra dan *background* citra kedalam bentuk hitam putih. Tahapan segmentasi citra digital tentu memerlukan perhitungan algoritma. Salah satu algoritma yang dapat diterapkan adalah algoritma *local Adaptive Thresholding*.

*Algoritma Local Adaptive Thresholding* merupakan metode yang menggunakan nilai ambang secara otomatis, yaitu dengan mengubah citra digital warna abu-abu menjadi warna hitam putih berdasarkan perbandingan nilai ambang dengan nilai warna *pixel* (Basuki dan Hariadi, 2012). Tahapan yang dilakukan dalam segmentasi naskah kuno dalam penelitian ini yaitu proses akuisi citra, proses *preprocessing*, dan segmentasi dengan menggunakan *Algoritma Local Adaptive Thresholding*.

Berdasarkan uraian diatas, maka peneliti mengangkat judul “**Segmentasi Citra Pada Citra Naskah Kuno Dengan Menggunakan *Algoritma Local Adaptive Thresholding***”.

### 1.1 Rumusan Masalah

Dari latar belakang diatas, maka rumusan masalah dalam penelitian ini yaitu sebagai berikut:

1. Bagaimana menerapkan *Algoritma Local Adaptive Thresholding* untuk segmentasi citra naskah kuno ?
2. Bagaimana merancang dan membangun aplikasi segmentasi citra naskah kuno menggunakan bahasa pemograman C# ?

### 1.2 Batasan Masalah

Agar penelitian ini tidak keluar dari pokok permasalahan yang dirumuskan, maka ruang lingkup pembahasan diberi batasan masalah sebagai berikut:

1. Citra yng diambil yaitu citra naskah kuno di Museum Negeri Provinsi Sumatera Utara.
2. Metode segmentasi yang digunakan *Algoritma Local Adaptive Thresholding*.
3. Proses segmentasi citra naskah kuno dengan *Algoritma Local Adaptive Thresholding* menurunkan ukuran asli citra dengan resolusi yang sama.
4. Dalam segmentasi ini aplikasi yang digunakan yaitu *Microsoft Visual Studio 2012* dengan bahasa pemograman C#.

### 1.3 Tujuan Penelitian

Berdasarkan rumusan masalah diatas, maka tujuan dari penelitian ini yaitu sebagai berikut:

1. Menerapkan *algoritma Adaptive Thresholding* sebagai metode segmentasi pada citra naskah kuno.
2. Mengetahui hasil akhir dari naskah kuno yang di segmentasikan.
3. Untuk melestarikan warisan budaya peninggalan sejarah.
4. Diharapkan mampu meningkatkan kualitas citra yang akan disegmentasi

#### **1.4 Manfaat Penelitian**

Berdasarkan latar belakang diatas, maka manfaat dari penelitian ini yaitu sebagai berikut:

1. Membantu masyarakat lebih mudah dalam membaca tulisan dari naskah kuno tersebut.
2. Agar minat generasi muda dalam melestarikan pwninggalan sejarah dapat meningkat.
3. Menambah ilmu pengetahuan dalam bidang pendidikan khususnya ilmu komputer.
4. Kedepannya penelitian ini bisa menjadi bahan referensi untuk peneliti yang melakukan penelitian dengan menggunakan metode yng sama untuk objek yang berbeda.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Pengertian Naskah Kuno**

Naskah di ambil dari bahasa Arab, yaitu al-naskhah artinya “manuskrip”. Kata manu di dapat dari kata *manus* artinya yaitu tangan, dan *scriptus* di dapat dari kata *scriber* artinya yaitu menulis. Jadi manuskrip artinya tulisan tangan. Naskah merupakan hasil tulisan tangan dari budaya bangsa terdahulu yang menyimpan berbagai ungkapan pikiran dan perasaan. Naskah kuno merupakan warisan budaya bangsa yang memiliki berbagai pemikiran, pengetahuan, perilaku, serta adat istiadat dari masyarakat masa lalu (Latiar, 2018).

Dalam jurnal lain naskah kuno merupakan warisan berdasarkan pada sebuah kehidupan pada masa lalu yang mempunyai nilai informasi yang sangat berharga yang ditulis dalam bermacam-macam bahasa dan aksara (Tajuddin, dkk, 2017).

#### **2.2 Pengertian Citra**

Citra adalah gabungan antara titik, garis, bidang, dan warna untuk menciptakan sesuatu yang menyerupai dari suatu objek. Menurut (Furqan, dkk, 2020) ada dua macam citra, yaitu citra analog, dan citra digital, yang dikatakan citra analog yaitu citra yang berupa sinyal video, foto atau gambar di layer televisi, sedangkan citra digital adalah citra yang bias disimpan langsung di media penyimpanan. Dalam jurnal lain, citra adalah fungsi menerus (*continue*) berdasarkan intensitas cahaya dalam bidang 2 dimensi yang menjadi salah satu komponen multimedia yang memegang peranan yang sangat penting dalam bentuk informasi visual (Apriliani & Murinto, 2013).

Menurut (Prasetyo, 2011) suatu citra bisa diartikan fungsi dua dimensi  $f(x,y)$ , dimana  $x$  dan  $y$  merupakan suatu koordinat spasial, dan amplitudo dari  $f$  pada

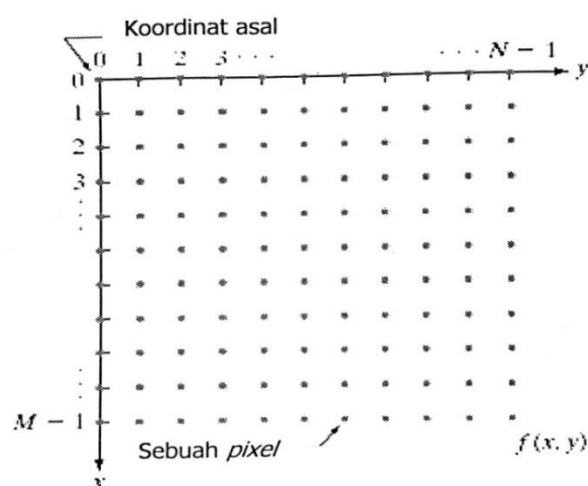
sembarang pasangan koordinat  $(x,y)$  adalah intensitas (*intensity*) atau level keabuan (*gray level*) dari citra pada titik tersebut.

### 2.2.1 Citra Analog

Citra analog merupakan citra yang bersifat kontinu, misalnya gambar dalam televisi, foto sinar X, foto yang tercetak pada kertas, lukisan, pemandangan, dan lain sebagainya. Citra analog tidak bisa diproses secara langsung dikomputer. Oleh karena itu, supaya citra ini bisa diproses pada komputer, terlebih dahulu citra analog wajib diubah ke citra digital. Citra analog didapatkan dari alat-alat analog, misalnya video kamera analog, kamera foto sistem radar, sensor ultrasound dalam sistem USG dan lain-lain (Sutoyo, dkk, 2009).

### 2.2.2 Citra Digital

Citra digital adalah sebuah larik (array) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan menggunakan deretan bit tertentu. Suatu citra bisa didefinisikan menjadi fungsi 2 dimensi  $f(x,y)$  ukuran M baris dan N kolom, dimana x dan y merupakan koordinat spasial, dan amplitudo  $f$  dititik koordinat  $(x,y)$  dinamakan intensitas atau tingkat keabuan berdasarkan citra dalam titik tersebut (Putra, 2010).



**Gambar 2.1** Koordinat citra digital (Putra, 2010)

Nilai pada suatu irisan antara baris dan kolom dalam posisi  $x,y$  dinamakan dengan *picture elements*, *image elements*, *pels* atau *pixel* (Putra, 2010). Sebuah citra digital bias diwakili dari sebuah matriks yang terdiri dari  $M$  kolom dan  $N$  baris, perpotongan antara kolom dan baris dinamakan *pixel*, *pixel* merupakan elemen yang terkecil dalam sebuah citra. *Pixel* memiliki 2 parameter, yaitu koordinat dan intensitas atau warna. Nilai yang ada dalam koordinat  $(x,y)$  adalah  $f(x,y)$ , yaitu besar intensitas atau warna pada *pixel* dari koordinat tersebut. Bentuk matriks pada citra digital dapat ditulis seperti berikut ini:

$$f(x,y) \begin{bmatrix} f(0,0) & f(0,0) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \dots\dots\dots (1)$$

Dari gambar diatas, cintra digital secara matematis berisikan sebagai fungsi intensitas  $f(x,y)$ , koorninat posisi baris dan kolom adalah  $x$  dan  $y$ , nilai fungsi dalam setiap titik  $(x,y)$  merupakan besar intensitas citra atau tingkat keabuan berdasarkan *pixel* tersebut (Sutoyo, dkk, 2009).

## 2.3 Jenis-Jenis Citra

Jenis-jenis citra pada proses segmentasi citra dibagi menjadi tiga jenis, adapun penjelasan dari ketiga jenis-jenis citra tersebut sebagai berikut:

### 2.3.1 Citra Warna

Citra berwarna adalah citra yang memiliki tiga warna dasar yaitu RGB (*red*, *green*, *blue*). Setiap warna dalam citra RGB menggunakan 8 bit (nilainya berkisar dari 0 hingga 255). Dengan kata lain, kemungkinan warna yang didapat mencapai  $255 \times 255 \times 255$  atau 16.581.375 warna. Tabel 2.1 menunjukkan contoh warna dan nilai RGB.

**Tabel 2.1** Warna dan Nilai Penyusun Warna (Nafi'iyah, 2015)

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255

Menurut (Sutoyo, dkk, 2009) dalam bukunya, tiap piksel dalam citra warna mewakili campuran dari 3 warna dasar yaitu *red*, *green*, *blue*. Tiap warna dasar memakai penyimpanan 8 bit = 1 *byte*, yang maksudnya yaitu tiap warna memiliki gradasi sebanyak 255 warna. Berarti tiap piksel memiliki campuran warna sebanyak  $2^8 \cdot 2^8 \cdot 2^8 = 2^{24} = 16$  juta warna lebih. Seperti itu sebabnya format ini disebut *true color* karena memiliki jumlah warna yang lumayan besar sehingga dapat dikatakan nyaris mencakup seluruh warna di alam.

Citra warna dan citra *grayscale* disimpan ke dalam memori yang berbeda. 1 *byte* mewakili 256 gradasi warna dari tiap piksel citra *grayscale*. Sedangkan 3 *byte* mewakili 1 citra *true color*, jadi tiap masing-masing dari *byte* tersebut mewakili warna *red*, *green*, *blue*. Gambar 2.2 adalah contoh penyimpanan citra warna di dalam memori.

Citra warna	Penyimpanan di dalam memori				
	R = 50	R = 40	R = 90	R = 80	R = 70
	G = 65	G = 40	G = 90	G = 50	G = 70
	B = 50	B = 55	B = 90	B = 50	B = 70
	R = 40	R = 50	R = 40	R = 20	R = 50
	G = 80	G = 80	G = 90	G = 20	G = 60
	B = 30	B = 50	B = 80	B = 50	B = 70
	R = 80	R = 70	R = 80	R = 10	R = 90
	G = 60	G = 70	G = 90	G = 70	G = 90
	B = 40	B = 70	B = 70	B = 10	B = 90
	R = 90	R = 40	R = 70	R = 60	R = 50
	G = 60	G = 60	G = 70	G = 20	G = 80
	B = 70	B = 50	B = 70	B = 40	B = 50
	R = 60	R = 40	R = 80	R = 70	R = 90
	G = 60	G = 60	G = 80	G = 60	G = 80
	B = 60	B = 80	B = 80	B = 50	B = 70

**Gambar 2.2** Contoh penyimpanan citra warna di dalam memori

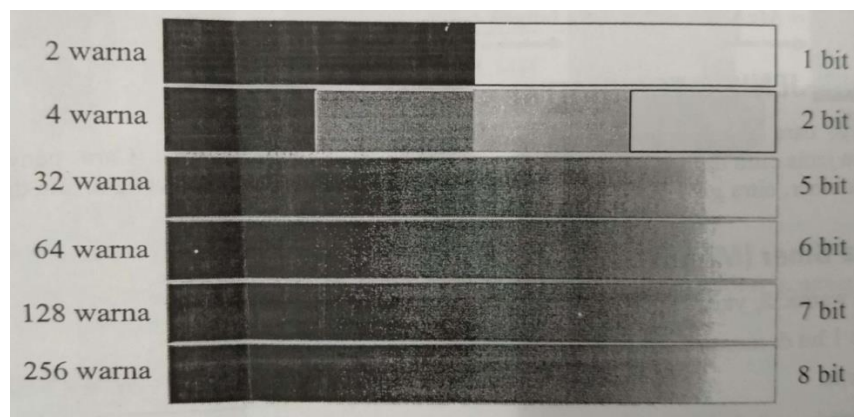
### 2.3.2 Citra *Grayscale*

Citra *grayscale* yaitu citra yang nilai pikselnya berdasarkan derajat keabuan. Nilai tersebut digunakan untuk menunjukkan nilai intensitas. Ada tiga jenis warna pada citra *grayscale* yaitu warna abu-abu, warna putih (nilai intensitas tertinggi), dan warna hitam (nilai intensitas terendah). Citra *grayscale* mempunyai kedalaman warna 8 *bit* (256 kombinasi warna keabuan) (Putra, 2010). Gambar 2.3 menunjukkan citra *grayscale*.



**Gambar 2.3** Citra *grayscale* (Putra, 2010)

Menurut (Sutoyo, dkk, 2009), banyaknya warna pada citra *grayscale* tergantung dari jumlah *bit* yang disediakan di memori untuk menampung kebutuhan warna. Semakin besar jumlah *bit* warna yang disediakan, maka semakin halus gradasi warna yang terbentuk. Gambar 2.3 menunjukkan perbandingan gradasi warna untuk jumlah *bit* tertentu.



**Gambar 2.4** Perbandingan gradasi warna 1 *bit*, 2 *bit*, 5 *bit*, 6 *bit*, 7 *bit*, 8 *bit*



### 2.3.3 Citra Biner

Citra biner merupakan citra yang hanya terdiri dari 2 warna, yaitu hitam dan putih. Tiap *pixel* pada citra tersebut direpresentasikan pada 1 *bit*. (Putra, 2010). Dalam jurnal (Nafi'iyah, 2015), citra biner adalah citra yang tiap pikselnya memiliki dua nilai kemungkinan yaitu nilai 1 (putih) dan 0 (hitam). Gambar 2.5 menunjukkan citra biner.



**Gambar 2.5** Citra biner (Putra, 2010)

## 2.4 Pengolahan Citra Digital

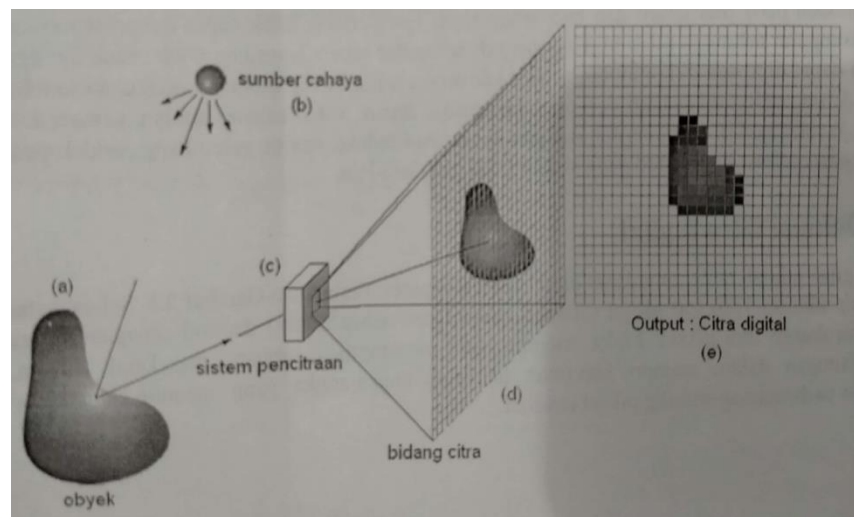
Menurut (Putra, 2010), pengolahan citra digital secara umum merupakan suatu pemrosesan gambar 2 dimensi dengan menggunakan komputer. Pengolahan citra digital adalah suatu bidang ilmu yang mempelajari bagaimana citra tersebut dapat diproses, dibnetuk, dan dianalisis. Tujuan dari pengolahan citra itu sendiri ialah untuk memperbaiki kualitas suatu foto seperti peningkatan kontras, transformasi warna, serta restorasi citra, kemudian transformasi foto seperti skala, translasi, rotasi, serta transformasi geometrik, melaksanakan pemilihan *feature images* untuk tujuan analisis, melaksanakan proses penarikan data yang terdapat dalam citra, melaksanakan kompresi ataupun reduksi informasi, transmisi

informasi, serta waktu proses informasi. *Input* dari pengolahan citra merupakan citra, sebaliknya *outputnya* merupakan citra hasil pengolahan (Sutoyo, dkk, 2009).

Ada tiga tahapan dalam proses pengolahan citra, yaitu akuisi citra, *sampling*, dan kuantisasi.

#### 2.4.1 Akuisi citra

Akuisi citra merupakan langkah pertama dalam mendapatkan suatu citra digital. Tujuan akuisi citra ialah memilih metode perekaman pada citra digital serta untuk menentukan data yang dibutuhkan (Sutoyo, dkk, 2009). Langkah pertama dalam proses akuisi citra ialah pengambilan gambar objek, mempersiapkan alat-alat, hingga pada pencitraan (Sutoyo, dkk, 2009).



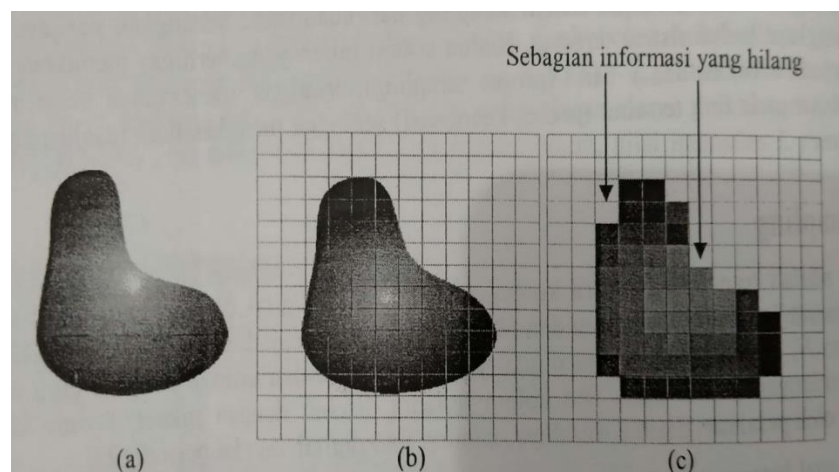
**Gambar 2.6** Contoh proses akuisi citra digital (Sutoyo, dkk, 2009)

Dari gambar diatas dapat dijelaskan bahwa terdapat suatu objek yang hendak diambil gambarnya untuk dijadikan citra digital. Sumber sinar sangat dibutuhkan untuk menerangi objek, yang artinya terdapat intensitas cahaya yang diterima oleh objek. Oleh objek, intensitas cahaya ini sebagian diserap serta sebagian lagi dipantulkan ke area dekat objek yang radial, sistem pencitraan menerima sebagian dari intensitas cahaya yang dipantul oleh objek tadi. Didalam sistem pencitraan ada sensor optik yang digunakan untuk mengetahui intensitas sinar yang masuk kedalam sistem. Keluaran dari sensor ini berbentuk arus yang besarnya sebanding dengan intensitas cahaya yang mengenainya. Setelah itu arus

tersebut diubah menjadi informasi digital yang kemudian dikirim ke unit penampil ataupun unit pengolah yang lain. Secara keseluruhan hasil *output* pada sistem pencitraan berbentuk citra digital (Sutoyo, dkk, 2009).

### 2.4.1 Sampling

Sampling adalah membagi citra analog (*continue*) menjadi M kolom dan N baris sehingga menjadi citra diskrit, sehingga mengubah citra kontinu menjadi citra digital. Semakin besar nilai M dan N, semakin halus gambar yang dihasilkan, yang berarti semakin tinggi resolusi gambar. Proses pengambilan sampling dihasilkan dari peralatan digital, seperti *scanner*, foto digital, serta kamera digital (Sutoyo, dkk, 2009).



**Gambar 2.7** (a) Citra analog, (b) Citra analog disampling menjadi 14 baris dan 12 kolom, (c) Citra digital hasil sampling berukuran 14 x 12 piksel (Sutoyo, dkk, 2009)

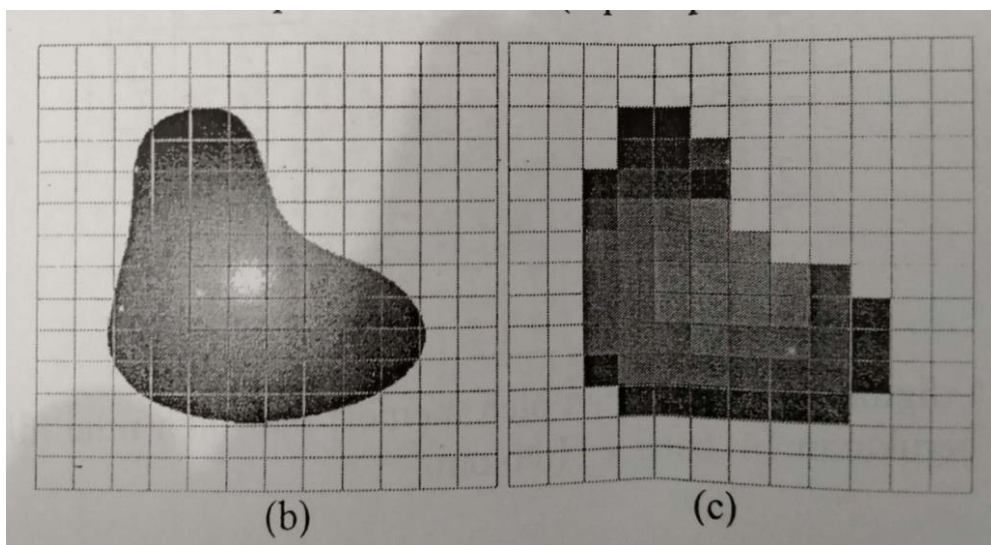
Gambar di atas merupakan proses sampling dari citra analog menjadi citra digital berukuran 14 x 12 piksel, yang mengakibatkan beberapa bagian informasi citra hilang (Sutoyo, Mulyanto, Suhartono, & Nurhayati, 2009).

### 2.4.2 Kuantisasi

Proses kuantisasi ialah proses pengubahan nilai amplitudo kontinu menjadi nilai baru berupa nilai diskrit. Nilai amplitudo terkuantisasi yaitu nilai pada

koordinat diskrit yang dihasilkan oleh proses pengambilan sampling (Putra, 2010).

Warna suatu citra digital ditentukan oleh intensitas piksel dalam komposisinya. Warna ini didapat dari besar kecilnya intensitas cahaya yang ditangkap oleh sensor. Skala intensitas cahaya (tingkat intensitas simulasi) di alam tidak terbatas, dan dapat menghasilkan warna yang tak terhingga jumlahnya. Sangat disayangkan, hingga saat ini, belum ada sensor yang mampu menangkap semua level warna tersebut (tidak mungkin). Pembatasan ini mengharuskan kita untuk membuat gradasi warna sesuai kebutuhan. Transformasi kontinu dari intensitas analog ke daerah intensitas diskrit disebut kuantisasi. Proses kuantisasi dihasilkan dari peralatan digital, seperti *scanner*, foto digital, serta kamera digital, seperti yang ditunjukkan pada Gambar 2.8 (b) dan (c).



**Gambar 2.8** Kuantisasi (Sutoyo, dkk, 2009)

Misalnya pada Gambar 2.8 menggunakan memori sebesar 3 bit untuk penyimpanan warna, maka gradasi warna yang dihasilkan pada citra analog ditunjukkan pada Gambar 2.8 (b) (harusnya jumlah warna yang dimiliki tidak terhingga). Warna pada tiap *pixel* disesuaikan menggunakan gradasi warna yang disediakan memori (Sutoyo, dkk, 2009).

## 2.5 Format File Citra

Dalam pengolahan citra ada dua macam format gambar yang sering dipakai, seperti *bitmap* dan *vector* (Sutoyo, dkk, 2009). Ada beberapa jenis format *file* citra yang dipakai pada saat ini. Format tersebut untuk menyimpan citra ke dalam *file*. Masing- masing format mempunyai karakteristik tersendiri (Putra, 2010). Contoh format citra yang umum dipakai, yaitu:

### 1. *Bitmap* (.bmp)

Format .bmp adalah format penyimpanan standar tanpa kompresi yang umum dapat digunakan untuk menyimpan citra biner hingga citra warna. Format ini terdiri dari beberapa jenis yang setiap jenisnya ditentukan dengan jumlah bit yang digunakan untuk menyimpan sebuah nilai *pixel*.

### 2. *Tagged Image Format* (.tif, .tiff)

Format .tif merupakan format yang dipakai dalam menyimpan citra bitmap sampai citra dengan warna palet terkompresi.

### 3. *Portable Network Graphics* (.png)

Format .png digunakan untuk menyimpan citra yang terkompresi yang bisa dipakai pada citra *grayscale*, citra dengan palet warna, serta citra *fullcolor*. Format ini dapat menyimpan informasi sebesar 1 sampai dengan 16 *bit* per kanal.

### 4. JPEG (.jpg)

jpg merupakan format yang dipakai untuk menyimpan citra hasil dari kompresi dengan metode JPEG.

### 5. MPEG (.mpg)

Format yang diperuntukkan untuk format penyimpanan citra video dan dipakai didunia internet.

### 6. *Graphics Interchange Format* (.gif)

Format ini bias dipakai pada citra warna dengan palet sebesar 8 *bit*. Umumnya dipakai pada aplikasi web.

Pada saat sekarang ini berbagai format *file* standar telah digunakan oleh hampir seluruh *Software Developer* yang mengembangkan aplikasi pengolahan citra. Dengan menggunakan format *file* standar dapat meningkatkan kemudahan warna citra dapat dipertukarkan dan memungkinkan dibaca oleh aplikasi lain dalam jangka yang panjang. Namun untuk kebanyakan proyek pengembangan aplikasi pengolahan citra, pemilihan format *file* yang tepat bukanlah hal yang mudah, akan tetapi perlu dipertimbangkan berulang kali (R & Sriani, 2016).

## 2.6 Segmentasi Citra

Segmentasi ialah proses membagi citra menjadi beberapa bagian dengan tujuan untuk memisahkan bagian objek dengan bagian latar belakang sehingga objek mudah untuk dianalisis (Sinaga, 2017).

Menurut (Sutoyo, dkk, 2009), segmentasi merupakan metode yang dipakai untuk mengubah gambar input menjadi gambar output dari atribut yang di ambil dari gambar tersebut. Segmentasi membagi citra pada daerah intensitasnya masing-masing sehingga objek dan latar belakang bisa dibedakan. Dalam hal ini pembagian tergantung dari masalah yang akan diselesaikan. Segmentasi tidak dilanjutkan jika masing-masing dari objek tersebut sudah terlihat jelas. Tingkat keakuratan segmentasi bergantung pada tingkat keberhasilan prosedur analisis yang dilakukan. Proses segmentasi diharapkan mempunyai tingkat keakuratan yang tinggi.

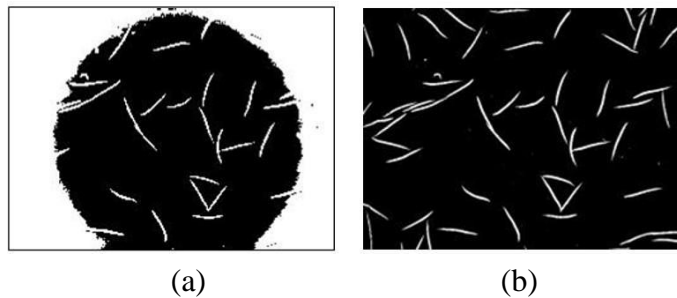
Beberapa pendekatan yang bisa dilakukan dalam segmentasi citra, yaitu:

1. Pendekatan batas, yaitu pendekatan yang dilakukan untuk mendapatkan batas antar daerah.
2. Pendekatan tepi, yaitu pendekatan yang dilakukan untuk menghubungkan piksel menjadi batas yang diinginkan dan mengidentifikasi piksel tepi.
3. Pendekatan daerah, tujuan pendekatan ini ialah untuk membagi citra pada daerahsehingga mendapatkan daerah yang diinginkan (Putranto, dkk, 2010).

## 2.7 Algoritma Local Adaptive Thresholding

Dalam *local adaptive thresholding*, nilai ambang batas berubah secara spasial dan ditentukan sesuai dengan konten lokal dari gambar yang digunakan. Perbedaan dari metode global ialah bahwa *local threshold* memiliki kinerja yang lebih baik untuk noise atau error yang disebabkan oleh informasi penting yang berdekatan dengan gambar teks atau objek gambar (Som, *et al.* 2011).

Pendekatan langsung pada *algoritma adaptive* ialah dengan membagi gambar menjadi beberapa bidang yang berukuran  $m \times m$ , dan kemudian memilih ambang batas  $T_{ij}$  untuk bagian gambar sesuai dengan histogram bagian ke- $ij$  ( $1 \leq i, j \leq m$ ). Hasil akhir dari proses ini yaitu kombinasi dari bagian-bagian citra yang sebenarnya berasal dari citra yang lebih besar (Ahmad, U. 2005). Gambar 2.9 memperlihatkan perbedaan hasil dari implementasi *thresholding* pada citra bergradasi.



**Gambar 2.9** Perbedaan global dengan *adaptive thresholding* (Zhou, *et al.* 2010)

Menurut Hermawati, F.A. (2013), langkah-langkah dalam proses *local adaptive thresholding* adalah sebagai berikut:

1. Masukkan sub gambar pertama yang berukuran  $m \times m$ .
2. Kemudian hitunglah *variance* dari sub gambar setelah melakukan perhitungan *mean*.

Rumus *Mean* Sub gambar :

$$\text{Mean}_{S[i]} = \bar{x}_i = \frac{\sum_{j=1}^n x_j}{n} \dots \dots \dots (2)$$

Rumus Varian Sub gambar


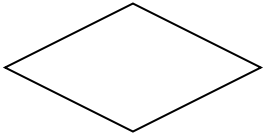
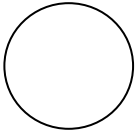
$$\text{Var [i]} = \frac{\sum_{j=1}^n (x_i - \bar{x}_i)^2}{n-1} \dots\dots\dots(3)$$

3. Jika  $var > 100$ , maka  $T = \frac{\max(subimg)}{2}$ , kemudian gunakan nilai  $T$  untuk melakukan *global thresholding*.
4. Jika  $var \leq 100$ , maka  $T = \frac{\max(img)}{2}$ , setelah itu gunakan langsung nilai  $T$  dalam menentukan output citra biner.
5. Ulangi langkah (1) pada sub gambar berikutnya hingga seluruh sub gambar selesai diproses.

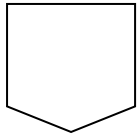
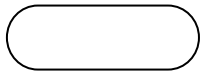
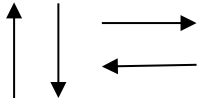


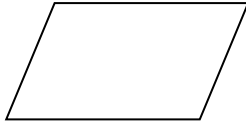
## 2.8 Flowchart

*Flowchart* ialah langkah-langkah yang dilakukan untuk menyelesaikan suatu masalah dengan rangkaian simbol-simbol grafis yang khusus (Nuraini, 2017). Menurut (Rahman & Julianto, 2015), *flowchart* adalah bagan yang menjelaskan tentang prosedur dalam menyelesaikan suatu masalah dari suatu program.

**Tabel 2.2** Simbol-simbol *flowchart* (Rahman & Julianto, 2015)

Simbol	Nama	Fungsi
	Proses Terdefinisi	Simbol untuk permulaan suatu sub program
	Keputusan	Simbol yang memberikan dua pilihan untuk langkahberikutnya, yaitu ya dan tidak
	Penghubung	Simbol untuk menghubungkan suatu proses ke proses lainnya pada halaman yang sama



	Penghubung	Simbol untuk menghubungkan suatu proses ke proses lainnya pada halaman yang berbeda
	<i>Terminator</i>	Simbol yang digunakan untuk menyatakan awal dan akhir suatu program
	Garis Alir	Simbol untuk menunjukkan arah arus dari sebuah proses
	Persiapan	Simbol untuk proses inisialisasi atau pemberian nilai awal
	Proses	Simbol untuk proses pengolahan data oleh computer
	<i>Input/Output</i>	Simbol untuk proses <i>input</i> dan <i>output</i> data

## 2.9 Visual Studio 2012

*Microsoft Visual Studio* adalah software lengkap yang bisa dipakai dalam melakukan pengembangan aplikasi, seperti aplikasi personal, aplikasi bisnis, ataupun komponen aplikasinya, dalam bentuk aplikasi *console*, aplikasi *Windows*, ataupun aplikasi *Web*. *Visual Studio* mencakup kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN

*Library*). Kompiler yang dimasukkan ke dalam paket *Visual Studio* antara lain *Visual C#*, *Visual C++*, *Visual Basic*, *Visual Basic.NET*, *Visual J++*, *Visual InterDev*, *Visual SourceSafe*, *Visual FoxPro*, dan *Visual J#*.

*Microsoft Visual Studio* bisa dipakai dalam mengembangkan aplikasi pada *native code* (dalam bentuk bahasa mesin yang berjalan diatas *Windows*) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* diatas *.NET Framework*). Selain itu, *Visual Studio* juga bisa dipakai untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan diatas *.NET Compact Framework*) (Wikipedia, 2020).

### 2.9.1 Bahasa Pemrograman C#

C# (*C Sharp*) ialah bahasa pemrograman berbasis objek yang digunakan oleh *Microsoft NET Framework*. Bahasa pemrograman ini merupakan bahasa pemrograman baru yang dikembangkan oleh *Microsoft* di bawah kepemimpinan Anders Hejlsberg, yang menciptakan berbagai bahasa pemrograman antara lain *Borland Turbo C++* dan *Orland Delphi*. Bahasa C# juga telah distandarisasi oleh ECMA. Seperti bahasa pemrograman lainnya, C# dapat digunakan untuk membangun berbagai jenis aplikasi, seperti aplikasi berbasis *windows (desktop)*, aplikasi berbasis *web* dan aplikasi berbasis *web services* (wikipedia, 2020).

### 2.10 Penelitian Terkait

(Alfiansyah & Fitriah Wulandari) dalam jurnalnya yang berjudul segmentasi pembuluh darah retina menggunakan *local adaptive thresholding* hasil penelitian menunjukkan bahwa nilai dari PNSR terhadap hasil segmentasi berada diatas rata-rata nilai 30, dimana nilai tersebut berada diatas nilai rata-rat standar. Hasil gambar segmentasi sangat jelas mampu membedakan antara pembuluh darah dan *background* meskipun masih banyak terdapat *noise* berupa lingkaran-lingkaran putih kecil.

(Ni Putu Sutramiani, dkk) dalam jurnalnya yang berjudul *Local Adaptive Thresholding Pada Preprocessing Citra Lontar Aksara Bali*. Dari percobaan dapat disimpulkan bahwa *window (W) = 70* dan *threshold (C) = 0,05* memperoleh hasil

yang paling besar, karena dibandingkan dengan percobaan lain noise masih dapat terlihat pada citra.

## BAB III

### METODE PENELITIAN

#### 3.1 Tempat dan Waktu Penelitian

##### 3.1.1 Tempat Penelitian

Tempat peneliti melakukan penelitian dalam skripsi ini adalah Museum Negeri Provinsi Sumatera Utara yang berada pada jalan H.M Jhoni No.51 Medan Kota, Kota Medan Sumatera Utara.

##### 3.1.2 Waktu dan Jadwal Pelaksanaan Penelitian

Waktu yang dibutuhkan peneliti dalam melaksanakan penelitian ini ialah sejak dikeluarkannya ijin penelitian dalam kurun waktu kurang lebih 3 bulan, yaitu 1 bulan pengumpulan data, 1 bulan analisis kebutuhan data, dan 1 bulan perancangan sistem serta penerapan. Jadwal dari tahap penelitian ini dapat dilihat pada tabel berikut ini :

**Tabel 3.1** Jadwal Pelaksanaan Penelitian

No	Keterangan	November		Desember				Januari		
		III	IV	I	II	III	IV	I	II	III
1	Pengumpulan Data	√	√							
2	Analisis Kebutuhan			√	√	√				
3	Perancangan Sistem						√	√	√	
4	Penerapan									√

### 3.2 Bahan dan Alat Penelitian

Analisis kebutuhan bahan dan alat penelitian digunakan untuk menentukan alat yang dibutuhkan untuk membuat aplikasi segmentasi citra kuno (termasuk perangkat keras dan perangkat lunak). Melalui penggunaan analisis kebutuhan sistem, dapat melihat persyaratan minimum yang diperlukan untuk membuat suatu aplikasi. Berikut ini menjelaskan spesifikasi perangkat keras dan perangkat lunak yang diperlukan untuk aplikasi sistem.

#### 3.2.1 Perangkat Keras

Perangkat keras yang digunakan peneliti dalam merancang dan membangun aplikasi segmentasi citra naskah kuno adalah sebagai berikut:

**Tabel 3.2** Kebutuhan *Hardware*

Nama Komponen	Spesifikasi
<i>Procesor</i>	Intel(R) Coie i3 3,0 Ghz
<i>Memory</i>	4 GB
<i>Harddisk</i>	500 GB

#### 3.2.2 Perangkat Lunak

Perangkat lunak (*software*) yang digunakan peneliti dalam merancang dan membangun aplikasi segmentasi naskah kuno adalah sebagai berikut:

1. Aplikasi Camera
2. *Microsoft Visual Studio 2012*
3. *Microsoft Visio 2010*

### 3.3 Cara Kerja Penelitian

Cara kerja dalam penelitian ini terdiri dari beberapa sub menu antara lain sebagai berikut:

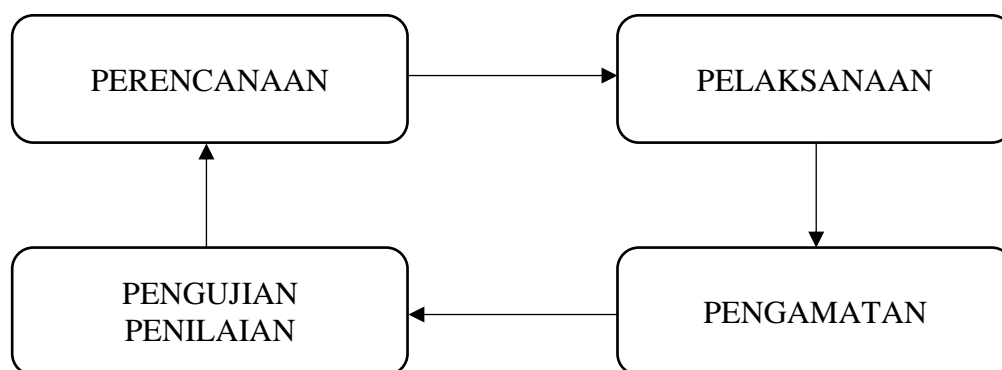
#### 3.3.1 Metode Penelitian

Metode yang digunakan peneliti dalam melakukan penelitian ini ialah penelitian tindakan (*action research*). Penelitian tindakan mengacu pada kegiatan atau tindakan uji coba suatu ide kedalam praktek atau situasi yang nyata dalam skala

mikro, dalam kegiatan tersebut diharapkan dapat memperbaiki, meningkatkan kualitas, dan melakukan perbaikan sosial.

Tidak hanya itu, penelitian tindakan bisa dikatakan memiliki tujuan meningkatkan keterampilan- keterampilan baru ataupun metode pendekatan baru serta untuk menyelesaikan permasalahan dengan pelaksanaan secara langsung di dunia kerja. Dari penafsiran serta tujuan tersebut peneliti merasa sesuai memakai tata cara penelitian tindakan ataupun action research guna menunjang tercapainya tujuan peneliti.

Secara garis besar, langkah- langkah dalam penelitian kegiatan tersebut yaitu perencanaan (*planning*), pelaksanaan (*acting*), pengamatan (*monitoring*), pengujian serta penilaian. Keempat langkah tersebut bisa dilihat dari bagan berikut ini:



**Gambar 3.1** Bagan Langkah Penelitian Tindakan

Dari gambar diatas, kita bisa mengetahui bahwa langkah- langkah tersebut bisa jadi satu siklus. Yang maksudnya, siklus dari keempat langkah tersebut bisa berulang. Apabila peneliti telah merasa puas dengan hasil yang dicapainya, maka siklus bisa berhenti.

Adapun langkah-langkah pokok dalam penelitian tindakan pada penelitian ini adalah sebagai berikut:

1. Rumusan permasalahan serta tujuan riset bersama- sama antara periset, pekerja praktis serta decision maker.
2. Himpun informasi yang ada tentang hal- hal yang berhubungan dengan permasalahan maupun metode- metode dengan melaksanakan riset kepustakaan.
3. Rumuskan hipotesa dan strategi pendekatan dalam menyelesaikan masalah.

4. Buat desain riset bersama- sama antara periset dengan pelaksana program dan rumuskan prosedur, perlengkapan serta keadaan pada riset tersebut hendak dilaksanakan.
5. Tentukan kriteria penilaian, metode pengukuran, dan teknik- teknik analisa yang digunakan.
6. Kumpulkan informasi, analisa, beri interpretasi dan generalisasi serta saran-saran
7. Laporkan riset dengan penulisan ilmiah

Adapun bagan dari keseluruhan metode penelitian dalam skripsi ini adalah sebagai berikut :

### **3.3.2 Desain Penelitian**

Metode penelitian yang digunakan oleh peneliti adalah metode penelitian deskriptif dan komparatif. Penelitian deskriptif ialah riset terhadap fenomena ataupun populasi tertentu yang didapat oleh periset dari subjek berbentuk individu, organisasional, industri ataupun perspektif lainnya. Penelitian deskriptif dilakukan guna menanggapi persoalan tentang: apa, siapa, kapan, di mana, serta bagaimana yang berkaitan dengan ciri populasi ataupun fenomena tersebut. Penelitian ini juga dilakukan dengan memakai tata cara komparatif yang dicoba dengan menyamakan teori yang terdapat pada praktek yang ditemui serta menarik kesimpulan.

Untuk mempermudah dalam melaksanakan suatu penelitian yang berhubungan dengan permasalahan yang diteliti, maka peneliti melaksanakan pendekatan riset permasalahan. Dengan memakai pendekatan ini, informasi yang dikumpulkan bisa disesuaikan dengan kondisi yang sesungguhnya serta dibandingkan dengan teori yang mendukung. Dengan demikian, bisa memberikan gambaran yang lumayan jelas dan bisa menarik kesimpulan dari objek yang diteliti.

### **3.3.3 Teknik Pengumpulan Data**

Peneliti menggunakan jenis dan metode pengumpulan data guna mendapatkan data untuk bahan kajian dalam penulisan skripsi ini yang bertujuan untuk membuat suatu perancangan aplikasi perbaikan citra naskah kuno di Museum

Negeri Provinsi Sumatera Utara. Dalam hal ini, peneliti menggunakan metode pengumpulan data yang berupa sumber data primer (observasi, wawancara serta pengamatan sistem) dan sumber data sekunder (dokumentasi).

#### 1. Sumber Data Primer

Sumber data primer ialah informasi yang diperoleh secara langsung dari sumber informasi tersebut yang berhubungan dengan riset yang dilakukan, yakni data-data yang diperoleh dari wawancara serta survei ataupun pengamatan langsung, yang dipakai sebagai bahan acuan dalam pembuatan suatu aplikasi. Contoh data primer yang diperlukan peneliti guna mendukung pembuatan aplikasi ialah data naskah kuno yang difoto ataupun di *scan* di Museum Negeri Provinsi Sumatera Utara untuk dilakukan pengamatan citra hasil *scannya*.

#### 2. Sumber Data Sekunder

Informasi yang diperoleh dari informasi peneliti dalam bentuk yang telah jadi yang bersifat data serta kutipan, baik dari internet ataupun literatur, pustaka, jurnal yang berhubungan dengan riset yang dibuat. Contoh data sekunder yang diperlukan peneliti yaitu bagaimana menjalankan *algoritma adaptive thresholding* guna memperjelas citra tulisan naskah kuno yang sudah sulit untuk dibaca.

### 3.3.4 Analisa Kebutuhan

Aplikasi yang hendak dirancangan membutuhkan beberapa kebutuhan sistem minimum supaya aplikasi bisa berjalan dengan sebagaimana mestinya. Adapun kebutuhan sistem secara minimum dalam aplikasi segmentasi naskah kuno pada pembahasan ini dibagi jadi 2 bagian yakni, kebutuhan minimum perangkat keras serta kebutuhan minimum perangkat lunak.

#### 1. Kebutuhan Minimum Perangkat Keras

Untuk menjalankan suatu aplikasi membutuhkan perangkat keras yang memiliki spesifikasi minimal. Hal ini dilakukan dengan tujuan untuk mengoptimalkan jalannya suatu aplikasi. Adapun kebutuhan minimal perangkat keras yaitu sebagai berikut:



- a. *Processor Intel Inside Dual Core 3.0Ghz*
- b. RAM 1 GB DDR3
- c. *Harddisk SATA/IDE* dengan minimal kapasitas 100 GB
- d. Monitor
- e. *Keyboard dan Mouse*

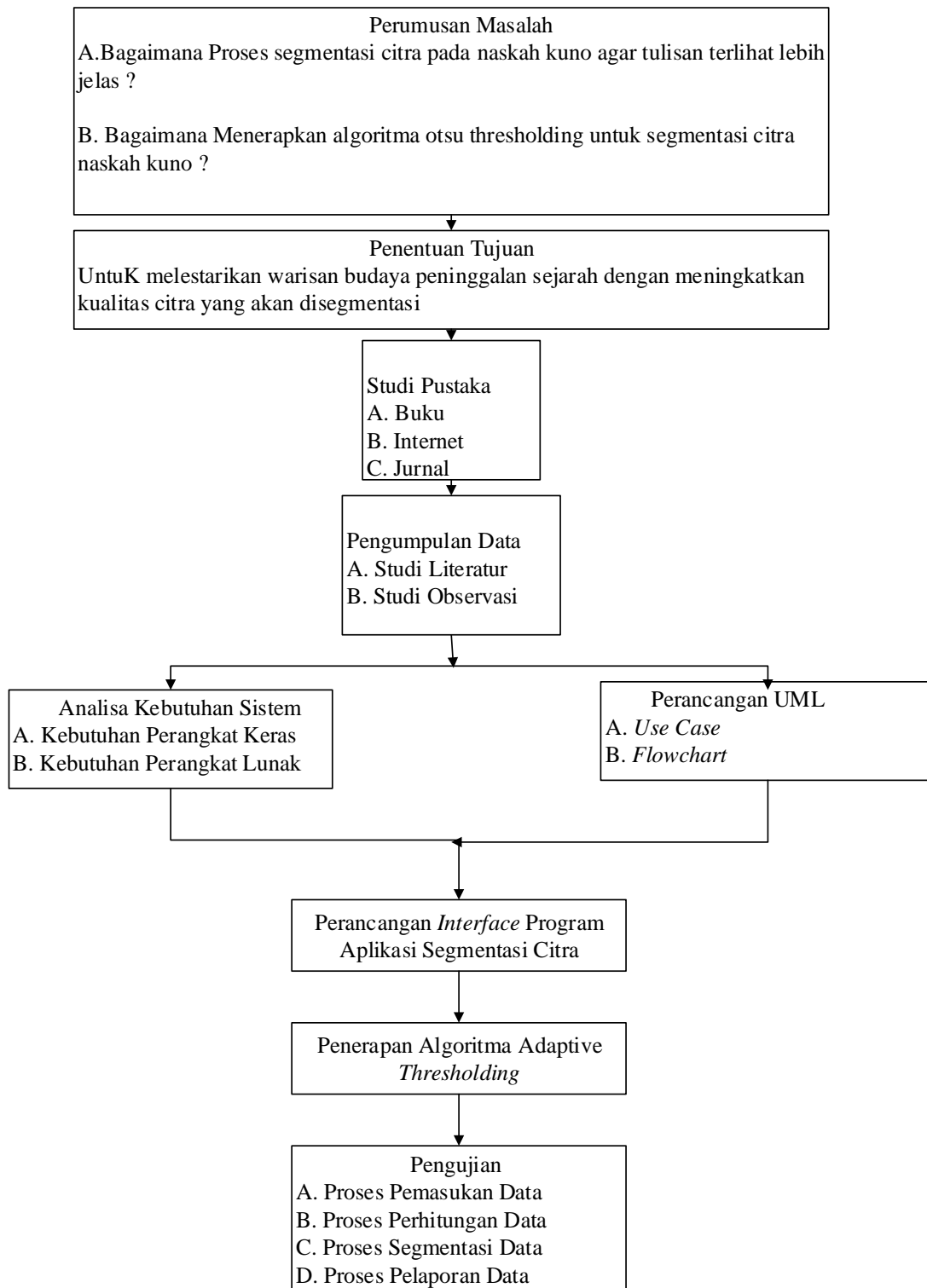
## 2. Kebutuhan Perangkat Lunak

Perangkat lunak berperan penting untuk membantu serta mengoptimalkan jalannya suatu aplikasi. Adapun perangkat lunak minimal yang harus dimiliki didalam perangkat keras komputer dalam menjalankan aplikasi ini adalah sebagai berikut:

- a. Sistem Operasi Windows 7
- b. *Microsoft Visual Studio 2012*

### **3.3.5 Perancangan Sistem**

Agar penelitian semakin terarah dan sinkron dengan tujuan yang ingin dicapai, maka dibutuhkan sebuah rancangan yang dipakai sebagai panduan pada penelitian, menggunakan tahapan yang akan penulis gambarkan menggunakan bagan sebagai berikut:



**Gambar 3.2** Bagan Rancangan Penelitian

Berikut peneliti akan menjelaskan rincian tentang rancangan penelitian yang penulis gunakan.

#### 1. Studi Pustaka

Peneliti mendapatkan data skunder dari hasil studi pustaka. Tahap yang dilakukan yaitu dengan mempelajari referensi dari buku-buku ataupun sumber-sumber yang sesuai dalam skripsi ini, baik dari *text book* ataupun internet.

#### 2. Studi Lapangan

Data primer dikumpulkan pada tahap ini. Proses pengumpulan data dilakukan dengan mewawancarai pihak yang berwenang di Museum Negeri Provinsi Sumatera Utara serta melakukan survei.

#### 3. Analisis Data

Ketika seluruh data sudah terkumpul, maka tahap berikutnya yakni melakukan proses analisis data. Pada tahap ini proses analisis yang dilakukan terdiri dari 2 hal, yakni yang pertama analisis data yang diperoleh serta analisis kebutuhan, dan yang kedua yaitu definisi berdasarkan kebutuhan tersebut.

#### 4. Perancangan Sistem

Peneliti memakai alat bantu untuk perancangan yakni UML berupa *Use Case Diagram*, *Activity Diagram* serta *Flowchart* dalam melakukan perancangan sistem.

#### 5. Implementasi

Pada tahapan ini mulai dilakukan penyusunan dan perhitungan algoritma *Adaptive Thresholding* dalam melakukan segmentasi citra naskah kuno agar dapat terlihat lebih jelas dan lebih baik dari pada sebelumnya.

#### 6. Pengujian Sistem (Testing)

Setelah selesai dilakukan seluruh proses implementasi, maka tahap berikutnya yaitu pengujian dengan tujuan untuk menguji kebenaran perhitungan algoritma *Adaptive Thresholding* dalam melakukan segmentasi citra naskah kuno agar dapat terlihat lebih jelas dan lebih baik dari pada sebelumnya

#### 7. Pembuatan Laporan

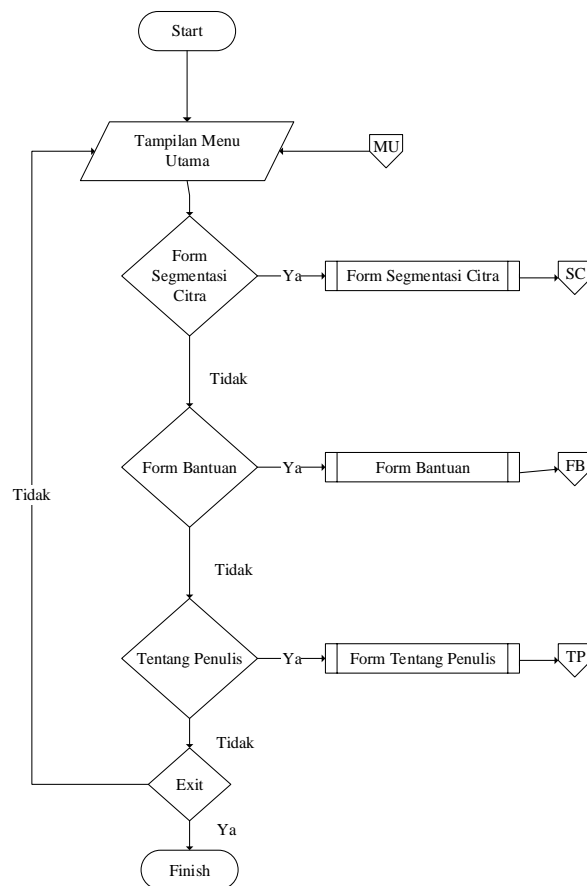
Dan yang terakhir yakni tahapan pembuatan laporan dengan tujuan dapat dijadikan sebagai dokumentasi hasil penelitian.

### 3.3.6 Rancangan Flowchart Penelitian

*Flowchart* sistem merupakan bagian-bagian yang memiliki arus yang menggambarkan langkah-langkah dalam penyelesaian suatu permasalahan. Adapun langkah-langkahnya dapat dilihat pada gambaran *flowchart*.

#### 1. *Flowchart* Sistem Keseluruhan

*Flowchart* Sistem aplikasi keseluruhan menjelaskan tentang alur yang akan dirancang dengan antarmuka tampilan *form* pertama yang muncul ketika aplikasi dibuka oleh *user*. Adapun alur-alur dari sistem aplikasi keseluruhan dapat dijelaskan pada gambar *flowchart* dibawah ini :



**Gambar 3.3** *Flowchart* Sistem

Berdasarkan pada gambar *Flowchart* Sistem diatas bias dijelaskan seperti berikut ini:

#### a. Tampilan Menu Utama

Form menu utama ialah *form* yang muncul pertama kali pada saat pengguna membuka aplikasi segmentasi citra naskah kuno.

b. *Form Segmentasi Citra*

*Form* segmentasi citra yang berfungsi untuk melakukan proses segmentasi citra naskah kuno yang diinputkan dengan menggunakan algoritma *Adaptive Thresholding*.

c. *Form Bantuan*

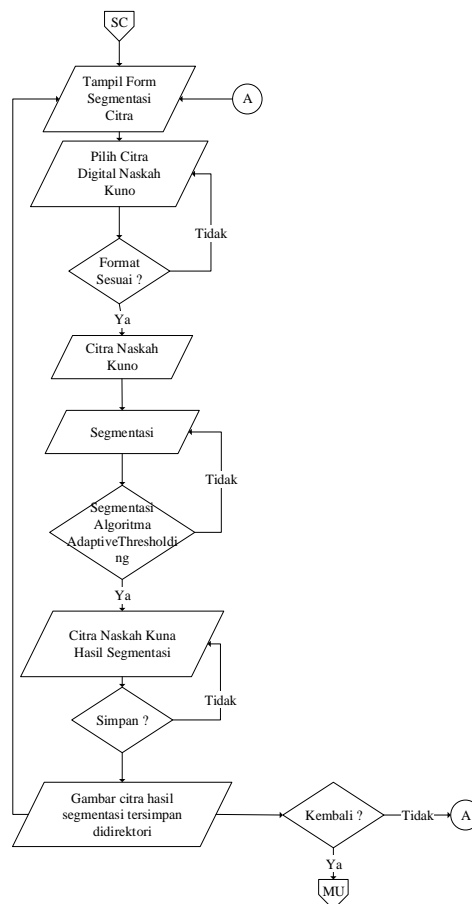
*Form* bantuan yang berfungsi untuk menampilkan informasi aplikasi segmentasi citra naskah kuno.

d. *Form Tentang Penulis*

*Form* tentang penulis fungsinya untuk menampilkan informasi dari penulis pada penelitian ini.

2. *Flowchart Segmentasi*

*Flowchart* proses segmentasi pada aplikasi yang akan dibangun menggunakan algoritma *Adaptive Thresholding* keseluruhan dapat dilihat pada gambar berikut :



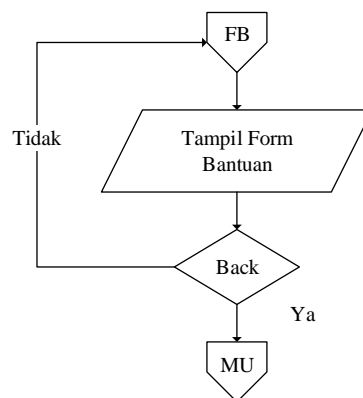
**Gambar 3.4** *Flowchart* Segmentasi

Pada gambar diatas, proses segmentasi bisa dijelaskan sebagai berikut ini:

- 1) Pilih citra naskah kuno yang sudah tersimpan didalam direktori komputer dengan format yang telah ditentukan yaitu .jpg.
- 2) Melakukan proses segmentasi citra naskah kuno menggunakan algoritma *Adaptive Thresholding*.
- 3) Menampilkan hasil segmentasi citra naskah kuno dan menyimpan kedalam direktori komputer.

### 3. *Flowchart Form Bantuan*

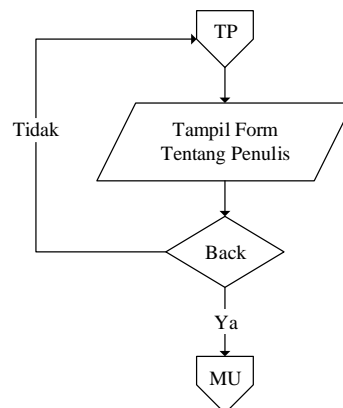
*Flowchart form* bantuan akan menjelaskan aliran kegiatan yang dapat dilakukan oleh *user* pada saat mengakses *form* aplikasi ini. Berikut *flowchart* cara pemakaian :



**Gambar 3.5** *Flowchart Form Bantuan*

### 4. *Flowchart Tentang Penulis*

*Flowchart* tentang penulis akan menampilkan data perancang aplikasi yang dibangun. Berikut *flowchart* tentang penulis:



**Gambar 3.6** *Flowchart Tentang Penulis*

## BAB IV PERANCANGAN DAN HASIL

### 4.1 Rancangan Interface

Tujuan perancangan interface yakni guna merancang aplikasi antar muka yang akan dibangun dalam sebuah perangkat lunak, sehingga mempermudah dalam pembuatan aplikasi serta lebih mudah dipahami. Berikut ini merupakan bentuk rancangan sistem yang nantinya akan diimplemetasikan kedalam sebuah aplikasi.

#### 1. Rancangan Menu Utama

Menu utama ialah *form* yang tampil pertama kali pada saat program aplikasi dibuka. Adapun tampilan dari rancangan *form* menu utama sebagai berikut:

Menu Utama X			
Segmentasi Citra	Bantuan	Tentang Penulis	Exit
a	b	c	d
SEGMENTASI CITRA NASKAH KUNO LOCAL ADAPTIVE THRESHOLDING			
<div style="display: flex; justify-content: center; align-items: center;"><div style="border: 1px solid black; border-radius: 50%; width: 100px; height: 100px; display: flex; align-items: center; justify-content: center; margin-right: 10px;">Logo</div><div style="border: 1px solid black; border-radius: 50%; width: 30px; height: 30px; display: flex; align-items: center; justify-content: center;">e</div></div>			
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA PROGRAM STUDI ILMU KOMPUTER 2020			

**Gambar 4.1** Rancangan Menu Utama

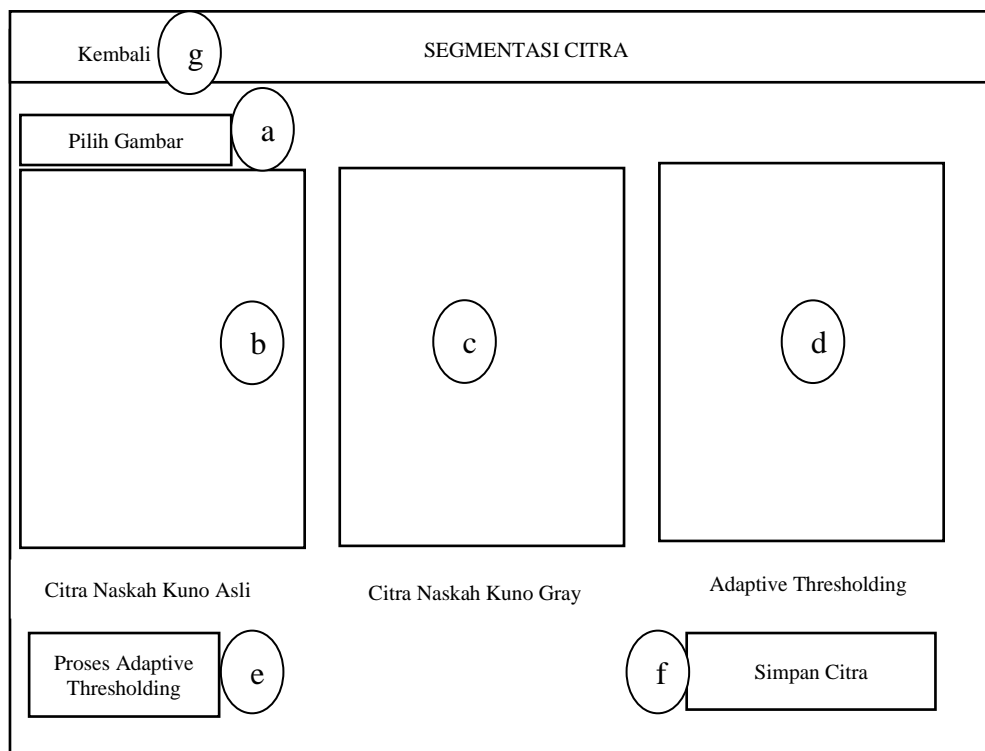
Berdasarkan gambar 4.1 diatas bisa dijelaskan keterangan sebagai berikut:

- a. Menu yang menampilkan *form* untuk melakukan proses segmentasi citra naskah kuno.
- b. Menu yang menampilkan *form* bantuan dan informasi aplikasi.
- c. Menu yang menampilkan *form* tentang penulis dan perancang aplikasi.

- d. Menu untuk menutup aplikasi.
- e. Logogambar Universitas Islam Sumatera Utara

## 2. Rancangan Segmentasi Citra

*Form* segmentasi citra fungsinya sebagai *interface* bagi pengguna aplikasi ketika melakukan suatu kegiatan dari segmentasi citra naskah kuno. Melalui *form* segmentasi citra *user* memilih citra naskah kuno yang ingin disegmentasi dan user dapat menyimpan hasil segmentasi citra naskah kuno. Berikut rancangan *interface form* segmentasi citra bisa dilihat pada gambar berikut :



**Gambar 4.2** Rancangan *Form* Segmentasi Citra

Berdasarkan gambar 4.2 diatas bisa dijelaskan keterangan sebagai berikut:

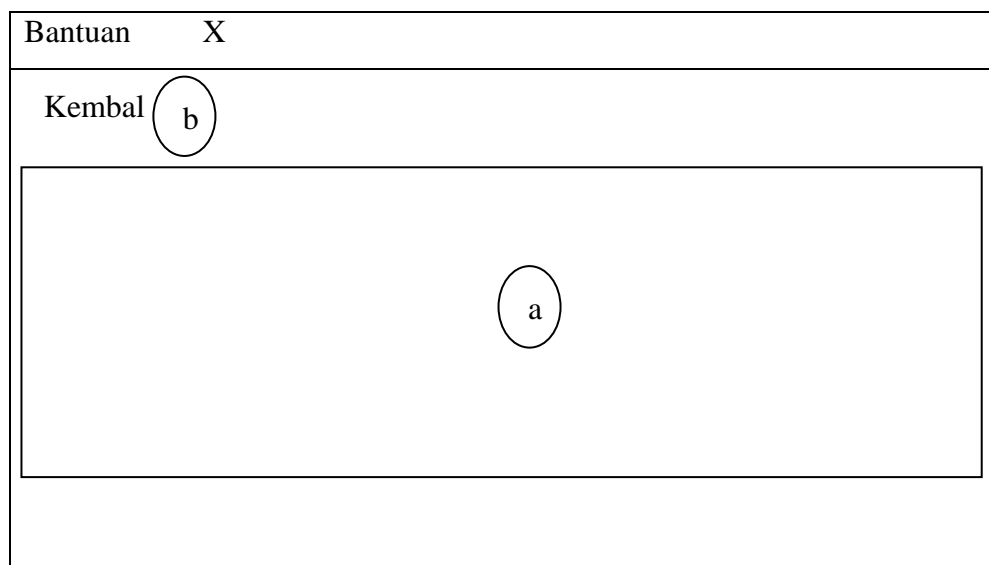
- a. *Button* untuk memilih citra naskah kuno dari media penyimpanan.
- b. *PictureBox* untuk menampilkan citra naskah kuno asli sebelum disegmenta.
- c. *PictureBox* untuk menampilkan citra naskah kuno gray.
- d. *PictureBox* untuk menampilkan citra naskah kuno setelah disegmentasi menggunakan *local adaptive thresholding*.
- e. *Button* untuk memproses segmentasi citra naskah kuno.



- f. *Button* untuk menyimpan citra naskah kuno hasil segmentasi *adaptive thresholding*.
- g. *Stripmenu* untuk kembali pada menu utama.

### 3. Rancangan *Form* Bantuan

*Form* bantuan fungsinya untuk memunculkan rincian bantuan untuk cara pemakaian dan informasi pada aplikasi. Berikut ini gambar rancangan *interface* dari *form* bantuan:



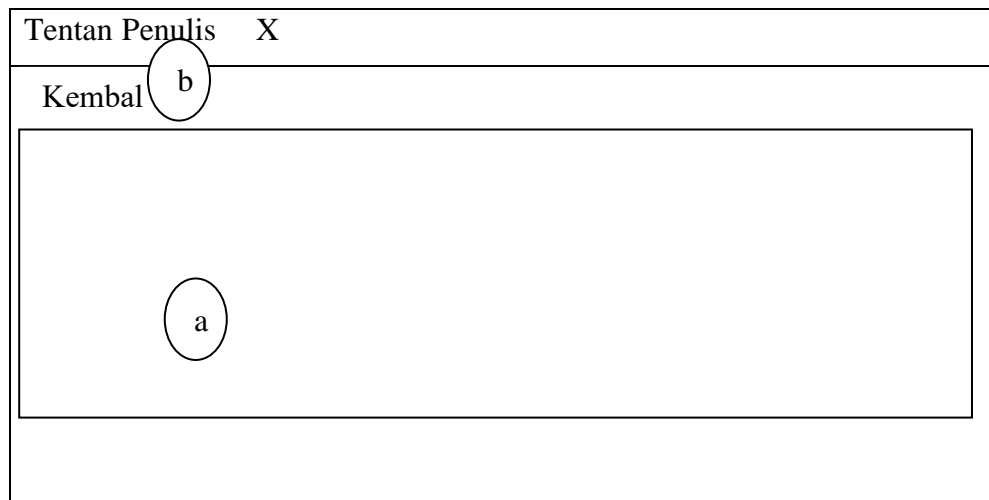
**Gambar 4.3** Rancangan *Form* Bantuan

Berdasarkan gambar 4.3 diatas bisa dijelaskan keterangan sebagai berikut:

- a. *RichTextBox* yaitu menampilkan informasi dari cara pemakaian aplikasi.
- b. *Strip Menu* untuk kembali ke menu utama aplikasi.

### 4. Rancangan *Form* Tentang Penulis

*Form* tentang penulis fungsinya yakni menampilkan informasi tentang penulis dan perancangan aplikasi. Berikut ini gambar rancangan *interface* pada *form* tentang penulis:



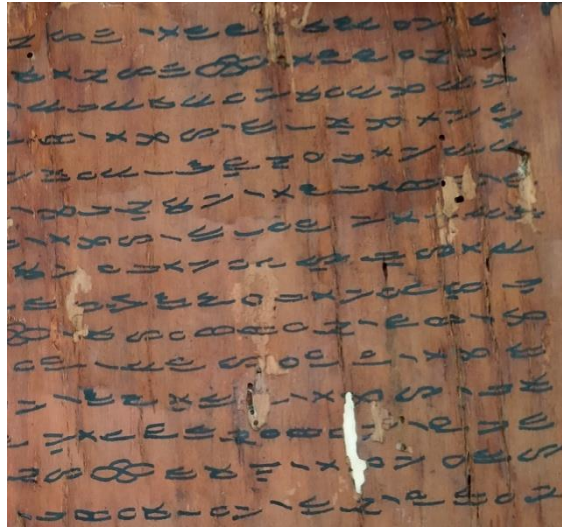
**Gambar 4.4** Rancangan *Form* Tentang Penulis

Berdasarkan gambar 4.4 diatas bisa dijelaskan keterangan sebagai berikut:

- a. *RichTextBox* yakni menampilkan informasi data diri perancang aplikasi.
- b. *Strip Menu* untuk kembali ke menu utama aplikasi.

#### **4.2 Pengujian *Algoritma Local Adaptive Thresholding***

Pada pengujian ini tahapan segmentasi citra naskah kuno memakai *Algoritma Local Adaptive Thresholding*, alur proses dari awal hingga akhir akan dijelaskan sampai menghasilkan citra segmentasi naskah kuno. Pertama, perhitungan akan dilakukan secara manual untuk mengubah citra naskah kuno RGB menjadi keabuan dengan memakai metode luminosity grayscale. Kemudian yang kedua, dilakukan perhitungan lagi untuk mengubah citra grayscale menjadi citra biner menggunakan *Algoritma Local Adaptive Thresholding*. Adapun contoh sampel naskah kuno yang akan dilakukan perhitungan manual adalah sebagai berikut:



**Gambar 4.5.** Sampel Citra Naskah Kuno

Berdasarkan dari gambar diatas, sampel citra naskah kuno berformat ,jpg dengan resolusi 256 x 256 pixel. Selanjutnya adalah melakukan konversi citra kedalam bentuk *grayscale*. Perhitungan mengkonversi citra RGB menjadi citra *grayscale* secara manual dilakukan pada ukuran matriks 3 x 3 pada pixel yang bias dilihat pada Tabel 4.1.

**Tabel 4.1** Nilai pixel 3 x 3 Sampel Naskah Kuno

x\y	0	1	2
0	150 251 64	255 130 5	130 0 66
1	0 64 112	120 30 201	76 105 111
2	190 172 98	99 125 145	170 140 53

*Luminosity grayscale* dapat dihitung dengan memakai persamaan seperti berikut:

$$G(x, y) = (0.21 \times R) + (0.71 \times G) + (0.07 \times B)$$

Dari persamaan diatas, maka perhitungan untuk matriks (0,0) hingga matriks (2,2) yaitu:

$$G(0,0) = (0.21 \times 150) + (0.71 \times 251) + (0.07 \times 64) = 214,19$$

$$G(0,1) = (0.21 \times 0) + (0.71 \times 64) + (0.07 \times 112) = 53,28$$

$$G(0,2) = (0.21 \times 190) + (0.71 \times 172) + (0.07 \times 98) = 168,88$$

$$G(1,0) = (0.21 \times 255) + (0.71 \times 130) + (0.07 \times 5) = 146,2$$

$$G(1,1) = (0.21 \times 120) + (0.71 \times 30) + (0.07 \times 201) = 60$$

$$G(1,2) = (0.21 \times 99) + (0.71 \times 125) + (0.07 \times 145) = 119,69$$

$$G(2,0) = (0.21 \times 130) + (0.71 \times 0) + (0.07 \times 66) = 31,92$$

$$G(2,1) = (0.21 \times 76) + (0.71 \times 105) + (0.07 \times 111) = 98,28$$

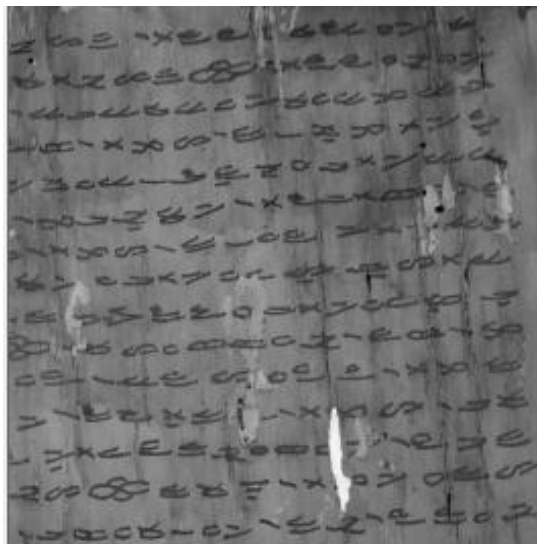
$$G(2,2) = (0.21 \times 170) + (0.71 \times 140) + (0.07 \times 53) = 138,81$$

Dari perhitungan sebelumnya, maka mendapatkan hasil nilai pixel 3 x 3 tipe *grayscale* yang sudah dibulatkan seperti pada tabel 4.2.

**Tabel 4.2** Hasil Nilai Pixel 3 x 3 *Grayscale* Sampel Naskah Kuno Citra

x\y	0	1	2
0	214	146	32
1	53	60	98
2	169	120	139

Adapun hasil pengujian dari metode *luminosity grayscale* pada citra naskah kuno bisa dilihat pada gambar berikut ini:



**Gambar 4.6.** Hasil *Grayscale* Sampel Citra Naskah Kuno

Citra naskah kuno yang sudah melalui proses *grayscale* berikutnya akan dihitung dengan memakai *Algoritma Adaptive Thresholding*. Citra naskah kuno dibagi menjadi beberapa blok, setelah itu dihitung secara adaptif dari blok pertama hingga blok terakhir sesuai dengan nilai *threshold* masing-masing. Selanjutnya adalah menghitung nilai mean (rata-rata) intensitas citra dari tiap sub citra dengan persamaan citra sebagai berikut:

$$\text{Means}_{[i]} = \bar{x}_i = \frac{\sum_{j=1}^n x_j}{n}$$

$n$  adalah jumlah sub citra. Berdasarkan pada contoh citra sampel 3 x 3 maka perhitungan *mean* sub citra akan diproses hingga sub citra ke-3. Penjumlahan dimulai dari *pixel* pertama ( $j = 1$ ) hingga *pixel* ke 3 x 3 ( $n$ ), selanjutnya dibagi dengan  $n$ . Berikut ini adalah perhitungan yang dilakukan secara manual untuk kelanjutan dari kasus uji coba citra naskah kuno dari subcitra pertama.

Means

$$X_0 = (0,0) - (0,2) = \frac{214 + 53 + 169}{3} = 145$$

$$X_1 = (1,0) - (1,2) = \frac{146 + 60 + 98}{3} = 101$$

$$X_2 = (2,0) - (2,2) = \frac{32 + 98 + 139}{3} = 90$$

Hasil dari nilai *mean* yang didapat dari sub citra pertama ialah 145, sub citra kedua adalah 101 dan sub citra terakhir adalah 90 sesuai dengan sampel citra 3 x 3. Langkah berikutnya yakni menghitung *variance* dari sub citra tersebut. *Variance* dipakai sebagai nilai batas intensitas *pixel* untuk menentukan *output* citra biner. Berikut adalah persamaan untuk menghitung *variance*:

$$\text{var}_{[i]} = \frac{\sum_{j=1}^n (x_j - \bar{x}_i)^2}{n - 1}$$

Rumus diatas menyatakan penjumlahan untuk tiap *pixel* dari sub citra ke-*i* yang dikurangi dengan hasil nilai mean, selanjutnya hasil penjumlahan dibagi dengan total blok *pixel* sub citra dikurang 1. Nilai *variance* yang diperoleh akan masuk ke *statement* kondisi dengan persamaan:

$$var_{[i]} > 100 \text{ atau } var_{[i]} \leq 100$$

Perhitungan yang dilakukan secara manual untuk dua nilai *variance*. Berikut perhitungan pada kasus uji citra naskah kuno:

$$Var_0 = (0, 0) - (0, 2) = Var = \frac{(214 - 145)^2 + (53 - 145)^2 + (169 - 145)^2}{9-1}$$

$$Var = \frac{26,887 + 8,464 + 576}{8} = 4,490.875$$

$$Var_1 = (1, 0) - (1, 2) = Var = \frac{(146 - 101)^2 + (60 - 101)^2 + (120 - 101)^2}{9-1}$$

$$Var = \frac{2,025 + 574 + 361}{8} = 370$$

$$Var_2 = (2, 0) - (2, 2) = Var = \frac{(32 - 90)^2 + (98 - 90)^2 + (139 - 90)^2}{9-1}$$

$$Var = \frac{3,364 + 64 + 2,401}{8} = 5,829$$

Dari hasil perhitungan manual yang sudah dilakukan, nilai *variance* menunjukkan  $> 100$ . Pada citra uji naskah kuno ini, tidak ada ditemukannya nilai *variance* dibawah angka 100. Kesimpulannya adalah pengubahan citra biner lebih efektif dilakukan dengan intensitas nilai yang ada pada tiap-tiap sub citra. Sehingga selanjutnya adalah dilakukan Thresolding secara adaptif dengan rumus

$$(Var_{[i]} > 100) \rightarrow T_i = \frac{bloksub[i]}{2}$$

Jika nilai *variance*  $> 100$ , maka dilakukan perhitungan nilai batas *T* (*threshold*) untuk tiap sub citra ke-*i* dengan terlebih dahulu mencari nilai intensitas *pixel* dari setiap blok (*bloksub*). Adapun nilai *T* untuk sub citra ke-*i* sebagai berikut:

$$\text{Blok } (0,0) = \frac{214}{2} = 107$$

$$\text{Blok (0,1)} = \frac{53}{2} = 27$$

$$\text{Blok (0,2)} = \frac{169}{2} = 85$$

$$\text{Blok (1,0)} = \frac{146}{2} = 73$$

$$\text{Blok (1,1)} = \frac{60}{2} = 30$$

$$\text{Blok (1,2)} = \frac{120}{2} = 60$$

$$\text{Blok (2,0)} = \frac{32}{2} = 16$$

$$\text{Blok (2,1)} = \frac{98}{2} = 49$$

$$\text{Blok (2,2)} = \frac{139}{2} = 70$$

**Tabel 4.3** Hasil Nilai Batas Threshold

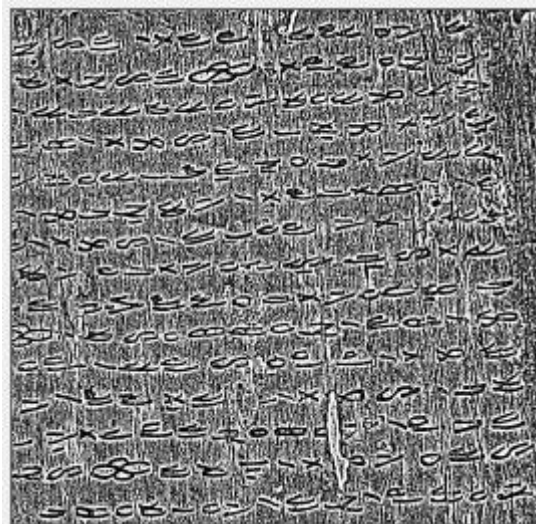
x\y	0	1	2
0	107	73	16
1	27	30	49
2	85	60	70

Setelah mengetahui nilai batas *threshold* ( $T$ ), berikutnya yaitu menentukan hasil *pixel* citra biner (0 atau 1) dengan cara membandingkan nilai intensitas *pixel* dan nilai batas  $T$ . Apabila nilai batas threshold lebih besar dari nilai *pixel* tersebut, maka *pixel* citra akan dikonversi menjadi 0 (hitam). Sebaliknya, apabila nilai batas threshold lebih kecil dari nilai *pixel*, maka *pixel* citra akan dikonversi menjadi 1 (putih). Berikut *output pixel* citra yang diuji dalam citra sampel 3 x 3:

**Tabel 4.4** Hasil nilai biner pixel 3 x 3 naskah kuno

x\y	0	1	2
0	1	0	0
1	0	0	0
2	0	0	0

Berdasarkan dari tabel diatas, dapat terlihat dengan jelas pemisahan antara *background* dengan objek citra naskah kuno. Berhubungan *citra* sampel yang dihitung dalam perhitungan manual hanya 3 x 3 maka tidak terlalu terlihat pembagian secara merata. Adapun hasil ilustrasi dari keseluruhan citra naskah kuno hasil *thresolding* secara adaptif bisa dilihat pada gambar berikut ini:

**Gambar 4.7** Hasil *Adaptive Thresholding* Sampel Citra Naskah Kuno

### 4.3 Hasil Rancangan Interface Pada Aplikasi

Berdasarkan dari perancangan program aplikasi segmentasi citra naskah kuno, berikut adalah hasil dari proses perancangan program aplikasi pada *software Microsoft Visual Studio 2012* yang terdiri dari *interface* menu utama, menu segmentasi citra, menu bantuan dan menu tentang penulis. Adapun menu-menu tersebut dapat dilihat pada gambar dibawah ini:



## 1. Menu Utama

Menu utama ialah menu yang tampil pertama kali ketika user membuka aplikasi segmentasi citra naskah kuno. Berikut tampilan dari menu utama yang sudah diterapkan kedalam aplikasi:



**Gambar 4.8** Menu Utama Aplikasi

Berdasarkan dari gambar diatas, menu utama mempunyai tampilan yang terdiri dari beberapa *strip* menu yaitu *strip* menu segmentasi citra, *strip* menu bantuan dan *strip* menu tentang penulis serta *strip* menu *exit*.

## 2. Menu Segmentasi Citra

Menu segmentasi citra adalah yang digunakan untuk melakukan suatu proses segmentasi citra naskah kuno menggunakan metode *local adaptive thresholding*. Adapun tampilan dari menu segmentasi citra pada aplikasi sebagai berikut:

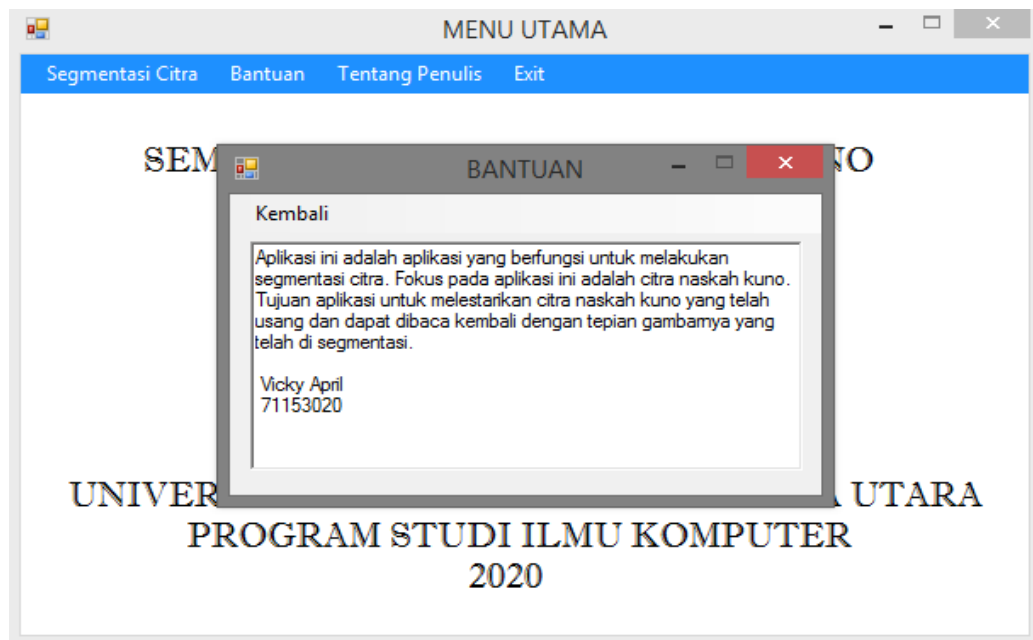


**Gambar 4.9** Menu Segmentasi Citra

Berdasarkan dari gambar 4.8, menu segmentasi citra memiliki beberapa objek yang mempunyai fungsinya masing-masing yang akan dijelaskan pada tahap implementasi.

### 3. Menu Bantuan

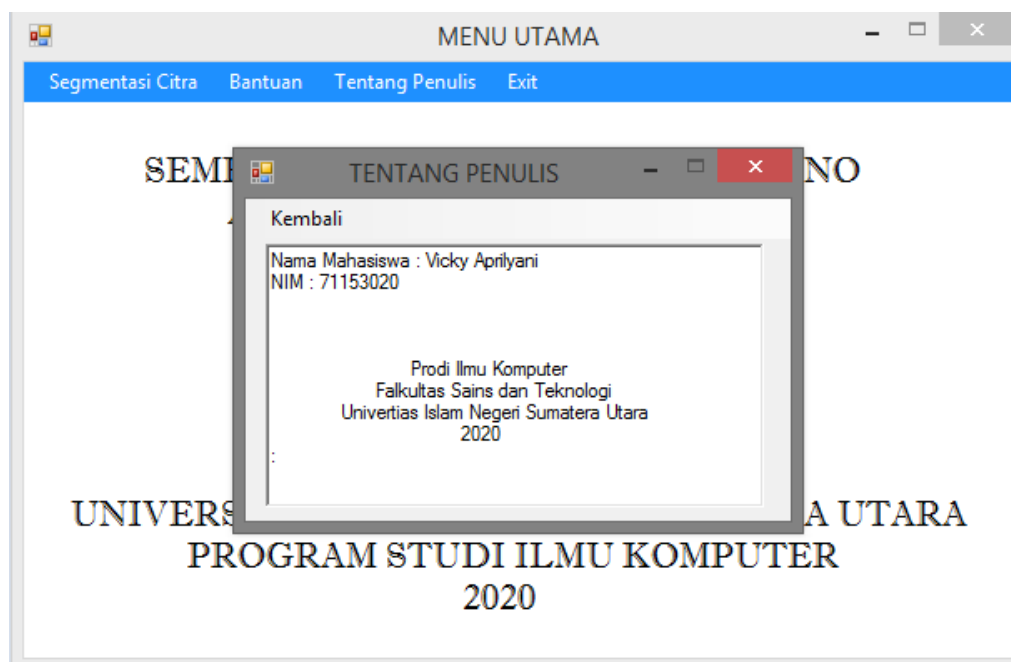
Menu bantuan yaitu untuk memberikan informasi tentang aplikasi. Adapun tampilan dari menu bantuan pada aplikasi bisa dilihat pada gambar dibawah ini:



**Gambar 4.10** Menu Bantuan

#### 4. Menu Tentang Penulis

Menu tentang penulis adalah menu yang berfungsi untuk menampilkan informasi dari penulis penelitian ini. Adapun tampilan dari menu tentang penulis dalam aplikasi bisa dilihat pada gambar dibawah ini:


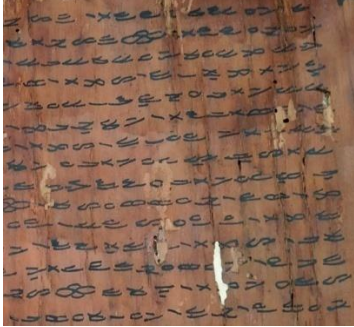

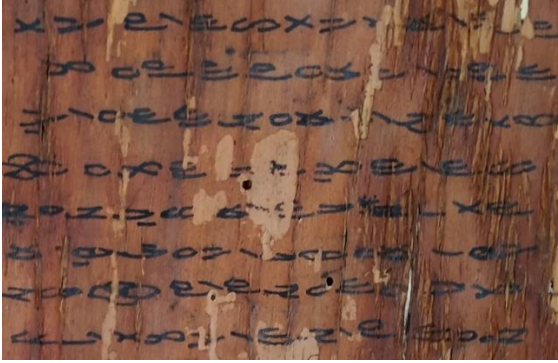
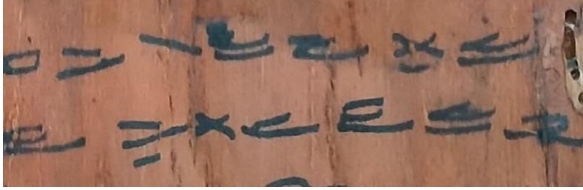


**Gambar 4.11** Menu Tentang Penulis

#### 4.3.1 Proses Segmentasi Citra Pada Aplikasi

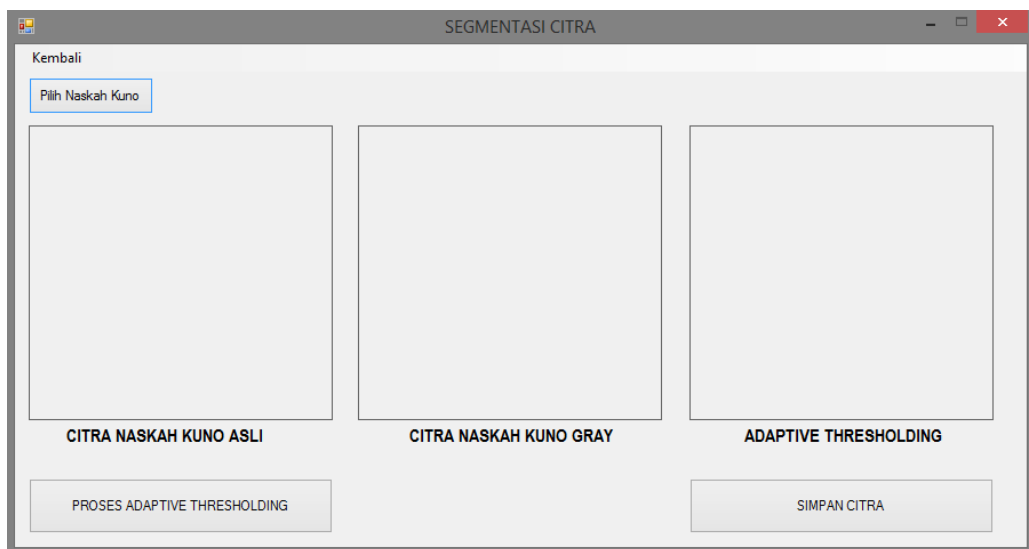
Proses segmentasi citra naskah kuno dilakukan pada menu segmentasi. Pada tahap ini adapun sampel yang diambil untuk dilakukan segmentasi adalah citra naskah kuno Batak yang di foto menggunakan camera smartphone. Citra naskah kuno Batak berasal dari Museum Negeri Provinsi Sumatera Utara. Berikut adalah gambar citra naskah kuno yang telah diambil dan dilakukan *cropping* untuk memperjelas citra:

**Tabel 4.5** Citra Naskah Kuno

Nama	Citra	Pixel	Ukuran
Sampel1		840x282	44KB
Sampel2		1805x1681	743KB
Sampel3		3121x2313	2MB
Sampel4		1573x1021	363KB
Sampel5		849x273	62KB

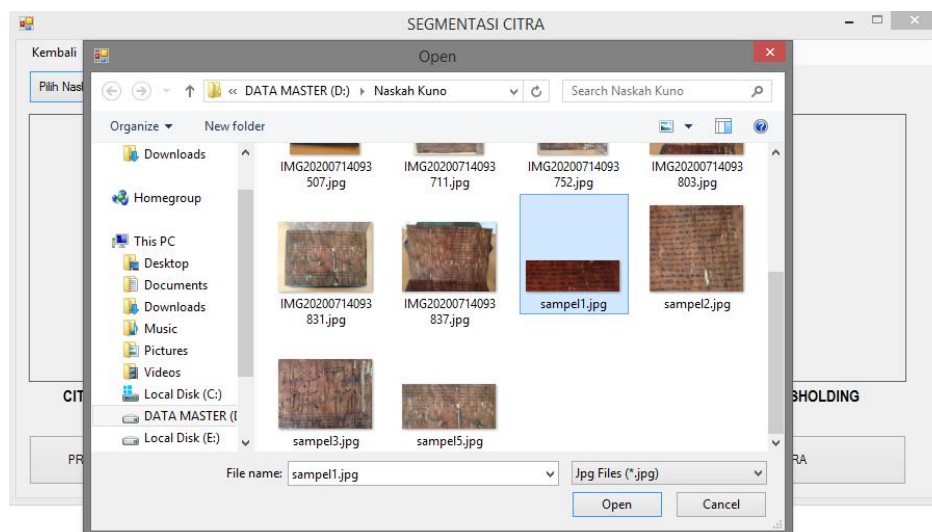
### 1. Proses Segmentasi Citra Naskah Kuno Sampel1

Berdasarkan pada tabel 4.6, selanjutnya semua citra naskah kuno yang telah didapatkan dimasukkan kedalam aplikasi segmentasi citra pada menu segmentasi citra dengan tahap sebagai berikut:



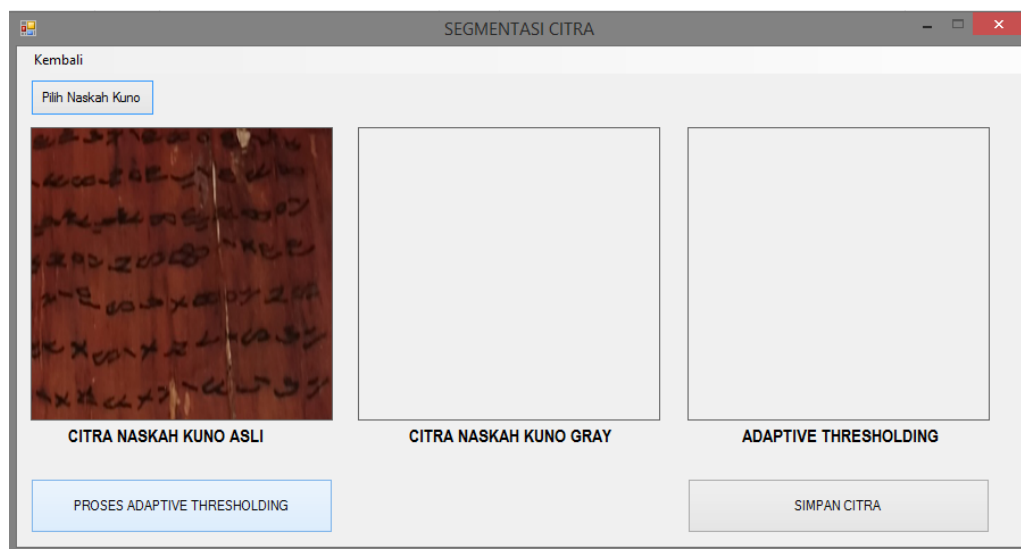
**Gambar 4.12** Proses Pemilihan Citra Naskah Kuno

Berdasarkan dari gambar diatas, untuk mencari citra naskah kuno yang sudah tersimpan didirektori komputer maka *user* dapat menekan *button* “Pilih Naskah Kuno”, kemudian muncul *pop up* menu pencarian seperti pada gambar dibawah ini:



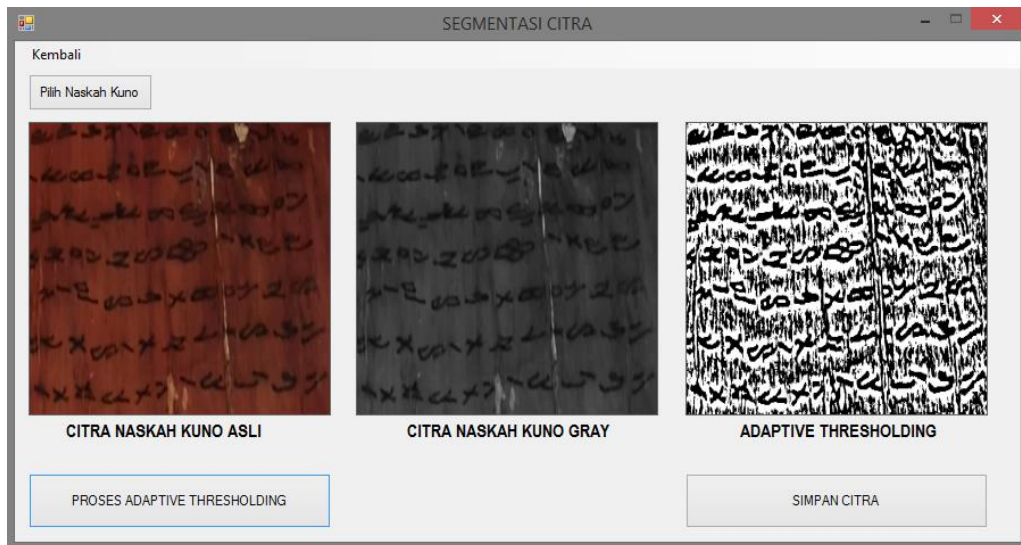
**Gambar 4.13** Pemilihan Citra Naskah Kuno

Berdasarkan dari gambar di atas, setelah dilakukan pemilihan citra naskah kuno maka *user* menekan *button* “*Open*” untuk memasukan citra naskah kuno pilihan kedalam aplikasi segmentasi, sehingga menampilkan seperti gambar dibawah ini:



**Gambar 4.14** Pemilihan Citra Naskah Kuno Sampel1

Berdasarkan pada gambar diatas, pada *box* pertama terdapat citra naskah kuno dengan nama sampel1 hasil pemilihan proses sebelumnya. Selanjutnya dilakukan proses *Adaptive Thresholding* dengan menekan menu button “Proses *Adaptive Thresholding*” sehingga akan menampilkan hasil proses seperti pada gambar dibawah ini:



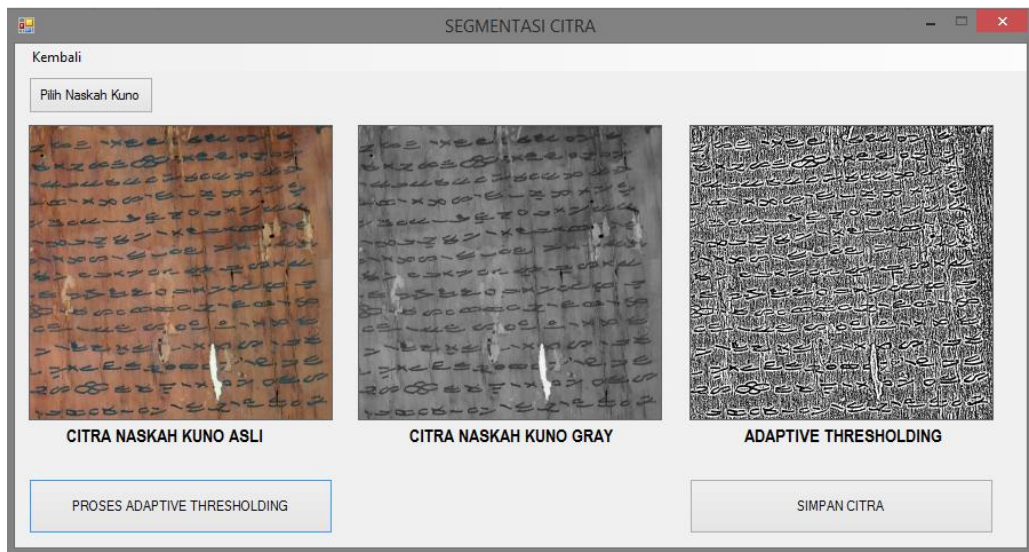
**Gambar 4.15** Hasil Proses Segmentasi Citra Naskah Kuno Sampel1

Dari gambar diatas, setelah dilakukannya proses segmentasi, maka pada bagian *box2* diisi oleh gambar citra naskah kuno *grayscale* yang diproses dari citra naskah kuno asli, dan pada *box3* merupakan hasil akhir segmentasi dengan *Algoritma Adaptive Thresolding* yang diambil dari citra naskah kuno *grayscale*. Terlihat perbedaan yang signifikan antara citra asli citra hasil segmentasi, *Algoritma Adaptive Thresolding* berhasil memisahkan objek citra *background* dengan teks citra naskah kuno asli dengan jelas.

## 2. Proses Segmentasi Citra Naskah Kuno Sampel2

Adapun dilakukan kembali uji coba segmentasi citra pada citra naskah kuno sampel ke2 dengan cara yang sama pada proses sebelumnya, sehingga menghasilkan citra naskah kuno hasil segmentasi seperti berikut:



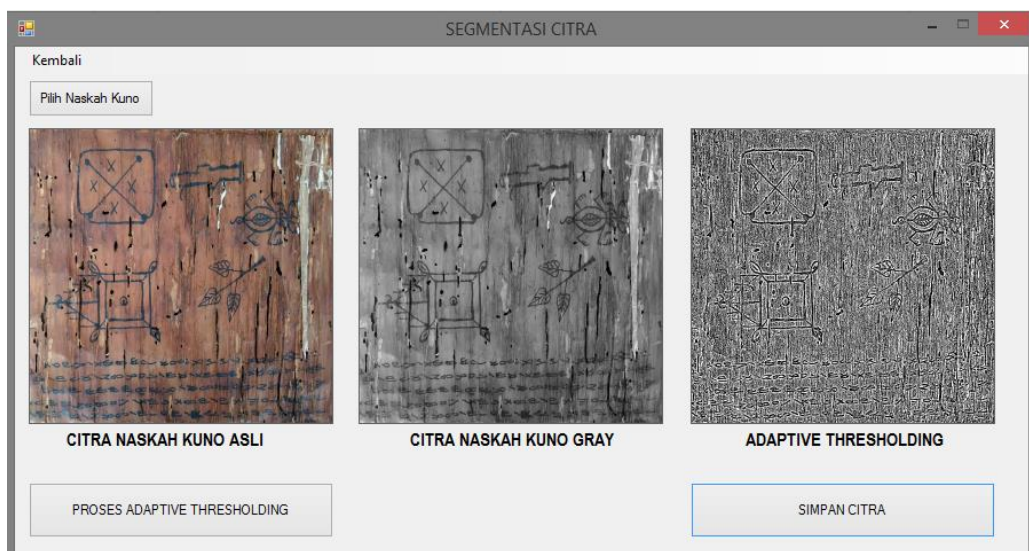


**Gambar 4.16** Hasil Proses Segmentasi Citra Naskah Kuno Sampel2

Berdasarkan pada gambar di atas, proses segmentasi citra naskah kuno dengan teks yang lebih banyak berhasil dilakukan menggunakan *Algoritma Adaptive Thresholding*.

### 3. Proses Segmentasi Citra Naskah Kuno Sampel3

Adapun dilakukan kembali uji coba segmentasi citra pada citra naskah kuno sampel ke3 dengan cara yang sama pada proses sebelumnya, sehingga menghasilkan citra naskah kuno hasil segmentasi seperti berikut:



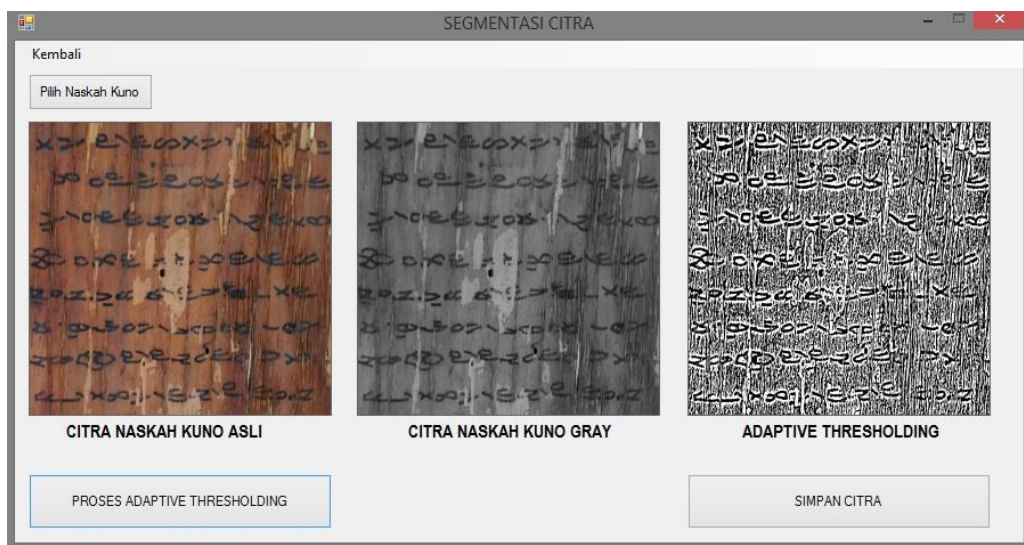
**Gambar 4.17** Hasil Proses Segmentasi Citra Naskah Kuno Sampel3



Berdasarkan pada gambar di atas, proses segmentasi citra naskah kuno dengan teks yang lebih banyak berhasil dilakukan menggunakan *Algoritma Adaptive Thresholding*, akan tetapi dengan background yang lebih luas dan karakter yang terlalu kecil maka hasil pemisahan objek tidak terlalu jelas.

#### 4. Proses Segmentasi Citra Naskah Kuno Sampel4

Adapun dilakukan kembali uji coba segmentasi citra pada citra naskah kuno sampel ke4 dengan cara yang sama pada proses sebelumnya, sehingga menghasilkan citra naskah kuno hasil segmentasi seperti berikut:

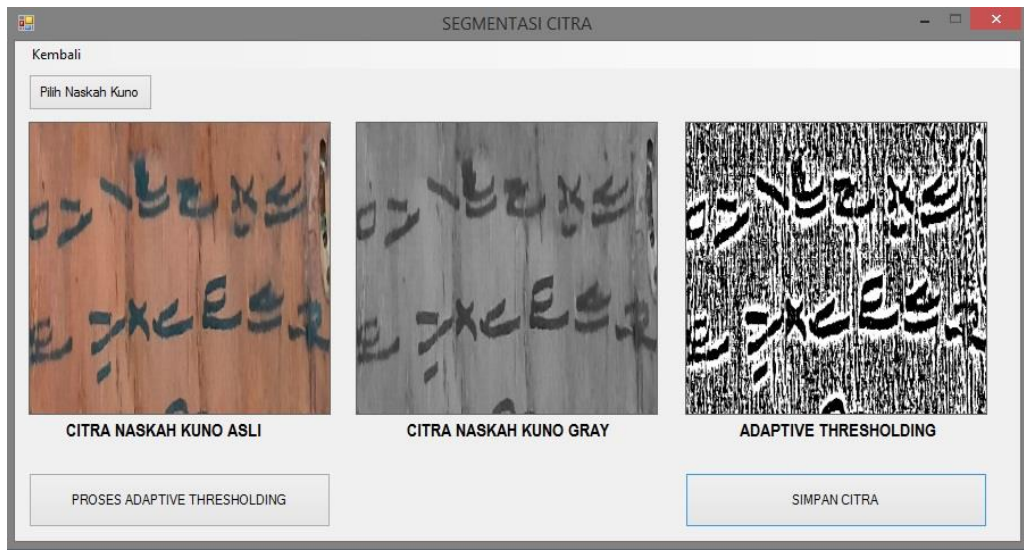


**Gambar 4.18** Hasil Proses Segmentasi Citra Naskah Kuno Sampel4

Berdasarkan pada gambar di atas, proses segmentasi citra naskah kuno dengan teks yang lebih sedikit berhasil dilakukan menggunakan metode *Adaptive Thresholding*, pemisahan objek terlihat sangat jelas.

#### 5. Proses Segmentasi Citra Naskah Kuno Sampel5

Adapun dilakukan kembali uji coba segmentasi citra pada citra naskah kuno sampel ke5 dengan cara yang sama pada proses sebelumnya, sehingga menghasilkan citra naskah kuno hasil segmentasi seperti berikut:

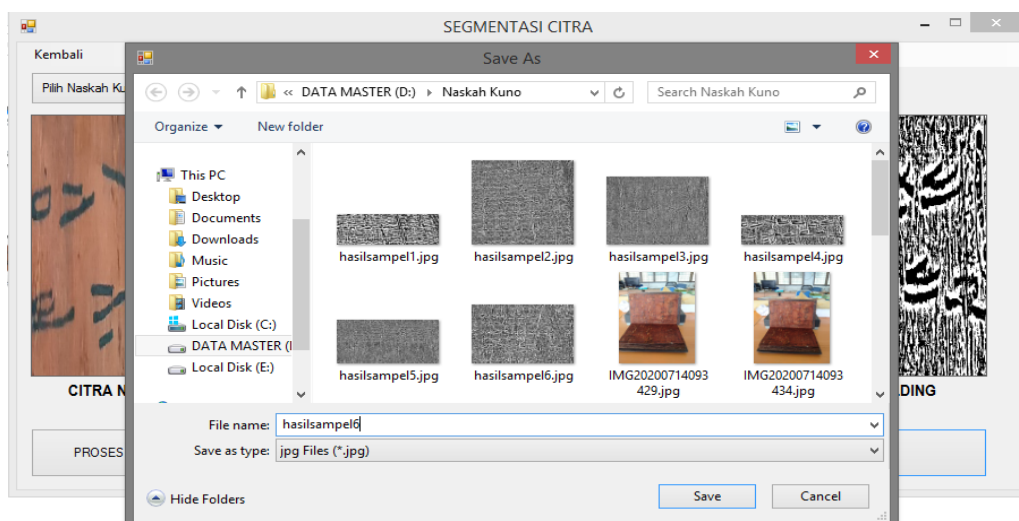


**Gambar 4.19** Hasil Proses Segmentasi Citra Naskah Kuno Sampel5

Berdasarkan pada gambar di atas, proses segmentasi citra naskah kuno dengan teks yang sedikit berhasil dilakukan menggunakan metode *Adaptive Thresholding*, dengan teks yang lebih pendek pemisahan objek terlihat sangat jelas.

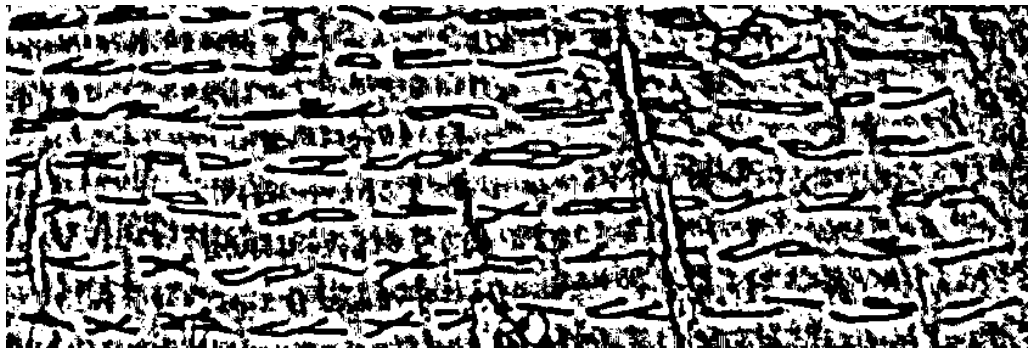
#### 6. Proses Penyimpanan Citra Hasil Segmentasi

Setelah dilakukannya segmentasi citra naskah kuno, maka citra hasil segmentasi dapat disimpan kelambil dengan menekan *button* “Simpan Citra”, sehingga menampilkan pop menu penyimpanan kedalam direktori seperti pada gambar dibawah ini:



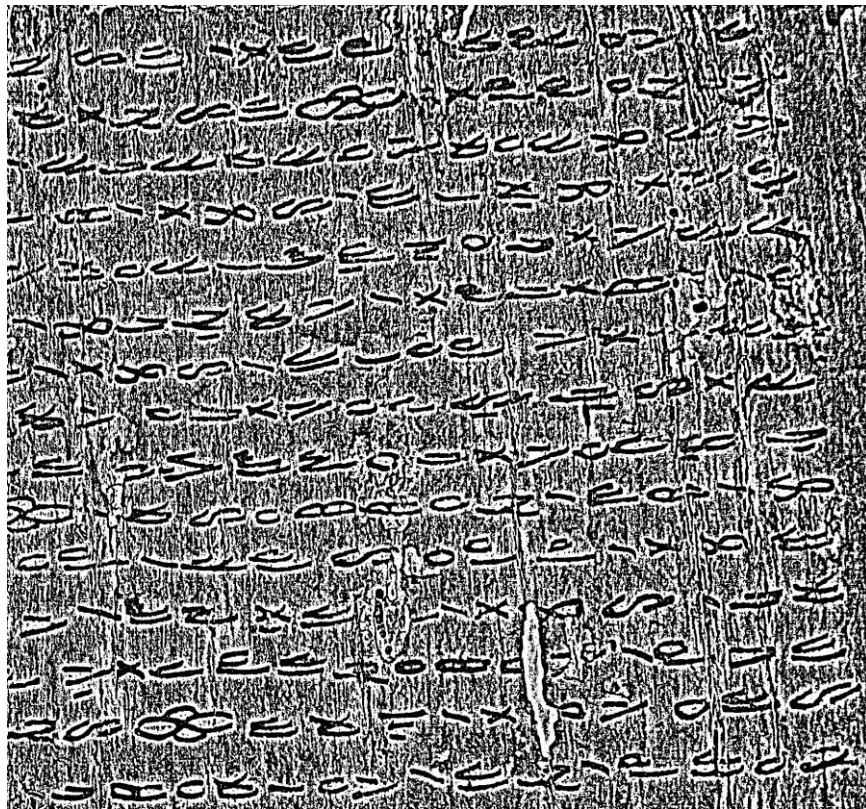
**Gambar 4.20** Proses Penyimpanan Citra Hasil Segmentasi

Berdasarkan pada gambar di atas, untuk menyimpan citra naskah kuno hasil segmentasi *user* menekan *button* “*Save*” dan citra tersimpan kedalam direktori. Berikut hasil dari proses segmentasi pada citra naskah kuno sampel1 hingga sampel 5:

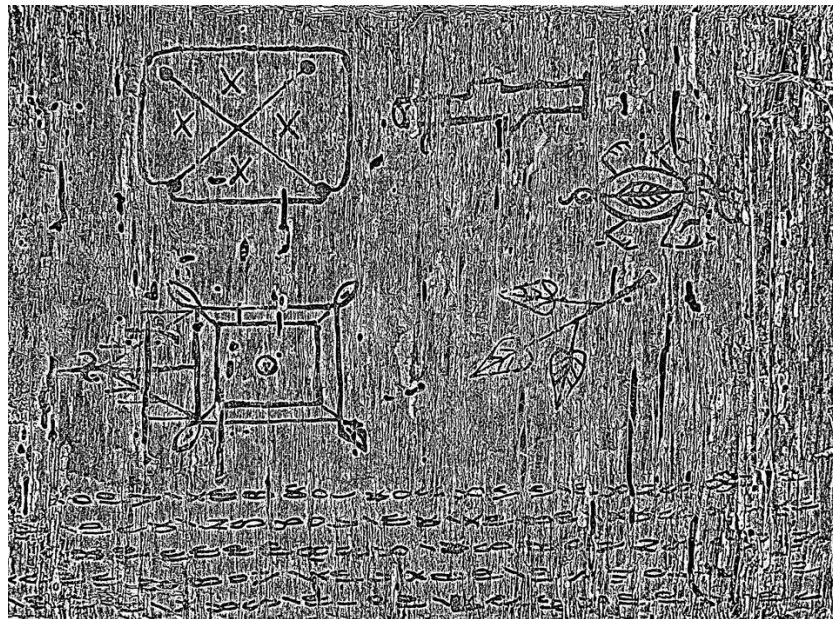


**Gambar 4.21** Citra Naskah Kuno Hasil Segmentasi Sampel1

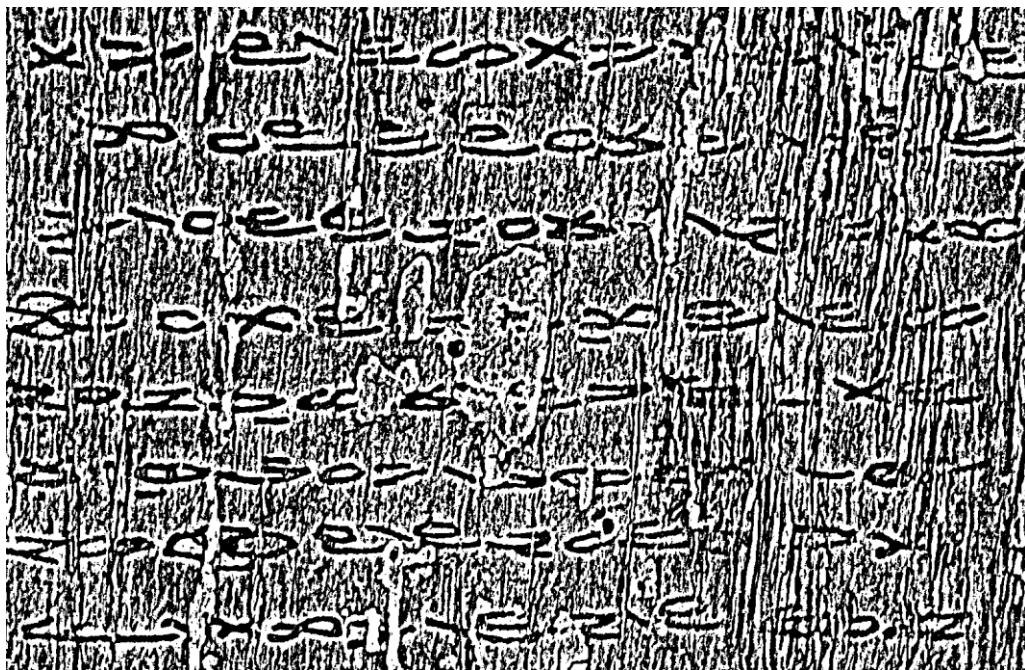
Berdasarkan pada gambar di atas, terlihat jelas bahwa citra *background* dan citra teks naskah kuno terpisah dan dapat dilihat dengan jelas. Jika *background* lebih halus maka hasil segmentasi akan lebih baik.



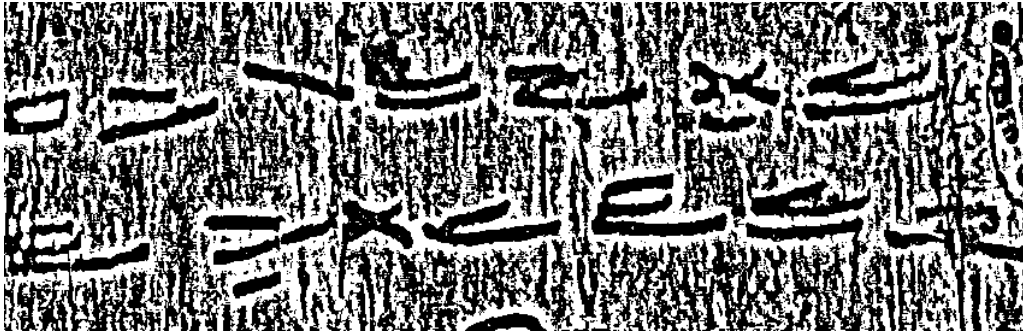
**Gambar 4.22** Citra Naskah Kuno Hasil Segmentasi Sampel2



Gambar 4.23 Citra Naskah Kuno Hasil Segmentasi Sampel3



Gambar 4.24 Citra Naskah Kuno Hasil Segmentasi Sampel4



**Gambar 4.25** Citra Naskah Kuno Hasil Segmentasi Sampel5

Berdasarkan pada gambar-gambar hasil segmentasi naskah kuno di atas, terlihat jelas bahwa citra *background* dan citra teks naskah kuno terpisah dan dapat dilihat dengan jelas pada teks naskah kuno yang sedikit. Sedangkan pada teks citra naskah kuno yang terlalu banyak maka hasilnya sedikit tidak jelas, hal ini disebabkan oleh *background* dari citra naskah kuno yang tidak halus. Jika *background* lebih halus maka hasil segmentasi akan lebih baik.

#### 4.4 Hasil dan Pembahasan

Berdasarkan hasil implementasi segmentasi citra naskah kuno pada program aplikasi, maka didapat hasil perbandingan ukuran citra naskah kuno dari dengan sampel yang telah di proses sebagai berikut:

**Tabel 4.6** Hasil Perbandingan Ukuran Citra Naskah Kuno

Nama Citra	Resolusi Citra Asli (Pixel)	Ukuran Citra Asli	Resolusi Citra Segmentasi (Pixel)	Ukuran Citra Segmentasi
Sampel1	840x282	44 KB	840x282	30 KB
Sampel2	1805x1681	743 KB	1805x1681	437 KB
Sampel3	3121x2313	2 MB	3121x2313	1,1 MB
Sampel4	1573x1021	363 KB	1573x1021	226 KB
Sampel5	849x273	62 KB	849x273	32 KB

Berdasarkan dari tabel 4.7 diatas, bisa disimpulkan bahwa *Algoritma Adaptive Thresholding* dengan proses segmentasi citra naskah kuno menurunkan ukuran asli citra dengan resolusi yang sama.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan pada hasil uji coba dari segmentasi citra naskah kuno dari bab sebelumnya bisa diambil kesimpulannya bahwa:

1. *Algoritma Adaptive Thresholding* dapat melakukan segmentasi citra naskah kuno dengan memisahkan teks citra naskah kuno dengan *background* citra sehingga terdapat perbedaan sisi yang jelas.
2. Pada citra naskah kuno yang memiliki teks lebih banyak, maka metode *Adaptive Thresholding* menghasilkan citra segmentasi yang kurang baik diakibat beberapa faktor seperti *background* yang tidak halus.
3. Terdapat perbedaan ukuran pada citra naskah kuno asli dengan citra naskah kuno hasil segmentasi, perbedaan ini ukuran tersebut tidak mempengaruhi jumlah pixel citra, hal ini dapat dilihat dari jumlah *pixel* yang tetap sama.
4. Dengan adanya aplikasi segmentasi citra naskah kuno, diharapkan dapat memberikan kemudahan pihak Museum Negeri Provinsi Sumatera Utara dalam melastarikan naskah kuno yang sudah mulai hancur.

#### **5.2 Saran**

Berdasarkan pada kesimpulan yang diperoleh dengan melakukan uji coba segmentasi citra naskah kuno, maka peneliti bisa memberikan saran sebagai berikut:

1. Dalam melakukan segmentasi citra, dapat ditambahkan dengan metode lain agar hasil segmentasi citra lebih halus dan jelas.
2. Pada aplikasi yang telah dibangun, belum terdapat algoritma yang dapat mendeteksi karakter dari citra hasil segmentasi yang dirubah kedalam karakter ASCII, maka dapat ditambahkan algoritma baru untuk menyempurnakan aplikasi.



3. Aplikasi yang telah dibangun masih menggunakan versi *dekstop*, diharapkan dapat dikembangkan menjadi aplikasi berbasis *mobile*.



## DAFTAR PUSTAKA

- Ahmad, U. 2005. *Pengolahan Citra Digital dan Teknik Pemogramanya*. Yogyakarta: Graha Ilmu
- Alfiansyah, & Wulandari, F., 2018, *Segmentasi Pembuluh Darah Retina Menggunakan Local Adaptive Thresholding. Annual Research Seminar (ARS)*, 40-43.
- Apriliani, D., & Murinto., 2013, *Analisis Perbandingan Teknik Segmentasi Citra Digital Menggunakan Metode Level-Set Chan & Vese dan Lankton. Jurnal Sarjana Teknik Informatika* , 232-240.
- Bahar, H., & Mathar, T., 2015, *Upaya Pelestarian Naskah Kuno di Perpustakaan dan Arsip Daerah Provinsi Sulawesi Selatan. Khizanah Al-Hikmah* , 89-100.
- Furqan, M., Hasugian, A. H., & Tanjung, R. H. (2020). Digital Image Enhancement Using The Method Of Multiscale Retina And Median Filter. *JURNAL INFOKUM*, 69-76
- Hatta, M., & Susrama, I. G., 2017, *Counting Sperma Aktif Menggunakan Metode Otsu Threshold Dan Local Adaptive Threshold. Teknika: Engineering and Sains Journal*, 47-54.
- Hermawati, F.A. 2013, *Pengolahan Citra Digital: Konsep &Teori*. Yogyakarta:Andi
- Kumaseh, M. R., Latumakulita, L., & Nainggolan, N., 2013, *Segmentasi Citra Digital Ikan Menggunakan Metode Thresholding. Jurnal Ilmiah Sains* , 74-79.
- Latiar, H., 2018, *Preservasi Naskah Kuno Sebagai Upaya Pelestarian Budaya Bangsa. Al-Kuttab* , 67-83.
- Nafi'iyah, N., 2015, *Algoritma Kohonen Dalam Mengubah Citra Graylevel Menjadi Citra Biner. Jurnal Ilmiah Teknologi dan Informasia ASIA (JITIKA)* , 49-55.
- Nuraini, R., 2017, *Desk Check Table Pada Flowchart Operasi Perkalian Matriks. Jurnal PETIR* , 60-65.
- Putra, D., 2010, *Pengolahan Citra Digital*. Yogyakarta: CV. ANDI OFFSET.

- Putranto, B. Y., Hapsari, W., & Wijana, K., 2010, *Segmentasi Warna Citra Dengan Deteksi Warna HSV Untuk Mendeteksi Objek*. *JURNAL INFORMATIKA* , 1-14.
- Rakhmat, R. K., & Sriani., 2016, *Penerapan Metode Walvelet Co-Occurrence Histogram Untuk Pengenalan Objek Pada Citra Digital*. *Konferensi Nasional Teknologi Informasi dan Komunikasi (KeTIK) 2016* (pp. 1-9). Medan: USU Press.
- Rahman, R., & Julianto, V., 2015, *Aplikasi Pemilihan Pintu Pagar Menggunakan Metode Simple Additive Weighting Method (SAW)*. *Jurnal Sains dan Informatika* , 62-71.
- Sinaga, A. S., 2017, *Implementasi Teknik Thresholding Pada Segmentasi Citra Digital*. *Jurnal Manajemen dan Informatika Pelita Nusantara* , 48-51.
- Sutramiani, N. P., & Putra, I. K., 2015, *Local Adaptive Thresholding Pada Processing Citra Lontar Aksara Bali*. *JURNAL TEKNOLOGI ELEKTRO*, 27-30.
- Som, H.M., Zain, J.M. & Ghazali, A.J. 2011. *Application of threshold techniques for readability improvement of jawi historical manuscript images*. *Advanced Computing: An International Journal* 2(2): 60 – 69.
- Sutoyo, T., Mulyanto, E., Suhartono, V., & Nurhayati, O. D., 2009, *Teori Pengolahan Citra Digital*. Yogyakarta dan Semarang: C.V ANDI OFFSET dan ANDINUS Semarang.
- Tajuddin, M., Husain, & Jaya, N. N., 2017, *Preservasi Naskah Kuno Sasak Lombok Berbasis Digital dan Webside*. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)* , 445-454.
- Wikipedia, 2020. C Sharp (Online) [https://id.wikipedia.org/wiki/C\\_sharp](https://id.wikipedia.org/wiki/C_sharp)
- Wikipedia, 2020. Microsoft Visual Studi (Online) [https://id.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://id.wikipedia.org/wiki/Microsoft_Visual_Studio)
- Zhou, H., Wu, J. & Zhang, J. 2010. *Digital Image Processing: Part II*. Bookboon.com(Online)<http://www.zums.ac.ir/files/research/site/ebooks/electrical-electronic-engineering/digitalimage-processing-part-two.pdf>.

## LISTING PROGRAM

### 1. Menu Utama

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace AdaptiveThresholdingApril
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        private void segmentasiCitraToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            Form1 f1 = new Form1();
            f1.Show();
            this.Hide();
        }

        private void exitToolStripMenuItem_Click(object sender, EventArgs
e)
        {
            this.Close();
        }

        private void tentangPenulisToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            Form4 f4 = new Form4();
            f4.Show();
        }

        private void bantuanToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            Form3 f3 = new Form3();
            f3.Show();
        }

        private void Form2_Load(object sender, EventArgs e)
        {

```

```

    }
}

```

## 2. Segmentasi Citra

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.Structure;
namespace AdaptiveThresholdingApril
{
    public partial class Form1 : Form
    {
        private Bitmap bmp2 = null;
        Image<Bgr, byte> imgInput;
        Image<Gray, byte> imgGray;
        Image<Gray, byte> imgBinarize;
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            OpenFileDialog ofd = new OpenFileDialog();
            ofd.Filter = "Jpg Files|*.jpg";
            if (ofd.ShowDialog() == DialogResult.OK)
            {
                imgInput = new Image<Bgr, byte>(ofd.FileName);
                pictureBox1.Image = imgInput.Bitmap;
            }
        }

        private void button2_Click(object sender, EventArgs e)
        {
            imgGray = imgInput.Convert<Gray, byte>();
            pictureBox2.Image = imgGray.Bitmap;
            imgBinarize = new Image<Gray, byte>(imgGray.Width,
            imgGray.Height, new Gray(0));
            CvInvoke.AdaptiveThreshold(imgGray, imgBinarize, 255,
            Emgu.CV.CvEnum.AdaptiveThresholdType.MeanC,
            Emgu.CV.CvEnum.ThresholdType.Binary, 25, 0.0);
            pictureBox3.Image = imgBinarize.Bitmap;
        }
    }
}

```

```

private void button3_Click(object sender, EventArgs e)
{
    bmp2 = (Bitmap)pictureBox3.Image;
    SaveFileDialog sfd = new SaveFileDialog();
    sfd.Filter = "jpg Files|.jpg";
    if (sfd.ShowDialog() == DialogResult.OK)
    {
        bmp2.Save(sfd.FileName);
    }
}

private void fileToolStripMenuItem_Click(object sender, EventArgs
e)
{
    Form2 f2 = new Form2();
    f2.Show();
    this.Close();
}

private void Form1_Load(object sender, EventArgs e)
{
}
}
}
}

```

### 3. Menu Bantuan

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace AdaptiveThresholdingApril
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }

        private void kembaliToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            this.Close();
        }
    }
}

```

```

        private void menuStrip1_ItemClicked(object sender,
ToolStripItemClickedEventArgs e)
        {

        }

        private void Form3_Load(object sender, EventArgs e)
        {

        }
    }
}

```

#### 4. Tentang Penulis

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace AdaptiveThresholdingApril
{
    public partial class Form4 : Form
    {
        public Form4()
        {
            InitializeComponent();
        }

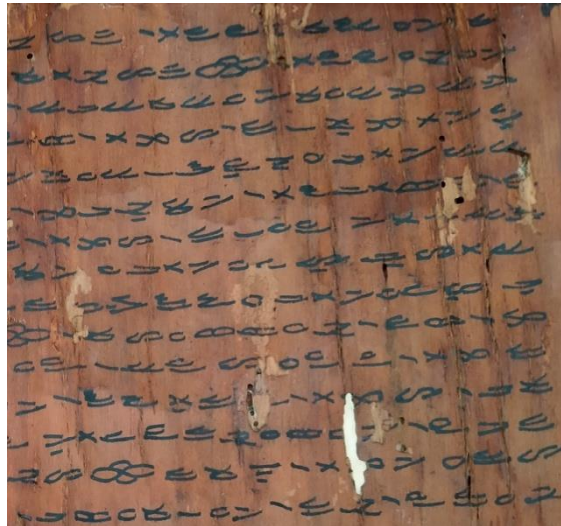
        private void kembaliToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            this.Close();
        }

        private void Form4_Load(object sender, EventArgs e)
        {

        }
    }
}

```

## PROSES HITUNGAN MANUAL



**Gambar 4.5.** Sampel Citra Naskah Kuno

Berdasarkan dari gambar diatas, sampel citra naskah kuno berformat ,jpg dengan resolusi 256 x 256 pixel. Selanjutnya adalah melakukan konversi citra kedalam bentuk *grayscale*. Perhitungan mengkonversi citra RGB menjadi citra *grayscale* secara manual dilakukan pada ukuran matriks 3 x 3 pada pixel yang bias dilihat pada Tabel 4.1.

**Tabel 4.1** Nilai pixel 3 x 3 Sampel Naskah Kuno

x\y	0	1	2
0	150 251 64	255 130 5	130 0 66
1	0 64 112	120 30 201	76 105 111
2	190 172 98	99 125 145	170 140 53

*Luminosity grayscale* dapat dihitung dengan memakai persamaan seperti berikut:

$$G(x, y) = (0.21 \times R) + (0.71 \times G) + (0.07 \times B)$$

Dari persamaan diatas, maka perhitungan untuk matriks (0,0) hingga matriks (2,2) yaitu:

$$G(0,0) = (0.21 \times 150) + (0.71 \times 251) + (0.07 \times 64) = 214,19$$

$$G(0,1) = (0.21 \times 0) + (0.71 \times 64) + (0.07 \times 112) = 53,28$$

$$G(0,2) = (0.21 \times 190) + (0.71 \times 172) + (0.07 \times 98) = 168,88$$

$$G(1,0) = (0.21 \times 255) + (0.71 \times 130) + (0.07 \times 5) = 146,2$$

$$G(1,1) = (0.21 \times 120) + (0.71 \times 30) + (0.07 \times 201) = 60$$

$$G(1,2) = (0.21 \times 99) + (0.71 \times 125) + (0.07 \times 145) = 119,69$$

$$G(2,0) = (0.21 \times 130) + (0.71 \times 0) + (0.07 \times 66) = 31,92$$

$$G(2,1) = (0.21 \times 76) + (0.71 \times 105) + (0.07 \times 111) = 98,28$$

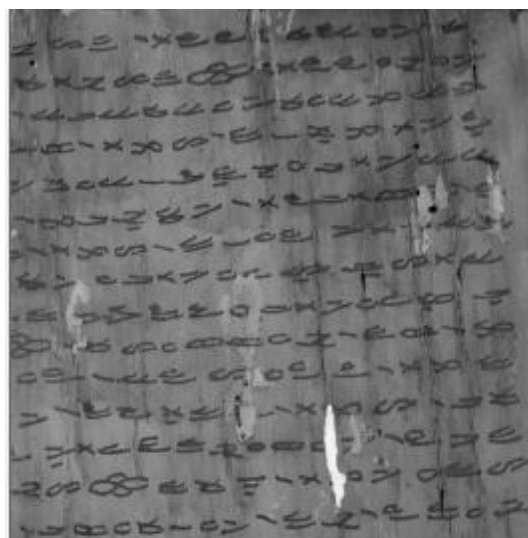
$$G(2,2) = (0.21 \times 170) + (0.71 \times 140) + (0.07 \times 53) = 138,81$$

Dari perhitungan sebelumnya, maka mendapatkan hasil nilai pixel 3 x 3 tipe *grayscale* yang sudah dibulatkan seperti pada tabel 4.2.

**Tabel 4.2** Hasil Nilai Pixel 3 x 3 *Grayscale* Sampel Naskah Kuno Citra

x\y	0	1	2
0	214	146	32
1	53	60	98
2	169	120	139

Adapun hasil pengujian dari metode *luminosity grayscale* pada citra naskah kuno bisa dilihat pada gambar berikut ini:





#### Gambar 4.6. Hasil *Grayscale* Sampel Citra Naskah Kuno

Citra naskah kuno yang sudah melalui proses *grayscale* berikutnya akan dihitung dengan memakai *Algoritma Adaptive Thresholding*. Citra naskah kuno dibagi menjadi beberapa blok, setelah itu dihitung secara adaptif dari blok pertama hingga blok terakhir sesuai dengan nilai *threshold* masing-masing. Selanjutnya adalah menghitung nilai mean (rata-rata) intensitas citra dari tiap sub citra dengan persamaan citra sebagai berikut:

$$\text{Means}_{[i]} = \bar{x}_i = \frac{\sum_{j=1}^n x_j}{N}$$

$n$  adalah jumlah sub citra. Berdasarkan pada contoh citra sampel 3 x 3 maka perhitungan *mean* sub citra akan diproses hingga sub citra ke-3. Penjumlahan dimulai dari *pixel* pertama ( $j = 1$ ) hingga *pixel* ke 3 x 3 ( $n$ ), selanjutnya dibagi dengan  $n$ . Berikut ini adalah perhitungan yang dilakukan secara manual untuk kelanjutan dari kasus uji coba citra naskah kuno dari subcitra pertama.

Means

$$X_0 = (0,0) - (0,2) = \frac{214 + 53 + 169}{3} = 145$$

$$X_1 = (1,0) - (1,2) = \frac{146 + 60 + 98}{3} = 101$$

$$X_2 = (2,0) - (2,2) = \frac{32 + 98 + 139}{3} = 90$$

Hasil dari nilai *mean* yang didapat dari sub citra pertama ialah 145, sub citra kedua adalah 101 dan sub citra terakhir adalah 90 sesuai dengan sampel citra 3 x 3. Langkah berikutnya yakni menghitung *variance* dari sub citra tersebut. *Variance* dipakai sebagai nilai batas intensitas *pixel* untuk menentukan *output* citra biner. Berikut adalah persamaan untuk menghitung *variance*:

$$\text{var}_{[i]} = \frac{\sum_{j=1}^n (x_j - \bar{x}_i)^2}{n-1}$$

Rumus diatas menyatakan penjumlahan untuk tiap *pixel* dari sub citra ke- $i$  yang dikurangi dengan hasil nilai mean, selanjutnya hasil penjumlahan dibagi

dengan total blok *pixel* sub citra dikurang 1. Nilai *variance* yang diperoleh akan masuk ke *statement* kondisi dengan persamaan:

$$var_{[i]} > 100 \text{ atau } var_{[i]} \leq 100$$

Perhitungan yang dilakukan secara manual untuk dua nilai *variance*. Berikut perhitungan pada kasus uji citra naskah kuno:

$$Var_0 = (0, 0) - (0, 2) = Var = \frac{(214 - 145)^2 + (53 - 145)^2 + (169 - 145)^2}{9-1}$$

$$Var = \frac{26,887 + 8,464 + 576}{8} = 4,490.875$$

$$Var_1 = (1, 0) - (1, 2) = Var = \frac{(146 - 101)^2 + (60 - 101)^2 + (120 - 101)^2}{9-1}$$

$$Var = \frac{2,025 + 574 + 361}{8} = 370$$

$$Var_2 = (2, 0) - (2, 2) = Var = \frac{(32 - 90)^2 + (98 - 90)^2 + (139 - 90)^2}{9-1}$$

$$Var = \frac{3,364 + 64 + 2,401}{8} = 5,829$$

Dari hasil perhitungan manual yang sudah dilakukan, nilai *variance* menunjukkan  $> 100$ . Pada citra uji naskah kuno ini, tidak ada ditemukannya nilai *variance* dibawah angka 100. Kesimpulannya adalah pengubahan citra biner lebih efektif dilakukan dengan intensitas nilai yang ada pada tiap-tiap sub citra. Sehingga selanjutnya adalah dilakukan Thresolding secara adaptif dengan rumus

$$(Var_{[i]} > 100) \rightarrow T_i = \frac{bloksub[i]}{2}$$

Jika nilai *variance*  $> 100$ , maka dilakukan perhitungan nilai batas *T* (*threshold*) untuk tiap sub citra ke-*i* dengan terlebih dahulu mencari nilai intensitas *pixel* dari setiap blok (*bloksub*). Adapun nilai *T* untuk sub citra ke-*i* sebagai berikut:

$$\text{Blok } (0,0) = \frac{214}{2} = 107$$

$$\text{Blok } (0,1) = \frac{53}{2} = 27$$

$$\begin{aligned} \text{Blok (0,2)} &= \frac{169}{2} = 85 \\ \text{Blok (1,0)} &= \frac{146}{2} = 73 \\ \text{Blok (1,1)} &= \frac{60}{2} = 30 \\ \text{Blok (1,2)} &= \frac{120}{2} = 60 \\ \text{Blok (2,0)} &= \frac{32}{2} = 16 \\ \text{Blok (2,1)} &= \frac{98}{2} = 49 \\ \text{Blok (2,2)} &= \frac{139}{2} = 70 \end{aligned}$$

**Tabel 4.3** Hasil Nilai Batas Threshold

x\y	0	1	2
0	107	73	16
1	27	30	49
2	85	60	70

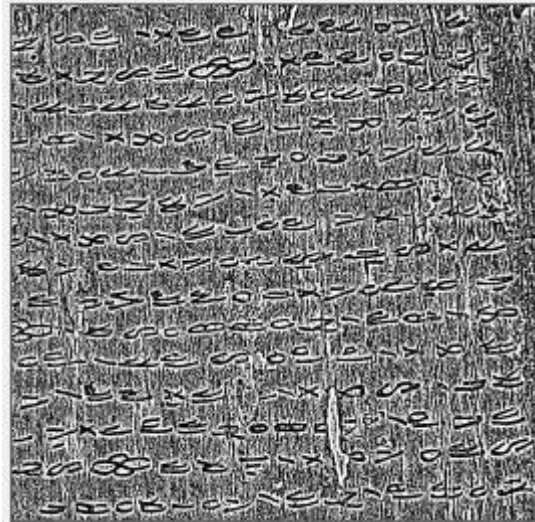
Setelah mengetahui nilai batas *threshold* ( $T$ ), berikutnya yaitu menentukan hasil *pixel* citra biner (0 atau 1) dengan cara membandingkan nilai intensitas *pixel* dan nilai batas  $T$ . Apabila nilai batas *threshold* lebih besar dari nilai *pixel* tersebut, maka *pixel* citra akan dikonversi menjadi 0 (hitam). Sebaliknya, apabila nilai batas *threshold* lebih kecil dari nilai *pixel*, maka *pixel* citra akan dikonversi menjadi 1 (putih). Berikut *output pixel* citra yang diuji dalam citra sampel 3 x 3:

**Tabel 4.4** Hasil nilai biner pixel 3 x 3 naskah kuno

x\y	0	1	2
0	1	0	0
1	0	0	0
2	0	0	0

Berdasarkan dari tabel diatas, dapat terlihat dengan jelas pemisahan antara *background* dengan objek citra naskah kuno. Berhubungan *citra* sampel yang dihitung dalam perhitungan manual hanya 3 x 3 maka tidak terlalu terlihat

pembagian secara merata. Adapun hasil ilustrasi dari keseluruhan citra naskah kuno hasil *thresolding* secara adaptif bisa dilihat pada gambar berikut ini:



**Gambar 4.7** Hasil *Adaptive Thresholding* Sampel Citra Naskah Kuno

## DAFTAR RIWAYAT HIDUP

### **Data Pribadi**

Nama : Vicky Apriyani  
Tempat Tanggal Lahir : Karang Sari, 04 November 1997  
Alamat : Karang Sari  
Desa : Damuli Pekan  
Kecamatan : Kualuh Selatan  
Kabupaten : Labuhan Batu Utara  
No. Telepon : 082276488197  
Email : vickyaprilia20@gmail.com  
Jenis Kelamin : Perempuan  
Agama : Islam  
Kewarganegaraan : Indonesia  
Nama Orang Tua  
Ayah : Kasir Ipa  
Ibu : Sugiati







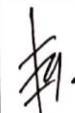





### **Latar Belakang Pendidikan**









1. Lulusan : TK Raudhatul Athfal Al-Kalam Kualuh Selatan  
2002-2003
2. Lulusan : SD Negeri 117511 Karang Sari  
2003-2009
3. Lulusan : MTs Negeri Damuli Pekan  
2009-2012
4. Lulusan : SMA Negeri 1 Kualuh Selatan  
2012-2015
5. Lulusan : Universitas Islam Negeri Sumatera Utara Medan  
2015-2020

### KARTU BIMBINGAN SKRIPSI

Semester Gasal/Genap Tahun Akademik ..... / .....

Nama : Vicky Aprizani	Pembimbing I : Dr. Mhd. Furqan, S.Si, M. Comp. Sc
NIM : 71153020	Pembimbing II : Rakhmat Kurniawan R, ST, M Kom
Prog. Studi : Ilmu Komputer	SK Pembimbing :
Judul Skripsi : Segmentasi Citra Pada Citra Mastah Kuno Dengan Menggunakan <del>Citra Thresholding</del> Algoritma Local Adaptive Thresholding	

P E R T	PEMBIMBING I			PEMBIMBING II		
	Tgl.	Materi Bimbingan	Tanda Tangan	Tgl.	Materi Bimbingan	Tanda Tangan
I	05/02/2020	Perbaiki Latar belakang		04/2019	Revisi Bab I	
II	06/02/2020	Revisi Bab II		20/2019	Revisi Bab II	
III	08/02/2020	Perbaiki flowchart		09/2019	Lanjut Bab III	
IV	09/02/2020	Perbaiki Bab III		21/2020	Perbaiki Bab III &	
V	10/02/2020	Acc Seminar Proposal		09/2020	ACC Seminar Proposal	

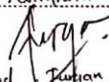
VI	11/2020 06	Revisi Bab IV		Revisi Bab IV	
VII	01/2020 07	Revisi Bab IV		Penguasaan materi Bab IV	
VIII	11/2020 04	Abstrak, Daftar Pustaka		Ace Sidang Munasabah	
IX	03/2020 08	Revisi semua bab dan penguasaan materi			
X	11/2020 08	Ace sidang munasabah			

Medan, 10 Agustus : 20 20

An. Dekan

Ketua Jurusan/Program Studi

Ilmu Komputer



Dr. Mhd. Fathan, S.Si. M. Comp. Sc

NIP. 198003062006091003

Catatan: Pada saat bimbingan, kartu ini harus diisi dan ditandatangani oleh pembimbing