



TUGAS AKHIR - SM141501

PENERAPAN ALGORITMA FUZZY C-MEANS UNTUK KLASIFIKASI CITRA TENUN BERDASARKAN FITUR TEKSTUR

'ISYI BIFAWA'IDATI
NRP 1212 100 043

Dosen Pembimbing
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT
Drs. Soetrisno, M.IKomp

JURUSAN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - SM141501

APPLICATION OF FUZZY C-MEANS ALGORITHM FOR TENUN IMAGE CLASSIFICATION BASED ON TEXTURE FEATURE

'ISYI BIFAWA'IDATI
NRP 1212 100 043

Dosen Pembimbing
Dr. Dwi Ratna Sulistyanningrum, S.Si, MT
Drs. Soetrisno, M.IKomp

JURUSAN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2017

LEMBAR PENGESAHAN
PENERAPAN ALGORITMA FUZZY C-MEANS
UNTUK KLASIFIKASI CITRA TENUN
BERDASARKAN FITUR TEKSTUR
APPLICATION OF FUZZY C-MEANS ALGORITHM
FOR TENUN IMAGE CLASSIFICATION BASED
ON TEXTURE FEATURE

TUGAS AKHIR

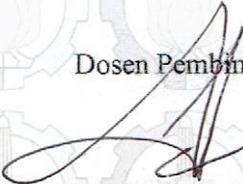
Diajukan untuk memenuhi salah satu syarat
Untuk memperoleh gelar Sarjana Sains
Pada bidang minat Ilmu Komputer
Program Studi S-1 Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

Oleh :
'**ISYI BIFAWA**'IDATI
NRP. 1212 100 043

Menyetujui,

Dosen Pembimbing II,

Dosen Pembimbing I,


Drs. Soetrisno, M.IKomp

NIP. 19571103 198603 1 003


Dr. Dwi Ratna Sulistyaningrum, S.Si., M.T.

NIP. 19690405 199403 2 003

Mengetahui,

Ketua Jurusan Matematika
FMIPA ITS


Dr. Imam Mukhlash, S.Si. MT

NIP. 19700831 199403 1 003

Surabaya, 19 Januari 2017

PENERAPAN ALGORITMA FUZZY C-MEANS UNTUK KLASIFIKASI CITRA TENUN BERDASARKAN FITUR TEKSTUR

Nama Mahasiswa : ‘Isyi Bifawa’idati
NRP : 1212 100 043
Jurusan : Matematika
Dosen Pembimbing : 1. Dr. Dwi Ratna S., S.Si, M.T.
2. Drs. Soetrisno, M.IKomp.

Abstrak

Tenun merupakan salah satu kain tradisional Indonesia yang memiliki makna, nilai sejarah, keindahan warna dan motif. Oleh karena itu, pelestarian kain tenun harus sangat diperhatikan. Salah satu yang dapat menangani hal tersebut adalah adanya kemudahan dalam mengakses dan mengolah data kain tenun pada media komputer. Jika terdapat kemudahan dalam akses penyebaran kain tenun, maka memungkinkan untuk menarik minat masyarakat baik dari Indonesia maupun mancanegara. Hal tersebut dapat meningkatkan nilai ekonomi Indonesia. Pada penelitian ini menggunakan tiga motif tenun dari tiga daerah yaitu, motif Endek dari Bali, motif Lurik dari NTT, dan motif Rangrang dari Jawa Tengah. Pada penelitian ini dilakukan klasifikasi tenun berdasarkan motif daerahnya. Pertama, citra akan melalui pra-proses yaitu *cropping*, *grayscale*, dan normalisasi. Kemudian citra diekstraksi menggunakan *Gabor Filter*. Ekstraksi menghasilkan 40 vektor setiap citranya. Selanjutnya hasil ekstraksi diklasifikasi menggunakan Fuzzy C-Means. Klasifikasi menghasilkan vektor pusat yang kemudian diukur kemiripannya dengan vektor fitur citra uji Hasil pengujian menunjukkan bahwa sistem dapat mengenali citra sesuai dengan motif daerahnya dengan tingkat akurasi sebesar 94%.

Kata kunci : Tenun, *Gabor filter*, *Fuzzy C-Means*, Tekstur.

“Halaman ini sengaja dikosongkan”

APPLICATION OF FUZZY C-MEANS ALGORITHM FOR TENUN IMAGE CLASSIFICATION BASED ON TEXTURE FEATURE

Name : 'Isyi Bifawa'idati
NRP : 1212 100 043
Department : Mathematics
Supervisor : 1. Dr. Dwi Ratna S., S.Si., M.T.
2. Drs. Soetrisno, M.IKomp.

Abstract

Tenun is one of traditional fabric from Indonesia which has meanings, historical value, beautiful colour and design. Therefore, Tenun fabric preservation must be payed. The easy of access and process Tenun fabric data in computer is one of solution for that situation. If we get it, then it possible to outdrawn common interest, either from Indonesia or foreign countries. So, it can increase economics value of Indonesia. In this research, we used Tenun design from three region,namely Endek design from Bali, Lurik design from NTT, and Rangrang design from Center of Java. We classified Tenun based on region's design. First, image through pra-process, i.e. cropping, grayscalling, and normalization. Then, image extracted by using Gabor Filter. This extraction produced 40 vectors for each image. The result of this extraction classified by using Fuzzy C-Means. This classification produced the center vector that we measured its resemblance with image test feature vector. The result of this testing show that the system can identify the image according to its region's design and the accuration rate is 94%.

Keywords : Tenun, Gabor Filter, Fuzzy C-Means, Texture.

“Halaman ini sengaja dikosongkan”

KATA PENGANTAR

Segala Puji bagi Allah SWT Tuhan semesta alam yang telah memberikan karunia, rahmat dan anugerah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul: **“Penerapan Algoritma Fuzzy C-Means untuk Klasifikasi Citra Tenun Berdasarkan Fitur Tekstur”** yang merupakan salah satu persyaratan akademis dalam menyelesaikan Program Studi S-1 pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Institut Teknologi Sepuluh Nopember Surabaya.

Tugas Akhir ini dapat diselesaikan dengan berkat kerjasama, bantuan, dan dukungan dari banyak pihak. Sehubungan dengan hal itu, penulis mengucapkan terima kasih kepada:

1. Dr. Imam Mukhlas, S.Si, MT selaku Ketua Jurusan Matematika.
2. Dr. Dwi Ratna Sulistyaningrum, S.Si, MT dan Drs. Soetrisno, M.IKomp selaku dosen pembimbing yang senantiasa membimbing dengan sabar dan memberikan kritik dan saran sebagai masukan dalam penyusunan Tugas Akhir ini.
3. Drs. M. Setijo Winarko, M.Si selaku Dosen Wali dan Dosen penguji Tugas Akhir ini.
4. Drs. Daryono BU, M.Si dan Prof. Dr. Basuki W, M.Sc selaku dosen penguji Tugas Akhir ini.
5. Dr. Didik Khusnul Arif, S.Si, M.Si dan Drs. Iis Herisman, M.Sc selaku Koordinator Tugas Akhir.
6. Seluruh jajaran dosen dan staf jurusan Matematika ITS.
7. Teman-teman mahasiswa jurusan Matematika ITS

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan saran dan kritik dari pembaca. Akhir kata, semoga Tugas Akhir ini bermanfaat bagi semua pihak yang berkepentingan.

Surabaya, Januari 2017

Penulis

“Halaman ini sengaja dikosongkan”

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
DAFTAR LAMPIRAN	xxi
BAB I. PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan.....	4
1.5 Manfaat	4
1.6 Sistematika Penulisan	4
BAB II. TINJAUAN PUSTAKA	
2.1 Tenun.....	7
2.2 Pengolahan Citra Digital	9
2.3 Ciri Citra.....	9
2.4 Normalisasi.....	10
2.5 Konvolusi.....	11
2.6 <i>Gabor Filter</i>	11
2.7 Algoritma <i>Fuzzy</i>	14
2.8 <i>Fuzzy C-Means</i>	15
2.9 <i>Euclidean Distance</i>	17
BAB III. METODE PENELITIAN	
3.1 Tahapan Penelitian.....	19
3.2 Proses Utama Pada Penelitian.....	21
BAB IV. PERANCANGAN DAN IMPLEMENTASI	
4.1 Perancangan Sistem	23
4.1.1 Perancangan Data.....	23

4.1.2	Gambaran Proses Secara Umum.....	25
4.1.3	Pra-Proses	27
4.1.4	Proses Ekstraksi Fitur Citra dengan <i>Gabor Filter</i>	28
4.1.5	Proses Klasifikasi dengan <i>Fuzzy C-Means</i>	30
4.1.6	Proses Testing dengan Menghitung Kemiripan Citra Menggunakan <i>Euclidean Distance</i>	33
4.2	Implementasi	33
4.2.1	Antarmuka Penelitian	33
4.2.2	Implementasi Proses Akuisisi Citra Tenun ...	37
4.2.3	Implementasi Proses Normalisasi	38
4.2.4	Implementasi Proses Ekstraksi Ciri.....	38
4.2.5	Implementasi Proses Penyimpanan Tenun....	39
4.2.6	Implementasi Proses Kalsifikasi.....	40
4.2.7	Implementasi Proses Perhitungan Jarak	40
BAB V. PENGUJIAN DAN PEMBAHASAN HASIL		
5.1	Lingkungan Pengujian Sistem	41
5.2	Proses Pengujian Tahap Akuisisi Citra.....	41
5.3	Proses Pengujian Tahap Pra-proses	42
5.4	Pengujian Tahap Ekstraksi Ciri	44
5.4.1	Pengujian Hasil Implementasi Proses Filterisasi.....	44
5.4.2	Pengujian Hasil Implementasi Proses Perhitungan Vektor Ciri.....	45
5.5	Pengujian Tahap Klasifikasi dengan <i>Fuzzy C-Means</i>	45
5.6	Perbandingan Hasil Klasifikasi dengan <i>Magnitude Gabor Filter</i>	47
5.7	Pengujian Tahap Perhitungan Kemiripan Citra	47
5.8	Pengujian dengan Perubahan Parameter <i>Fuzzy C-Means</i>	49
5.9	Pembahasan Penyebab Citra Tidak Teridentifikasi Benar Sesuai Kelasnya.....	50

BAB VI. PENUTUP	
6.1 Kesimpulan	51
6.2 Saran	51
DAFTAR PUSTAKA	53

“Halaman ini sengaja dikosongkan”

DAFTAR GAMBAR

	Halaman
Gambar 2.1	Salah Satu Citra Tenun dari (a) NTT. (b) Bali. (c) Jawa. 8
Gambar 2.2	Gabor Kernel dengan 8 Orientasi dan 5 Frekuensi 13
Gambar 2.3	Proses Mendapatkan <i>Feature Vector</i> 14
Gambar 3.1	Diagram Alir Tahap Penelitian 21
Gambar 4.1	Diagram Alir Proses (a) Pelatihan dan (b) pengujian 27
Gambar 4.2	Diagram Alir Pra-Proses 28
Gambar 4.3	Diagram Alir Proses Ekstraksi Fitur Citra dengan Menggunakan Gabor Filter 29
Gambar 4.4	Diagram Alir Proses Klasifikasi <i>Fuzzy C-Means</i> 32
Gambar 4.5	Antarmuka Halaman Utama 34
Gambar 4.6	Antarmuka Pelatihan dan Pengujian 35
Gambar 5.1	Hasil Akuisisi Citra Tenun (a) Tenun Bali, (b) Tenun NTT, (c) Tenun Jawa Tengah 42
Gambar 5.2	(a) Citra Hasil Akuisisi, (b) Citra Hasil Cropping 43
Gambar 5.3	Citra Hasil <i>Grayscale</i> 43
Gambar 5.4	Hasil Filterisasi Citra Tenun 45

“Halaman ini sengaja dikosongkan”

DAFTAR TABEL

	Halaman
Tabel 4.1	Data Proses..... 24
Tabel 4.2	Kegunaan Menu Sistem 35
Tabel 5.1	Lingkungan Pengujian Sistem 41
Tabel 5.2	Hasil Percobaan Normalisasi dengan Parameter Berbeda 44
Tabel 5.3	Tabel Parameter <i>Fuzzy C-Means</i> 46
Tabel 5.4	Tabel Hasil Klasifikasi..... 46
Tabel 5.5	Hasil Klasifikasi dengan <i>Magnitude</i> Filter yang Berbeda 47
Tabel 5.6	Tabel pengujian dengan Pengulangan Sepuluh Kali pada Citra <i>Database</i> 48
Tabel 5.7	Tabel Pengujian dengan Pengulangan Sepuluh kali pada Citra Uji..... 49
Tabel 5.8	Tabel Rata-rata Akurasi Pengujian dengan Pengulangan..... 49
Tabel 5.9	Hasil Percobaan Klasifikasi dengan Kriteria Penghentian yang Berbeda..... 50

“Halaman ini sengaja dikosongkan”

DAFTAR LAMPIRAN

	Halaman
LAMPIRAN A	55
A. Kode Prosedur Akuisisi Citra Tenun...	55
LAMPIRAN B	59
B.1 Kode Fungsi Pra-Pemrosesan.....	59
B.2 Kode Fungsi Proses Normalisasi.....	59
LAMPIRAN C	79
C.1 Kode Fungsi Proses Filterisasi	61
C.2 Kode Fungsi Proses Pembuatan Filter Gabor.....	61
LAMPIRAN D	67
D. Kode Prosedur Proses Klasifikasi	67
LAMPIRAN E	69
E. Kode Fungsi Proses Perhitungan Kemiripan.....	69
LAMPIRAN F	71
F.1 Data Citra Tenun dari Bali	71
F.2 Data Citra Tenun dari Jawa Tengah	73
F.3 Data Citra Tenun dari NTT	75
F.4 Data Citra Tenun yang Telah Melalui Proses <i>Cropping</i>	77
LAMPIRAN G	79
G.1 Tabel Nilai Keanggotaan Masing- masing Citra pada Tiap Kelas.....	79
G.2 Tabel Vektor Pusat Kelas	81
G.3 Tabel Hasil Pengujian Pengulangan dengan Pengambilan Data Secara Acak pada Database	83
G.4 Tabel Hasil Pengujian Pengulangan dengan Pengambilan Data Secara Acak pada Uji.....	83

“Halaman ini sengaja dikosongkan”

BAB I

PENDAHULUAN

Pada bab ini dijelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat, serta sistematika penulisan dalam tugas akhir.

1.1. Latar Belakang

Indonesia memiliki banyak cara untuk mengenalkan budayanya, dan salah satunya yaitu lewat kain tenun. Tenun memiliki makna, nilai sejarah, dan teknik yang tinggi baik dari segi warna, motif, jenis bahan dan benang yang digunakan serta tiap daerah memiliki ciri khas masing-masing. Proses pembuatan kain tenun yaitu dengan memasukan benang pakan secara horizontal pada benang-benang lungsi, biasanya telah diikat dahulu dan sudah dicelupkan ke pewarna alami. Sebagian besar daerah di Indonesia memiliki ciri dan motif kain tenun sendiri diantaranya Jawa Barat, Jawa Tengah, DI Yogyakarta, Bali, Kalimantan, Sulawesi, dan NTT[1].

Tenun menjadi salah satu kain tradisional Indonesia untuk mengenalkan budaya Indonesia ke mancanegara. Oleh karena itu, pelestarian kain tenun harus sangat diperhatikan. Salah satu yang dapat menangani hal tersebut adalah adanya kemudahan dalam mengakses dan mengolah data pada media komputer. Internet memiliki pengaruh besar dalam penyampaian informasi. Jika terdapat kemudahan dalam akses penyebaran kain tenun, maka nilai ekonomi Indonesia pun dapat meningkat seiring semakin luas dikenalnya kain tenun di mancanegara.

Pada awalnya, teknik penyimpanan data hanya tersimpan secara konvensional. Begitu pula pada teknik penyimpanan data citra motif tenun. Hal ini menyebabkan data menjadi tidak permanen

dan terbatas. Kemudian, mengakibatkan data menjadi tidak teratur sesuai dengan kelompoknya masing-masing. Permasalahan tentang cara menyimpan data citra motif tenun dalam jumlah besar telah memiliki solusi, salah satunya adalah penggunaan *database*. Namun, meskipun telah tersimpan secara permanen dan tidak terbatas, citra dalam *database* masih belum bisa terkelompokkan dengan baik.

Saat ini, perkembangan teknologi telah banyak memudahkan penggunaannya. Setelah data dapat tersimpan dalam jumlah besar, juga dapat dikelompokkan dengan baik secara terurut. Terdapat metode klasifikasi *database* dalam jumlah besar dengan waktu singkat. Juga terdapat metode *clustering* yang dapat mengelompokkan data meskipun tidak diketahui bagaimana data tersebut harus dikelompokkan. Jika pada klasifikasi, terdapat informasi bagaimana data tersebut dikelompokkan[2].

Penelitian tentang klasifikasi citra berdasarkan tekstur telah dilakukan oleh Thanh Nguyen dkk. pada tahun 2014. Penelitian tersebut berjudul “*Medical Data Classification Using Interval Type-2 Fuzzy Logic System and Wavelets*”. Penelitian ini mengklasifikasikan data medis dengan tahapan pertama yaitu mengekstraksi fitur citra menggunakan *Wavelet*. Tahap kedua, dilakukan *training Interval Type-2 Fuzzy Logic System* yaitu *unsupervised structure learning* dengan *Fuzzy C-Means Clustering* dan *supervised parameter tuning* dengan *Genetic Algorithm*. Kemudian dihasilkan klasifikasi pasien berdasarkan tipe kanker yang diderita.

Selain penelitian tersebut, terdapat juga penelitian milik Claudio A. Perez dkk. pada tahun 2014 dengan judul “*Rock Lithological Classification Using Multi-Scale Gabor Features from Sub-Images, and Voting With Rock Contour Information*”. Pada penelitian ini, dilakukan ekstraksi fitur berdasarkan tekstur menggunakan *Gabor Filter*. Setelah ekstraksi fitur, setiap sub-image

diklasifikasikan berdasarkan kelas ukiran batu menggunakan *Support-Vector Machine* (SVM).

Berdasarkan beberapa permasalahan di atas, akan dilakukan penelitian untuk melakukan klasifikasi citra tenun berdasarkan tekstur. Penelitian ini akan dilakukan dengan tahap ekstraksi fitur terlebih dahulu dengan *Gabor Filter* kemudian dilakukan klasifikasi dengan Algoritma *Fuzzy C-Means*. Sehingga penelitian ini diberi judul “Penerapan Algoritma *Fuzzy C-Means* untuk Klasifikasi Citra Tenun Berdasarkan Fitur Tekstur”.

1.2. Rumusan Masalah

Berdasarkan urutan pada bagian latar belakang masalah, maka dapat dirumuskan beberapa permasalahan sebagai berikut:

1. Bagaimana perlunya analisis yang detail tentang ekstraksi fitur citra kain tenun berbasis tekstur dengan *Gabor Filter*?
2. Bagaimana melakukan klasifikasi hasil ekstraksi fitur citra kain tenun dengan Algoritma *Fuzzy C-Means*?

1.3. Batasan Masalah

Batasan masalah yang diberikan sebagai asumsi dan ruang lingkup pada Tugas Akhir ini adalah:

1. Citra uji dan citra *database* yang digunakan berupa citra kain tenun dari Bali, Jawa Tengah, dan NTT.
2. Ukuran citra tenun yang akan di ekstraksi dan disimpan dalam dataset memiliki ukuran piksel yang sama yaitu 300×300 .
3. Metode yang digunakan pada penelitian ini adalah *Gabor Filter* dan *Fuzzy C-Means*.
4. *File* yang digunakan berbentuk JPG atau JPEG.

1.4. Tujuan

Tujuan yang akan dicapai pada Tugas Akhir ini adalah menganalisa dan mengklasifikasi citra kain tenun berdasarkan motif yang menjadi khas daerahnya. Yang dilakukan adalah mengekstraksi citra menggunakan *Gabor Filter* yang kemudian hasil ekstraksi tersebut diklasifikasikan menggunakan algoritma *Fuzzy C-Means*.

1.5. Manfaat

Manfaat yang dapat diperoleh dari penelitian ini diharapkan mampu memberikan informasi tenun berdasarkan klasifikasi daerahnya agar dapat menjadi pembelajaran kebudayaan pada bangsa Indonesia sendiri maupun turis asing yang datang ke Indonesia sehingga dapat melestarikan warisan motif tenun. Selain itu, dapat mendukung peningkatan nilai ekonomi dengan menjadi semakin mudahnya akses dan pengolahan data di internet.

1.6. Sistematika Penulisan

Tugas Akhir ini disusun berdasarkan sistematika penulisan sebagai berikut:

BAB I PENDAHULUAN

Berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan pada tugas akhir.

BAB II TINJAUAN PUSTAKA

Menjelaskan dasar teori yang digunakan penulis dalam mengerjakan tugas akhir. Pada bab ini berisi tentang tenun, citra digital secara umum, bentuk umum konvolusi, *Gabor Filter*, *Fuzzy C-means*, dan *Euclidean Distance*.

- BAB III METODE PENELITIAN**
Menjelaskan alur kerja dan metode yang digunakan penulis dalam mengerjakan tugas akhir.
- BAB IV PERANCANGAN SISTEM DAN IMPLEMENTASI**
Bab ini menjelaskan uraian mengenai perancangan dan implementasi sistem, proses pembuatan sistem secara utuh sehingga dapat digunakan untuk melakukan klasifikasi tenun.
- BAB V HASIL DAN PEMBAHASAN PENGUJIAN**
Bab ini menjelaskan mengenai hasil pengujian implementasi sistem dan hasil pengujian sistem dalam mengklasifikasi jenis-jenis tenun.
- BAB VI PENUTUP**
Berisi kesimpulan dari hasil analisa dan pembahasan yang telah dilakukan.

“Halaman ini sengaja dikosongkan”

BAB II

TINJAUAN PUSTAKA

Pada bab ini dibahas teori-teori yang terkait dengan permasalahan dalam Tugas Akhir ini. Pertama, membahas mengenai

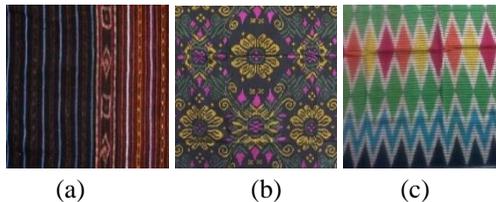
2.1 Tenun

Tenun merupakan salah satu warisan budaya Indonesia yang memiliki makna pada setiap motif daerah di seluruh Nusantara. Tenun memiliki makna, nilai sejarah, dan teknik yang tinggi baik dari segi warna, motif, jenis bahan dan benang yang digunakan. Tiap daerah memiliki ciri khas masing-masing. Proses pembuatan kain tenun yaitu dengan memasukan benang pakan secara horizontal pada benang-benang lungsi. Biasanya benang tersebut telah diikat dahulu dan sudah dicelupkan ke pewarna alami. Pewarna alami tersebut biasanya dibuat dari akar-akar pohon atau ada pula yang menggunakan dedaunan[3].

Jenis tenun dihasilkan dari peralatan ataupun teknik yang dipergunakan dalam menenun benang lungsi dan benang pakan. Benang lungsi adalah benang yang terletak memanjang pada alat tenun. Sedangkan benang pakan adalah benang yang masuk keluar pada lungsi saat menenun. Beberapa jenis tenun adalah tenun sederhana, tenun ikat lungsi, tenun ikat pakan, tenun ikat ganda, tenun songket, dan tenun dobby. Tenun sederhana merupakan jenis tenun yang polos, tidak memiliki corak atau corak garis-garis, kotak-kotak sesuai dengan warna dan jenis benang yang dipakai. Sehingga menghasilkan tenunan yang disebut tenun lurik (garis-garis) atau tenun poleng (kotak-kotak). Tenun ikat lungsi merupakan jenis tenun dengan benang lungsi yang memiliki berbagai warna. Tenun ikat pakan merupakan jenis

tenun dengan benang pakan yang memiliki berbagai warna. Tenun ikat ganda menggunakan benang lungsi dan benang pakan yang memiliki berbagai warna sehingga corak terbentuk dari persilangan kedua benang. Tenun ikat songket adalah tenun dengan teknik menyisipkan benang perak, emas, tembaga atau benang warna di atas benang lungsi. Tenun dobby dihasilkan melalui pola yang dibuat pada sebuah susunan kayu selebar 2 cm x 20 cm[4].

Setiap daerah pasti memiliki ciri khas motif tenun masing-masing. Tenun sederhana banyak dijumpai di daerah Jawa Barat, Jawa Tengah, DI Yogyakarta, Sulawesi Tenggara. Tenun ikat lungsi banyak dijumpai dari daerah NTB, NTT, Maluku, Kalimantan Timur, Kalimantan Barat, Sulawesi Barat, Sulawesi Utara, Papua Barat. Tenun ikat pakan banyak dijumpai dari daerah Bengkulu, Sumatera Selatan, Jawa Barat, Jawa Tengah, Jawa Timur, Bali, NTB, Kalimantan Barat, Kalimantan Timur, Sulawesi Selatan, Sulawesi Tengah. Hasil tenun ikat ganda dapat dijumpai dari Bali (Tenganan), Sulawesi Tengah, dan Sulawesi Tenggara. Tenun songket banyak terdapat di daerah Aceh, Sumatera Utara, Sumatera Barat, Jambi, Bengkulu, Sumatera Selatan, Lampung, Banten, Jawa Barat, Jawa Timur, Bali, NTB, NTT, Kalimantan Barat, Kalimantan Timur, Sulawesi Selatan, Maluku Utara. Tenun dobby banyak diproduksi di daerah Jawa Barat, Jawa Tengah[4].



Gambar 2.1 Salah Satu Citra Tenun dari (a) NTT. (b) Bali. (c) Jawa Tengah.

2.2 Pengolahan Citra Digital

Citra merupakan media informasi yang tersusun atas kombinasi warna. Suatu informasi citra dapat dianalisis dengan cara pengelompokan berdasarkan persamaan pada bentuk dimensi citra maupun warna. Citra yang mengandung banyak unsur warna dapat dikelompokkan ke dalam beberapa warna dasar tertentu. Bagian warna-warna pada citra yang memiliki kemiripan terhadap warna dasar yang ditentukan, dapat diarahkan untuk menjadi bagian pada kelompok warna dasar tersebut[5].

Citra digital direpresentasikan sebagai fungsi dua dimensi $f(x,y)$, dengan nilai x dan y masing-masing merupakan koordinat spasial dan nilai fungsi pada setiap titik (x,y) merupakan intensitas citra atau tingkat keabuan pada titik koordinat tersebut[5]. Proses pengolahan citra pada penelitian ini terdiri dari beberapa tahapan, yang masing-masing prosesnya saling terkait.

2.3 Ciri Citra

Ciri merupakan suatu tanda yang khas, yang membedakan antara satu dengan yang lain. Citra juga memiliki ciri yang dapat membedakannya dengan citra yang lain. Masing-masing ciri citra didapatkan dari proses ekstraksi ciri. Ciri – ciri dasar dari citra sebagai berikut [6]:

1. Warna

Ciri warna suatu citra dapat dinyatakan dalam bentuk histogram dari citra tersebut yang dituliskan dengan: $H(r, g, b)$, dengan (r, g, b) adalah jumlah munculnya pasangan warna r (*red*), g (*green*) dan b (*blue*) tertentu.

2. Bentuk

Ciri bentuk suatu citra dapat ditentukan oleh tepi (sketsa), atau besaran *moment* dari suatu citra. Pemakaian besaran

moment pada ciri bentuk ini banyak digunakan orang dengan memanfaatkan nilai-nilai transformasi *fourier* dari citra. Proses yang dapat digunakan untuk menentukan ciri bentuk adalah deteksi tepi, *threshold*, segmentasi, dan perhitungan *moment* seperti (*mean*, *median* dan standard deviasi dari setiap lokal citra).

3. Tekstur

Tekstur merupakan karakteristik intrinsik dari suatu citra yang terkait dengan tingkat kekasaran (*roughness*), granularitas (*granulation*), dan keteraturan (*regularity*) susunan struktural piksel. Aspek tekstural dari sebuah citra dapat dimanfaatkan sebagai dasar dari segmentasi, klasifikasi, maupun interpretasi citra.

2.4 Normalisasi

Normalisasi dibentuk untuk menghilangkan efek *noise sensor* dan deformasi derajat keabuan sebagai akibat dari ketidakseragaman pencahayaan. Citra yang telah dinormalisasi didefinisikan sebagai :

$$N(x, y) = \begin{cases} M_0 + \sqrt{\frac{V_0\{(I(x, y) - M)^2\}}{V}} & \text{jika } I(x, y) > M \\ M_0 - \sqrt{\frac{V_0\{(I(x, y) - M)^2\}}{V}} & \text{jika } I(x, y) < M \end{cases} \quad (1)$$

dengan:

$N(x, y)$ adalah derajat nilai keabuan yang telah dinormalisasi pada piksel (x, y) .

$I(x, y)$ adalah nilai intensitas pada piksel (x, y)

M adalah nilai rata-rata intensitas pada citra.

M_0 adalah nilai rata-rata intensitas yang ditetapkan/diinginkan.

V adalah nilai varian intensitas piksel pada citra.

V_0 adalah nilai varian intensitas piksel yang diinginkan [8]

2.5 Konvolusi

Konvolusi didefinisikan sebagai cara untuk mengkombinasikan dua buah deret angka yang menghasilkan deret angka yang ketiga. Operasi konvolusi dilakukan dengan menggeser kernel konvolusi piksel per piksel. Hasil konvolusi disimpan dalam matrik yang baru. Konvolusi dari dua buah fungsi $f(x)$ dan $g(x)$, dinotasikan dengan $f(x) * g(x)$. Dan untuk konvolusi dua dimensi didefinisikan sebagai :

$$\begin{aligned} h(x, y) &= f(x, y) * g(x, y) \\ &= \sum_{a=-\infty}^{\infty} \sum_{b=-\infty}^{\infty} f(a, b) g(x - a, y - b) \end{aligned}$$

Fungsi $g(x, y)$ merepresentasikan Gabor Kernel merupakan suatu jendela yang dioperasikan secara bergeser pada piksel citra masukan $f(x, y)$, dalam hal ini jumlah perkalian kedua fungsi pada setiap titik merupakan hasil konvolusi yang dinyatakan dengan keluaran $h(x, y)$. Proses konvolusi dibutuhkan untuk perbaikan kualitas citra (*image enhancement*), penghilang derau (*noise*), mengurangi erotan (serong), penghalusan / pembulatan citra, dll.

2.6 Gabor Filter

Fungsi *Gabor* pertama kali diperkenalkan oleh Denis Gabor pada tahun 1946 sebagai *tools* untuk deteksi sinyal dalam *noise*. Daugman mengembangkan kerja *Gabor* kedalam *filter* dua dimensi pada tahun 1980. *Gabor Filter* adalah *filter* linier yang digunakan dalam pengekstrasian fitur citra sebagai detektor ciri. *Gabor Filter* dikenal sebagai detektor ciri yang sukses karena memiliki kemampuan menghilangkan variabilitas yang

disebabkan oleh iluminasi kontras dan sedikit pergeseran serta deformasi citra. *Output Gabor Filter* telah banyak digunakan dengan sukses untuk pengenalan wajah pada penelitian sebelumnya[7].

Fungsi Gabor merupakan sinusoid kompleks yang dipengaruhi oleh fungsi Gaussian. *Gabor bank Filter* adalah multi-channel filter yang menggunakan frekuensi spasial tertentu dan orientasi dari sinusoidal dan Gaussian menyebar ke arah x dan y (σ_x dan σ_y). Gelombang *Sinusoidal* dinyatakan pada persamaan berikut [7]:

$$s(x, y) = \exp[2j\pi(ux \cos \theta + uy \sin \theta)] \quad (2)$$

Fungsi *Gaussian* dua dimensi didefinisikan dalam persamaan berikut :

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\} \quad (3)$$

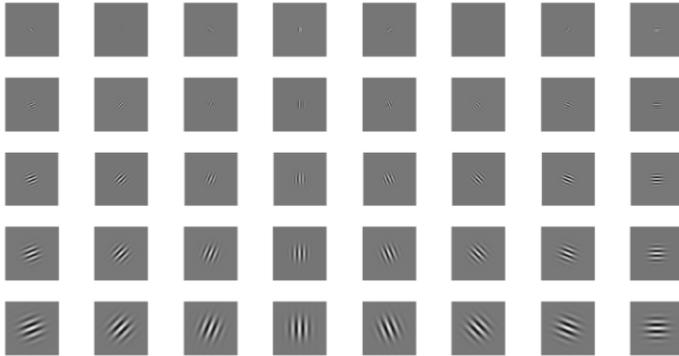
Dari persamaan (2) dan (3), fungsi dasar *Gabor Kernel* dapat didefinisikan sebagai berikut :

$$G(x, y) = g(x, y) s(x, y)$$

$$G(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \cdot \exp[2j\pi(ux \cos \theta + uy \sin \theta)]$$

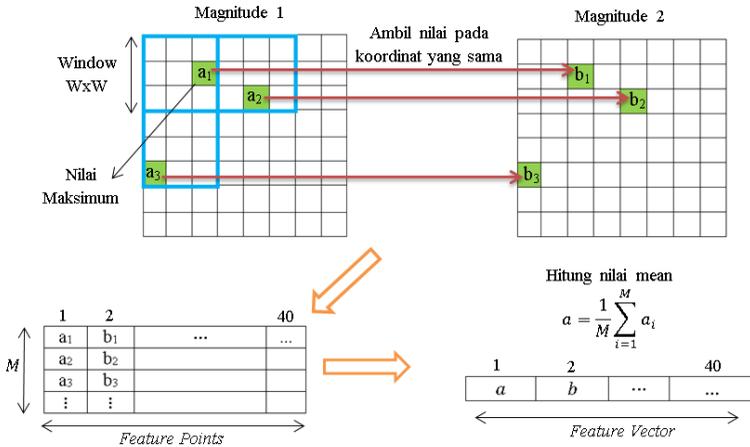
Dengan σ_x dan σ_y merupakan sebaran Gaussian ke arah x dan y berturut-turut. Diasumsikan σ_x dan σ_y sama, u_0 merupakan frekuensi tengah dari sinusoid dan orientasi dari sinusoid[7].

Persamaan Gabor Kernel tersebut kemudian di konvolusi dengan nilai piksel citra yang telah di *Grayscale*. Setelah citra di konvolusi dengan gabor kernel maka terbentuklah 40 magnitudo.



Gambar 2.2 *Gabor Kernel* dengan 8 Orientasi dan 5 Frekuensi [8]

Magnitude pertama dicari nilai maksimum dengan prosedur menempatkan window W_0 dengan ukuran $W \times W$. Pada setiap window berukuran $W \times W$, dicari nilai maksimumnya. Nilai maksimum tersebut merupakan *feature points*. Kemudian koordinat dari tiap nilai maksimum pada magnitude pertama akan di proyeksikan ke magnitude 2 sampai 40. Sehingga nilai yang diambil dari magnitude 2 sampai 40 adalah nilai yang berada pada koordinat yang sama dengan koordinat nilai maksimum pada magnitude pertama. Setelah didapatkan *feature points* maka disusun untuk membentuk suatu *feature vector* dengan menghitung mean dari *feature points* tiap magnitude[8]. Ilustrasinya dapat dilihat pada Gambar 2.3



Gambar 2.3 Proses Mendapatkan *feature vector*

2.7 Algoritma *Fuzzy*

Secara prinsip, di dalam teori *Fuzzy set* dapat dianggap sebagai ekstension dari teori konvensional atau “crisp” *set*. Di dalam teori “crisp” *set*, suatu elemen hanya dapat digolongkan sebagai anggota atau bukan anggota dari suatu set atau himpunan. Sehingga di dalam teori “crisp” *set*, suatu elemen yang merupakan anggota mempunyai tingkat keanggotaan (*membership level*) penuh atau satu dan suatu elemen yang bukan anggota mempunyai tingkat keanggotaan nol. Misal, jika set A adalah merupakan suatu himpunan bilangan real X dan $x \in X$, maka secara matematis tingkat keanggotaan suatu elemen x di dalam set A dapat dinyatakan dengan persamaan berikut[9]:

$$\mu_A(x) = \begin{cases} 1.0 & \text{jika } x \in A \\ 0.0 & \text{jika } x \notin A \end{cases}$$

dengan $\mu_A(x)$ menunjukkan tingkat keanggotaan elemen x di dalam set A . Dalam hal ini dinyatakan bahwa tingkat keanggotaan suatu elemen hanya dikenal sebagai 1.0 (anggota penuh) atau 0.0

(sama sekali bukan anggota), sehingga di dalam “crisp” set, tingkat keanggotaannya dinyatakan sebagai pemetaan ke 0 dan 1 yang secara matematis dinotasikan sebagai $\mu_A(x) \rightarrow \{0,1\}$ [9].

Akan tetapi, di dalam teori *Fuzzy set* dikenal adanya keanggotaan secara parsial. Dalam hal ini maka tingkat keanggotaan suatu elemen di didalam suatu set merupakan fungsi kontinu dari 0.0 sampai 1.0. Sehingga pemetaan tingkat keanggotaan pada teori *Fuzzy set* dapat dinotasikan sebagai $\mu_A(x) \rightarrow [0,1]$. Sebagai misal, jika A merupakan set atau himpunan bilangan real yang “dekat” dengan bilangan nol. Secara “crisp” akan sulit atau paling tidak akan sangat subjektif untuk menentukan bilangan-bilangan mana yang “dekat” dengan bilangan nol[9].

2.8 *Fuzzy C-Means*

Metode *Fuzzy C-Means Clustering* pertama kali dikenalkan oleh Jim Bezdek pada tahun 1981. Klasifikasi *Fuzzy* menjawab masalah ketidakpastian dalam melakukan ekstraksi. Dalam klasifikasi *Fuzzy*, setiap piksel mempunyai m nilai keanggotaan antara 0 dan 1 pada saat ditetapkan menjadi satu dari m kelas. Ada dua jenis klasifikasi *Fuzzy* yaitu klasifikasi *Fuzzy* terawasi (*Supervised Classification*) yang berbasis training area seperti *Fuzzy Maximum Likelihood* dan *Fuzzy* tak terawasi (*Unsupervised Classification*) yang berbasis *clustering* seperti *Fuzzy C-Means* (FCM). FCM adalah salah satu metode *Fuzzy Clustering* yang fungsi keanggotaannya didefinisikan berdasarkan fungsi jarak, sehingga derajat keanggotaan suatu piksel didefinisikan berdasarkan seberapa dekat jaraknya terhadap pusat kelas. *Fuzzy C-Means* adalah salah satu teknik pengelompokan data yang mana keberadaan tiap titik data dalam suatu kelompok (*cluster*) ditentukan oleh derajat keanggotaan. FCM bersifat

sederhana mudah diimplementasikan, dan memiliki kemampuan untuk mengelompokkan data yang besar[10].

Konsep dasar FCM, pertama kali adalah menentukan pusat cluster yang akan menandai lokasi rata-rata untuk tiap-tiap cluster. Tiap-tiap data memiliki derajat keanggotaan untuk tiap-tiap cluster. Dengan cara memperbaiki pusat cluster dan nilai keanggotaan tiap-tiap data secara berulang, maka dapat dilihat bahwa pusat cluster akan menuju lokasi yang tepat. Perulangan ini didasarkan pada minimasi fungsi obyektif[11].

$$J(U, V; X) = \sum_{k=1}^n \sum_{i=1}^C (\mu_{ik})^w (d_{ik})^2$$

dengan $w \in [1, \infty]$

$$d_{ik} = d(x_k - v_i) = \left[\sum_{j=1}^m (x_{kj} - v_{ij})^2 \right]^{\frac{1}{2}}$$

X adalah data yang dicluster

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{bmatrix}$$

Algoritma FCM secara lengkap diberikan sebagai berikut :

1. Tentukan :
 - a. Matriks X berukuran $n \times m$, dengan n = jumlah data yang akan di cluster, dan m = jumlah variabel (kriteria).
 - b. Jumlah cluster yang akan dibentuk ($C \geq 2$)
 - c. Pangkat (pembobot $w > 1$)
 - d. Maksimum iterasi
 - e. Kriteria penghentian (ε = nilai positif yang sangat kecil)
2. Bentuk matriks partisi awal U (derajat keanggotaan dalam cluster); matriks partisi awal biasanya dibuat secara acak.

$$U = \begin{bmatrix} \mu_{11}(x_1) & \mu_{12}(x_2) & \cdots & \mu_{1n}(x_n) \\ \mu_{21}(x_1) & \mu_{22}(x_2) & \cdots & \mu_{2n}(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{c1}(x_1) & \mu_{c2}(x_2) & \cdots & \mu_{cn}(x_n) \end{bmatrix}$$

3. Hitung pusat cluster V untuk setiap cluster.

$$v_{ij} = \frac{\sum_{k=1}^n (\mu_{ik})^w x_{kj}}{\sum_{k=1}^n (\mu_{ik})^w}$$

4. Perbaiki derajat keanggotaan setiap data pada setiap cluster (perbaiki matriks partisi).

$$\mu_{ik} = \left[\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{(w-1)}} \right]^{-1}$$

dengan

$$d_{ik} = d(x_k - v_i) = \left[\sum_{j=1}^m (x_{kj} - v_{ij}) \right]^{\frac{1}{2}}$$

5. Tentukan kriteria penghentian iterasi, yaitu perubahan matriks partisi pada iterasi sekarang dan iterasi sebelumnya. Apabila $\Delta < \varepsilon$ maka iterasi dihentikan.

$$\Delta = \|U^t - U^{t-1}\| \quad (4)$$

2.9 Euclidean Distance

Ruang *Euclidean* merupakan ruang dengan dimensi terbatas yang bilangannya bernilai riil. Dalam ruang tersebut anggap saja terdapat dua titik dua dimensi yang memiliki koordinat masing-masing. Dalam citra, digunakan mengukur kemiripan dua vektor ciri citra. Semakin kecil nilai $d(x, y)$, maka semakin mirip kedua vektor yang dicocokkan/dibandingkan. Sebaliknya semakin besar nilai $d(x, y)$ maka semakin berbeda kedua vektor yang dicocokkan. Jarak *Euclidean* x ke y adalah:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

dengan x adalah vektor ciri citra training, dan y adalah vektor ciri citra input test. Jika $x = (x_1, x_2, x_3, \dots, x_n)$ dan $y = (y_1, y_2, y_3, \dots, y_n)$ merupakan dua titik dalam Euclidean ruang n [8].

BAB III

METODE PENELITIAN

Pada bab ini dijelaskan metode yang digunakan dalam penelitian agar proses pengerjaan dapat terstruktur dengan baik dan dapat mencapai tujuan yang telah ditetapkan sebelumnya. Pembahasan metode penelitian diawali dengan penjelasan tentang tahapan penelitian dan blok diagram.

Proses pengerjaan terdiri dari lima tahap, yaitu studi literatur, pengumpulan data, perancangan aplikasi, uji coba dan evaluasi, serta penarikan kesimpulan dan penulisan tugas akhir.

3.1 Tahapan Penelitian

Dalam melakukan penelitian pada tugas akhir yang diusulkan ini, ada beberapa tahap yang akan dilakukan, yaitu :

1. Studi Literatur

Dari permasalahan dan tujuan yang telah dirumuskan, selanjutnya dilakukan studi literatur untuk mendukung pengerjaan tugas akhir. Studi literatur dilakukan terhadap jurnal-jurnal ilmiah, tugas akhir, dan buku-buku penunjang, serta referensi dari internet yang terkait dengan *Gabor Filter* dan Algoritma *Fuzzy*, seperti yang telah tercantum dalam daftar pustaka.

2. Pengumpulan Data

Pada tahap pengumpulan data, dilakukan dengan mengumpulkan beberapa informasi yang dibutuhkan dalam pengerjaan penelitian. Pada penelitian tugas akhir ini membutuhkan data citra kain tenun dari tiga daerah yang telah ditentukan. Beberapa data citra dikumpulkan dari koleksi citra tenun milik organisasi Cita Tenun Indonesia

dalam situs web nya www.citatenunindonesia.com dan dari situs tenun yang lain.

3. Perancangan dan Implementasi Aplikasi

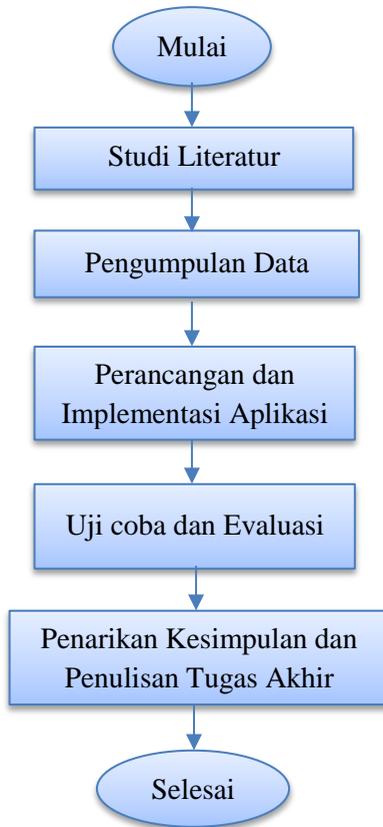
Pada tahap perancangan aplikasi, mulai dilakukan perancangan program yang dapat mengklasifikasi citra tenun sesuai dengan motif daerahnya. Ada beberapa tahap yang akan dilakukan yaitu, pra-proses data, ekstraksi citra, dan klasifikasi hasil ekstraksi citra. Kemudian pada tahap implementasi, dibagi menjadi dua tahap yaitu proses *training* dan *testing*.

4. Uji Coba dan Evaluasi

Pada tahap uji coba, dilakukan untuk melakukan simulasi terhadap aplikasi. Pada tahap ini, dilakukan dengan perhitungan kemiripan citra uji coba dengan citra pusat pada kelas yang telah diklasifikasi.

5. Penarikan Kesimpulan dan Penulisan Tugas Akhir

Pada tahap ini, dilakukan penarikan kesimpulan dari penelitian yang telah dilakukan. Kemudian dari kesimpulan tersebut juga dilakukan pendaftaran beberapa saran untuk penelitian selanjutnya. Semua tahap-tahap yang dilakukan pada penelitian, dijelaskan pada penulisan Tugas Akhir.



Gambar 3.1 Diagram Alir Tahapan Penelitian

3.2 Proses Utama Pada Penelitian

Beberapa proses utama dalam yang akan dilakukan pada penelitian tugas akhir ini adalah :

1. Pra-Proses

Data yang digunakan yaitu kumpulan citra kain tenun yang berasal dari tiga daerah yang memiliki ciri khas tekstur

berbeda pula. Data yang diperoleh berasal dari web organisasi Cita Tenun Indonesia . Proses awal yang dilakukan adalah *cropping* citra dengan ukuran piksel yang rata sama untuk semua citra latih dan citra uji. *Cropping* citra dilakukan dengan mengambil bagian pada citra tenun yang memiliki keteraturan pola. Selanjutnya dilakukan proses *grayscale* dan normalisasi.

2. Ekstraksi Fitur Ciri Menggunakan *Gabor Filter*

Pada tahap ini dilakukan ekstraksi fitur. Fitur yang digunakan adalah fitur tekstur. Ekstraksi fitur citra menggunakan *Gabor Filter*. Hasil dari ekstraksi fitur adalah vektor fitur dari citra hasil pre-proses. Kemudian vektor fitur tersebut disimpan pada *database* fitur.

3. Klasifikasi Hasil Ekstraksi Menggunakan *Fuzzy C-Means*

Pada tahap ini dilakukan klasifikasi dengan menggunakan *Fuzzy C-Means*. Setelah didapat vektor-vektor fitur tekstur, maka dilakukan proses pengelompokan data (*data clustering*). Data clustering adalah proses membagi elemen-elemen data ke klaster-klaster, sehingga elemen yang memiliki kesamaan ciri akan dikelompokkan pada satu kelas, dan terpisah dengan elemen yang tidak memiliki kesamaan ciri.

4. Perhitungan Kemiripan Ciri Citra

Tahap ini merupakan tahap uji coba aplikasi. Pada tahap ini, akan dihitung kemiripan ciri antara vektor uji coba dengan vektor pusat kelas pada masing-masing kelas. Untuk menghitung kemiripan ciri, menerapkan *Euclidean Distance*.

BAB IV

PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini menjelaskan mengenai perancangan sistem dan hasil implementasi semua proses yang telah dirancang sebelumnya. Pembahasan perancangan sistem diawali dengan penjelasan tentang lingkungan perancangan sistem, Gambaran kalsifikasi jenis-jenis tenun berdasarkan daerahnya dengan menggunakan metode *Fuzzy C-Means* secara umum. Selanjutnya membahas implementasi sistem yang dimulai dari lingkungan implementasi sistem dan dilanjutkan dengan hasil implementasi antarmuka dan keseluruhan proses di dalam sistem.

4.1 Perancangan Sistem

Tampilan dari sistem klasifikasi jenis tenun dibangun dengan tampilan yang sederhana. *Software* yang digunakan untuk membangun sistem klasifikasi jenis tenun adalah Matlab.

4.1.1 Perancangan Data

Data yang digunakan dalam sistem klasifikasi tenun dengan menggunakan Gabor Filter dibagi menjadi 3 macam, yaitu data masukan, data proses, dan data keluaran. Data masukan merupakan data input dari pengguna sistem. Data proses adalah data ketika tahap-tahap pengklasifikasian sedang dilakukan. Sedangkan data keluaran adalah data yang ditampilkan kepada pengguna sistem.

a. Data masukan

Data masukan merupakan data yang dimasukkan oleh pengguna sistem berupa citra tenun yang telah diakuisisi. Citra tenun disimpan dalam bentuk *.jpg* atau *.jpeg*. Citra tenun yang diambil terdapat 60 citra *database* dan 15 citra uji.

Sehingga total data citra tenun terdapat 75 citra. Dengan rincian 25 citra tenun Bali, 25 citra tenun NTT, dan 25 citra tenun Jawa Tengah. Citra tenun yang diambil dari beberapa situs di internet memiliki ukuran Gambar yang berbeda-beda.

b. Data proses

Data proses merupakan data masukan yang akan diproses menjadi data yang digunakan untuk menjalankan sistem perangkat lunak ini.

Tabel 4.1 Data Proses

Nama Data	Tipe Data	Keterangan
Citra <i>cropping</i>	Uint8	Data ini berupa nilai piksel hasil <i>cropping</i> citra pada bagian terbaik dari keseluruhan citra.
Citra RGB	Uint8	Data ini berupa matriks representasi citra tenun RGB dengan ukuran matriks citra 300x300x3.
Citra <i>Grayscale</i>	Uint8	Data ini berupa nilai piksel hasil proses citra keabuan dengan ukuran matriks citra 300x300
Citra ternormalisasi	Uint8	Data ini berupa nilai piksel citra keabuan hasil normalisasi dengan ukuran matriks citra 300x300
Matriks Citra Terfilter	Double	Data ini merupakan hasil proses konvolusi citra <i>grayscale</i> dan <i>Filter Gabor</i> . Berupa matriks 25x25.
Vektor fitur	Double	Data ini berupa vektor fitur hasil ekstraksi ciri energi citra terfilter
Vektor pusat	Double	Data ini berupa vektor tiap pusat kluster
Matriks U	Double	Data ini berupa matriks derajat keanggotaan yang menentukan kluster tiap data

c. Data keluaran

Data keluaran yang dihasilkan dari sistem ini adalah berupa deskripsi apakah citra tenun tersebut merupakan citra tenun daerah Jawa Tengah, NTT, atau Bali. Deskripsi tersebut berdasar pada kemiripan ciri citra uji dengan nilai pusat dari masing-masing kelas. Dari ketiga pusat kelas, yang memiliki nilai kemiripan yang paling dekat dengan fitur ciri citra uji tersebutlah yang menentukan citra uji berada di kelas mana. Sehingga keluaran data berupa informasi identitas citra uji berasal dari daerah mana.

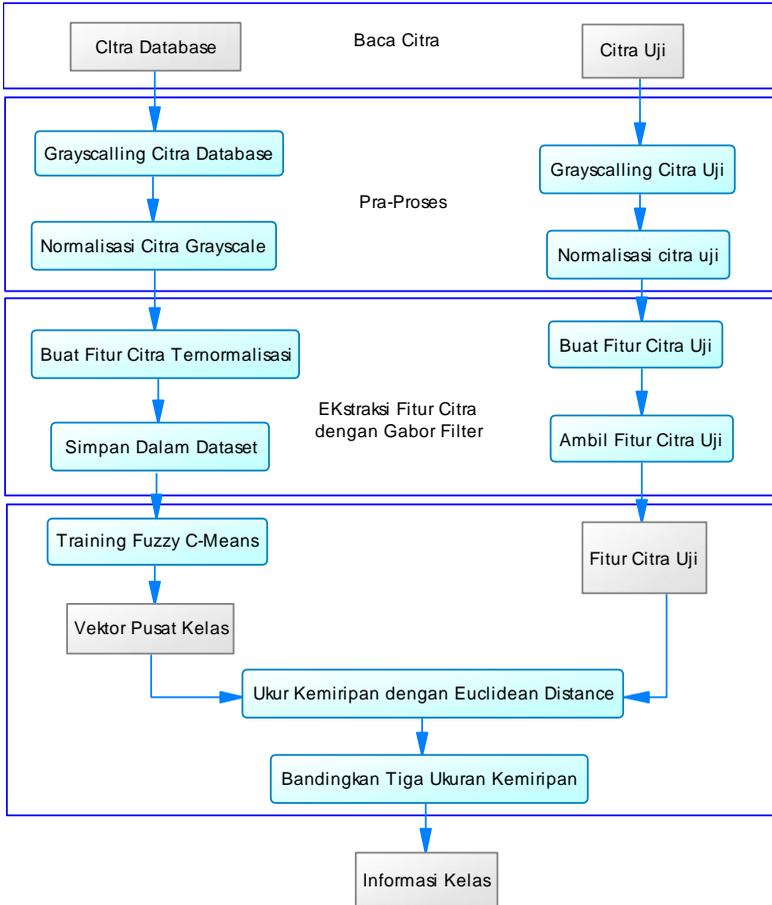
4.1.2 Gambaran Proses Secara Umum

Gambaran proses secara umum merupakan Gambaran keseluruhan proses yang dilakukan sistem serta algoritma-algoritma yang digunakan untuk mengerjakan masing-masing proses tersebut. Sistem ini terdiri dari 3 proses utama, yaitu :

1. Proses Pra-proses, adalah serangkaian proses untuk mempersiapkan citra tenun sebelum proses ekstraksi fitur dengan *Gabor Filter* dan klasifikasi *Fuzzy C-Means*. Proses pengolahan citra meliputi beberapa proses, yakni proses *cropping* secara manual, *grayscale*, dan normalisasi.
2. Proses ekstraksi fitur dengan *Gabor Filter* merupakan tahapan mengekstrak ciri/informasi dari objek di dalam citra yang ingin dikenali/dibedakan dengan objek lainnya. Hasil dari proses ekstraksi ini nantinya adalah vektor fitur.
3. Proses klasifikasi dengan *Fuzzy C-Means*, merupakan proses pengklasifikasian vektor fitur tenun. Vektor fitur akan diklasifikasikan menjadi 3 kelas, yaitu daerah Bali, NTT, dan Jawa Tengah.

4. Proses pengujian (*testing*), yaitu proses membandingkan fitur masukan baru dengan fitur yang ada pada referensi yang sebelumnya sudah dilatih kepada sistem.

Gambaran sistem klasifikasi tenun berdasarkan daerah dengan algoritma Fuzzy C-Means dapat dilihat pada Gambar 4.1



(a)

(b)

Gambar 4.1 Diagram Alir Proses (a) Pelatihan dan (b) Pengujian

4.1.3 Pra-Proses

Sebelum memperoleh fitur citra tenun, terlebih dahulu dilakukan pra-proses. Pra-proses dilakukan untuk menjamin kelancaran proses berikutnya. Proses-proses tersebut antara lain :

- a. Proses *cropping* citra, yaitu proses memotong citra tenun agar lebih fokus pada motif tenun yang teratur. Citra tenun yang telah didapat memiliki ukuran yang berbeda-beda, sehingga perlunya proses *cropping* untuk mendapatkan bagian terbaik dengan pola yang teratur dan penuh. Rata-rata dari semua citra, bagian yang memiliki pola yang penuh dan teratur adalah pada bagian tengah citra, sehingga citra di crop pada bagian tengah. Semua citra di crop dengan ukuran sama. Citra tenun di crop hingga berukuran 300x300 piksel. Contoh citra yang didapat dari internet, ditunjukkan dalam Lampiran G.1. contoh citra yang telah di crop pada bagian terbaik ditunjukkan dalam Lampiran G.2. pada proses *cropping* ini dilakukan secara manual.
- b. Proses *grayscale*, yaitu proses mengubah citra RGB (*Red, Green, Blue*) menjadi citra yang memiliki nilai keabuan. Proses ini membuat citra RGB yang memiliki tiga chanel warna yakni *chanel* warna merah, hijau dan biru menjadi citra *grayscale*.
- c. Proses normalisasi, yaitu proses yang dilakukan untuk menghilangkan efek *noise*. Persamaan yang digunakan untuk normalisasi dapat dilihat pada Persamaan (1). Pada proses ini menggunakan beberapa parameter yaitu $M_0 = 100$ dan $V_0 = 500$. Nilai didapat setelah melakukan beberapa percobaan. Sedangkan $I(x, y)$ adalah piksel citra yang telah di *grayscale*.



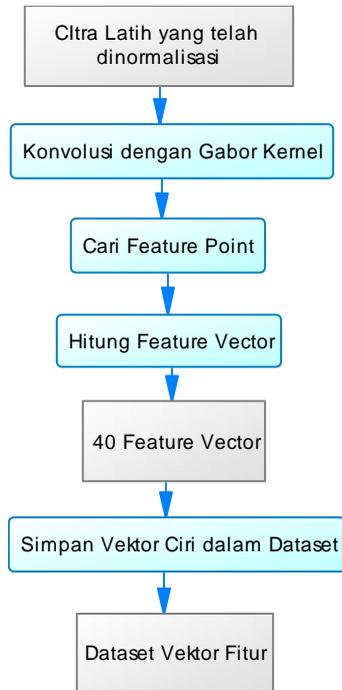
Gambar 4.2 Diagram Alir Pra-proses

4.1.4 Proses Ekstraksi Fitur Citra dengan *Gabor Filter*

Proses ekstraksi dilakukan untuk memperoleh ciri sebuah citra yang dapat membedakan dengan citra yang lain. Ekstraksi pada citra tenun ini diambil berdasarkan ciri teksturnya. Salah satu metode yang dapat digunakan untuk ekstraksi citra tenun berdasar tekstur adalah metode Gabor Filter.

Proses ekstraksi menggunakan *Gabor Filter* yang pertama yaitu proses penghitungan konvolusi piksel citra dengan *Gabor Kernel*. Piksel citra yang telah di grayscale, di konvolusi dengan *Gabor Kernel* piksel per piksel. Frekuensi yang digunakan ada lima yaitu 4, 5.6, 8, 11.3, dan 16. Sudut orientasinya ada delapan yaitu 0, 23, 45, 68, 90, 113, 135, dan 158. Konvolusi ini menghasilkan matriks baru. Kemudian akan dihitung *magnitude* dari hasil konvolusi *Gabor Filter* dan citra.

Magnitude merupakan nilai kompleks dari *Gabor Filter* yang merupakan penjumlahan *filter imaginer* dan *real*. Setelah cita di konvolusi dengan Gabor Kernel maka terbentuklah 40 *magnitude*.



Gambar 4.3 Diagram Alir Proses Ekstraksi Fitur Citra dengan Menggunakan Gabor Filter

Kemudian dilanjutkan proses mendapatkan *feature vector*. *Magnitude* pertama dicari nilai maksimum dengan prosedur menempatkan window W_0 dengan ukuran 4×4 . Pada setiap window berukuran 4×4 , dicari nilai maksimumnya. Nilai maksimum tersebut merupakan *feature points*. Kemudian koordinat dari tiap nilai maksimum pada *magnitude* pertama akan di proyeksikan ke *magnitude* 2 sampai 40. Sehingga nilai yang diambil dari *magnitude* 2 sampai 40 adalah nilai yang berada

pada koordinat yang sama dengan koordinat nilai maksimum pada *magnitude* pertama. Pada setiap *magnitude*, akan menghasilkan 5625 *feature points*, sehingga *feature points* yang didapat berukuran 5625×40 . Kemudian dihitung mean dari *feature points* sehingga didapatkan *feature vector* dengan ukuran 1×40 . Feature Vector yang dimiliki oleh 60 data citra tersebut disusun pada satu dataset sehingga membentuk matriks berukuran 60×40 .

4.1.5 Proses Klasifikasi dengan *Fuzzy C-Means*

Setelah proses ekstraksi menggunakan *Gabor Filter*, selanjutnya adalah proses klasifikasi. Dalam penelitian ini, proses klasifikasi dilakukan menggunakan algoritma *Fuzzy C-Means*. Algoritma Fuzzy C-Means pada awalnya merupakan algoritma klustering. Namun pada penelitian ini, algoritma FCM digunakan untuk klasifikasi dengan tidak mengubah parameter jumlah kluster. Akan dilakukan pengujian dengan mengubah parameter yang lain, sehingga hasil klasifikasi pada pelatihan bisa sesuai dengan kelasnya. Hasil ekstraksi ciri yang telah disimpan dalam dataset, akan diproses berdasarkan algoritma *Fuzzy C-Means*.

Berikut merupakan algoritma Fuzzy C-Means pada penelitian ini :

1. Parameter algoritma [11]:
 - a. Matriks X berukuran 60×40 , dengan 60 jumlah data yang akan di kluster, dan 40 jumlah *feature point* tiap citra pada data.
 - b. Jumlah kluster yang akan dibentuk adalah 3, sehingga $C = 3$
 - c. Pembobot $w = 2$
 - d. Maksimum iterasi adalah 100
 - e. Kriteria penghentian $\varepsilon = 0,00001$

2. Membentuk matriks partisi awal U (derajat keanggotaan dalam cluster) yang dibuat secara acak

$$U = \begin{bmatrix} \mu_{11}(x_1) & \mu_{12}(x_2) & \cdots & \mu_{160}(x_{60}) \\ \mu_{21}(x_1) & \mu_{22}(x_2) & \cdots & \mu_{260}(x_{60}) \\ \mu_{31}(x_1) & \mu_{32}(x_2) & \cdots & \mu_{360}(x_{60}) \end{bmatrix}$$

3. Menghitung pusat cluster V untuk setiap cluster

$$V_{ij} = \frac{\sum_{k=1}^{60} (\mu_{ik})^2 x_{kj}}{\sum_{k=1}^{60} (\mu_{ik})^2} \quad (5)$$

4. Memperbaiki derajat keanggotaan setiap data pada setiap cluster

$$\mu_{ik} = \left[\sum_{j=1}^3 \left(\frac{d_{ik}}{d_{jk}} \right)^2 \right]^{-1} \quad (6)$$

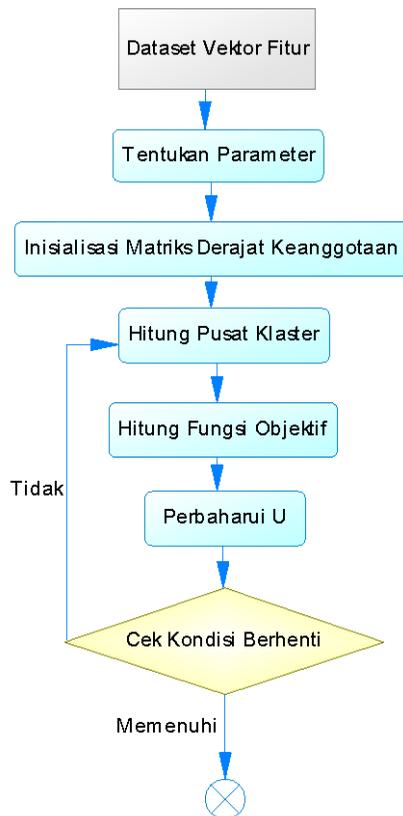
dengan

$$d_{ik} = d(x_k - v_i) = \left[\sum_{j=1}^{40} (x_{kj} - v_{ij}) \right]^{\frac{1}{2}}$$

5. Menentukan kriteria penghentian iterasi. Apabila $\Delta < 0,00001$ maka iterasi dihentikan. Perhitungan Δ didapat dengan menggunakan Persamaan (3).

Proses pertama algoritma Fuzzy C-Means adalah hitung pusat klaster (c). Setiap kelas memiliki pusat klaster yang membagi data menjadi beberapa kelas. Selanjutnya, hitung fungsi objektif yang akan menentukan kondisi berhenti dari iterasi Fuzzy C-Means. Kemudian dengan rumus perhitungan derajat keanggotaan, maka dilakukan update matriks derajat keanggotaan (U). Selanjutnya, dilakukan cek kondisi berhenti. Terdapat tiga keadaan iterasi berhenti. Pertama, jika nilai fungsi objektif baru dikurangi fungsi objektif awal lebih besar dari nilai akurasi, kedua jika baru dikurangi U lama lebih dari nilai akurasi, dan

ketiga jika jumlah iterasi kurang dari maksimal iterasi, maka iterasi dilakukan kembali. Iterasi dimulai lagi dengan meng-*update* pusat klaster. Dari beberapa proses tersebut, maka diperoleh beberapa data yaitu pusat kelas, nilai fungsi objektif, dan matriks derajat keanggotaan yang telah di *update*. Dari matriks derajat keanggotaan akan diketahui bahwa data tersebut termasuk kelas mana.



Gambar 4.4 Diagram Alir Proses Klasifikasi Fuzzy C-Means

4.1.6 Proses Testing dengan Menghitung Kemiripan Citra Menggunakan *Euclidean Distance*

Pada tahap ini, akan diambil satu citra secara acak, kemudian dihitung jaraknya dengan setiap kelas. Citra uji akan melalui pra proses untuk kemudian di ekstraksi menggunakan Gabor Filter. Pada proses ekstraksi, akan menghasilkan vektor fitur citra uji tersebut. Vektor fitur uji berukuran 1×40 . Vektor fitur ini, akan dihitung jaraknya dengan masing-masing pusat kelas hasil klasifikasi sebelumnya.

Dari hasil perhitungan jarak vektor fitur citra uji dengan pusat kelas, akan terlihat berapa jarak citra uji tersebut dengan tiap kelas. Citra uji termasuk dalam kelas yang menunjukkan hasil dengan jarak terkecil. Sehingga akan dapat diketahui bahwa citra uji tersebut berasal dari daerah mana.

4.2 Implementasi

Perancangan program yang telah dibangun selanjutnya diimplementasikan pada bahasa pemrograman dengan menggunakan *software* Matlab. Pembahasan dalam implementasi sistem meliputi implementasi antarmuka (*interface*) sistem, implementasi tahap akuisisi, pengolahan citra, implementasi tahap ekstraksi fitur dengan Gabor Filter, dan terakhir tahap klasifikasi dengan metode Fuzzy C-Means.

4.2.1 Antarmuka Penelitian

Pada tugas akhir ini, antarmuka sistem dibangun dengan menggunakan form dan kontrol yang terdapat pada Matlab. Adapun antarmuka yang diimplementasikan untuk menunjang penelitian tugas akhir ini adalah sebagai berikut :

1. Halaman Utama

Halaman utama merupakan antarmuka yang berisi menu-menu untuk menampilkan antarmuka-antarmuka lainnya dalam sistem. Hasil implementasi halaman utama dapat dilihat pada Gambar 4.5 berikut.



Gambar 4.5 Antarmuka halaman utama

Halaman utama dibuat dalam bentuk yang sederhana, terdiri dari 3 bagian utama, diantaranya :

1. Bagian *title bar*
2. Bagian *menu bar*
3. Bagian *main window*

Title bar merupakan bagian yang menunjukkan judul dari antarmuka yang sedang ditampilkan. Di bawah *title bar* terdapat *menu bar* yang berisi sederetan menu-menu yang digunakan oleh sistem. Adapun kegunaan menu-menu yang ditampilkan pada antarmuka utama sistem disajikan dalam Tabel 4.4 berikut ini :

Tabel 4.2 Kegunaan Menu Sistem

Menu	Kegunaan
Pelatihan dan Pengujian	Melakukan pelatihan agar sistem dapat mengenali motif tenun yang mirip untuk kemudian di klasifikasi sesuai motif. Dan juga melakukan pengujian sebua citra untuk di identifikasi citra tersebut masuk daerah mana
Keluar	Keluar dari sistem

Main window merupakan bagian antarmuka yang digunakan untuk menampilkan berbagai antarmuka lain di dalam sistem.

2. Antarmuka Pelatihan dan Pengujian

Antarmuka Pelatihan dan Pengujian berguna untuk melakukan pelatihan dan pengujian terhadap citra tenun. Hasil implementasi antarmuka pelatihan dan pengujian dapat dilihat pada Gambar 4.6 berikut.

**Gambar 4.6** Antarmuka pelatihan dan pengujian

Pada antarmuka ini, ada beberapa tombol, list, dll. dengan fungsi yang berbeda-beda yaitu:

- a. Tombol “Ambil Citra Latih” digunakan untuk mengambil citra latih tenun yang tersimpan dalam komputer
- b. Tombol “Ekstraksi Fitur” digunakan untuk melakukan pemrosesan terhadap citra yang akan dilatih. Pemrosesan tersebut meliputi pra-proses yang terdiri dari grayscaling dan normalisasi. Kemudian yang langsung dilanjutkan ke proses ekstraksi fitur dengan metode filterisasi gabor. Setelah diekstraksi, fitur yang telah didapat, disimpan dalam dataset.
- c. Tombol “Klasifikasikan” digunakan untuk melakukan pemrosesan terhadap dataset yang telah didapat pada proses sebelumnya, untuk di klasifikasikan dengan fuzzy c-means.
- d. Listbox tenun kelas 1 digunakan untuk menampilkan daftar tenun yang terklasifikasi pada kelas satu
- e. Edit text 1 digunakan untuk memasukkan string nama kelas yang cocok untuk kelas dua
- f. Tombol “simpan” 1 digunakan untuk mengambil dan menyimpan string yang dimasukkan pada edit text 1
- g. Listbox tenun kelas 2 digunakan untuk menampilkan daftar tenun yang terklasifikasi pada kelas dua
- h. Edit text 2 digunakan untuk memasukkan string nama kelas yang cocok untuk kelas dua
- i. Tombol “simpan” 2 digunakan untuk mengambil dan menyimpan string yang dimasukkan pada edit text 2
- j. Listbox tenun kelas 3 digunakan untuk menampilkan daftar tenun yang terklasifikasi pada kelas tiga
- k. Edit text 3 digunakan untuk memasukkan string nama kelas yang cocok untuk kelas tiga

- l. Tombol “simpan” 3 digunakan untuk mengambil dan menyimpan string yang dimasukkan pada edit text 3
- m. Axes 1 digunakan untuk menampilkan citra tenun yang akan diuji
- n. Tombol “Ambil Citra Uji” digunakan untuk mengambil citra uji tenun yang tersimpan dalam komputer
- o. Tombol “Pengujian” digunakan untuk pemrosesan ekstraksi fitur pada citra latih, yang kemudian vektor fitur tersebut di ukur jaraknya menggunakan euclidean distance
- p. Edit text 4 digunakan untuk menampilkan hasil citra uji dikelompokkan dalam daerah mana dengan mengambil nilai string yang telah disimpan tadi

4.2.2 Implementasi Proses Akuisisi Citra Tenun

Proses akuisisi citra tenun digunakan untuk mendapatkan citra tenun. Proses ini tidak dilakukan secara otomatis. Namun, membutuhkan interaksi pengguna untuk mencari dan mengambil citra yang dibutuhkan di dalam media penyimpanan.

Proses akuisisi citra tenun diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
folder_name = uigetdir(pwd, 'Select the
directory of images');
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran A.

4.2.3 Implementasi Proses Pra-pengolahan

Proses Pra-pengolahan diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
I=rgb2gray(I);
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran B.1. Fungsi ini menghasilkan sebuah image keabuan dengan nilai pixel berupa data *uint8*. *Image* ini berupa citra tenun yang telah dilakukan proses *grayscale*. Parameter dari fungsi merupakan *image* RGB dengan nilai pixel bertipe data *uint8* yang berisi data citra tenun masukan.

4.2.3 Implementasi Proses Normalisasi

Proses normalisasi diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
[I] = fg_norm(I);
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran B.2. Fungsi ini menghasilkan sebuah *image* keabuan dengan nilai pixel bertipe data *uint8*. *Image* ini merupakan data hasil proses normalisasi citra tenun. Parameter dari fungsi merupakan *image* keabuan dengan nilai pixel bertipe data *uint8* yang berisi data hasil proses pra-pengolahan.

4.2.4 Implementasi Proses Ekstraksi Ciri

A. Implementasi Gabor Kernel

Proses filterisasi diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
[V, Xj] = featurevector(image, matrik)
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran C.1. Fungsi ini menghasilkan array satu dimensi berupa *image* dengan format *grayscale* dan nilai pixel *double*. Array ini merupakan hasil konvolusi antara 40 Gabor kernel dan

citra tenun ternormalisasi. Pada Tugas Akhir ini menggunakan 40 Gabor kernel untuk proses filterisasi citra tenun, sehingga dalam fungsi ini memanggil fungsi konvolusi sebanyak 40 kali. Masing-masing pemanggilan fungsi akan menghasilkan sebuah citra tenun terfilter sesuai dengan frekuensi dan orientasi yang digunakan.

Sedangkan untuk proses pembuatan Gabor kernel diimplementasikan menjadi sebuah program dalam fungsi berikut ini:

```
function [gf , res]= gabor_filter (I,
sigx, sigy, omega, teta)
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran C.2.

B. Implementasi Proses Perhitungan Vektor Ciri

Proses penghitungan vektor ciri diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
function [isimaxM, saveg]=IsimaxM(image,
matrik, Mx, My)
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran C.3. Fungsi ini menghasilkan sebuah array bertipe double. Array ini merupakan hasil ekstraksi ciri citra tenun terfilter pada satu citra terfilter. Pada Tugas Akhir ini citra tenun terfilter memiliki 40 citra terfilter, sehingga fungsi ini dipanggil sebanyak 40 kali untuk mengekstraksi ciri setiap citra tenun. Kemudian dicari nilai maksimum yang merupakan *feature points*.

4.2.5 Implementasi Proses Penyimpanan Tenun

Proses penyimpanan vektor ciri diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
data(k, :) = [mean(mean(Xj))];
save('fiturvektor.mat', 'data');
```

Proses penyimpanan tenun diimplementasikan dalam penyimpanan identitas serta vektor ciri dalam basis data.

4.2.6 Implementasi Proses Klasifikasi

Proses klasifikasi vektor ciri diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
[center,U,obj_fcn] = fcm2(data, 3);
```

Fungsi ini menghasilkan nilai derajat keanggotaan tiap data dan vektor pusat tiap kelas. Kode program selengkapnya disajikan pada Lampiran D.

4.2.7 Implementasi Proses Perhitungan Jarak

Proses perhitungan jarak diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
[V, Xj] = featurevector(handles.Citra.RGB, 2);  
D1 = sqrt(sum((p1 - vtes) .^ 2));
```

Fungsi ini menghasilkan nilai ukuran yang dihasilkan dari vektor citra latih dengan pusat kelas. Kode program selengkapnya disajikan pada Lampiran E.

BAB V

HASIL DAN PEMBAHASAN PENGUJIAN SISTEM

Bab ini menjelaskan mengenai proses pengujian yang dilakukan terhadap sistem klasifikasi citra tenun dengan menggunakan metode fuzzy c-means. Hasil pengujian kemudian dibahas untuk mengetahui unjuk kerja sistem secara keseluruhan dalam mengidentifikasi tenun pengguna. Isi dari bab ini dimulai dengan mengenalkan lingkungan pengujian sistem yang digunakan. Selanjutnya dijelaskan mengenai hasil pengujian terhadap sistem yang telah diimplementasikan pada bab 4, yaitu pengujian hasil implementasi proses pra-pengolahan, proses ekstraksi ciri dan proses klasifikasi. Setelah itu, bab ini membahas tentang pengujian proses pengukuran jarak citra tenun uji. Bab ini diakhiri dengan pembahasan hasil pengujian yang telah dilakukan.

5.1 Lingkungan Pengujian Sistem

Lingkungan pengujian sistem meliputi beberapa perangkat keras dan perangkat lunak komputer. Detail spesifikasi lingkungan pengujian sistem dapat dilihat pada Tabel 5.1.

Tabel 5.1 Lingkungan Pengujian Sistem

Perangkat Keras	Prosesor : Intel® Core(TM)i3 CPU M 350 @2.27GHz 2.27 GHz
	Memory : 3 GB DDR3
Perangkat Lunak	Sistem Operasi : Windows 7 Home Basic 32-bit
	Tools : MATLAB

5.2 Proses Pengujian Tahap Akuisisi Citra

Tujuan dari pengujian tahap akuisisi adalah untuk mengetahui bahwa sistem telah mendapatkan citra tenun. Proses dilakukan dengan cara menginputkan file citra tenun ke dalam program yang telah dibuat. Hasil akuisisi citra Tenun Bali, Tenun NTT, dan Tenun Jawa Tengah ditunjukkan dalam Gambar 5.1



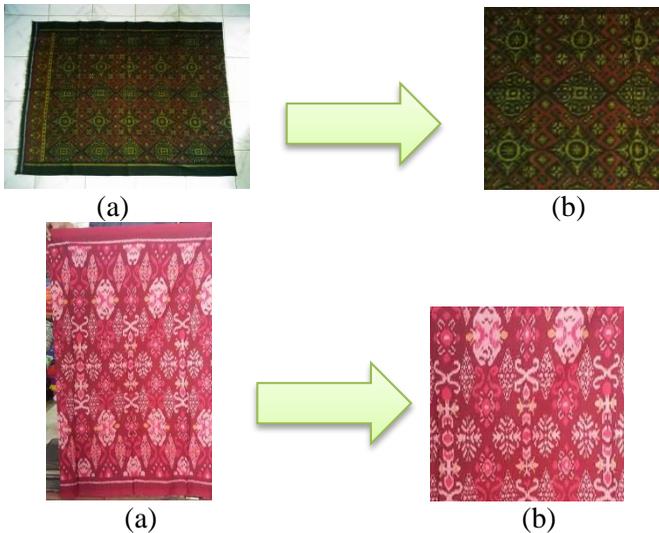
Gambar 5.1 Hasil akuisisi citra tenun (a) Tenun Bali (b) Tenun NTT (c) Tenun Jawa Tengah

Pada tahap ini, telah mendapatkan citra tenun sebanyak 25 citra tenun Bali, 25 citra tenun NTT, dan 25 citra tenun Jawa Tengah. Dari 75 citra, nantinya akan dibagi menjadi 60 citra latih dan 15 citra uji. Untuk detail data yang digunakan pada penelitian ini dapat dilihat pada lampiran F.

5.3 Proses Pengujian Tahap Pra-Proses

Pengujian ini dilakukan terhadap proses-proses pada tahap pengolahan citra. Pengujian bertujuan untuk mengetahui bahwa proses-proses pada tahap Pengolahan citra sudah benar, sehingga data yang disajikan dari tahap ini dipastikan dapat menjadi input pada tahap berikutnya. Pada tahap ini terdapat 4 proses yaitu:

1. Pada proses ini dilakukan *cropping* secara manual. Citra yang didapat memiliki ukuran citra yang berbeda beda. Sehingga, *cropping* citra dilakukan dengan mengambil bagian terbaik pada citra. Dari hasil *cropping*, didapatkan citra berukuran 300x300 piksel.



Gambar 5.2 (a) citra hasil akuisisi , (b) citra hasil *cropping*

2. Pengujian proses *grayscale*. Proses ini bertujuan untuk mengetahui bahwa sistem telah mengubah citra peluru hasil *cropping* yang masih dalam bentuk citra berwarna menjadi citra *grayscale*. Setelah diperoleh beberapa citra dengan ukuran yang sama, citra diubah menjadi citra *grayscale* seperti pada Gambar 5.3

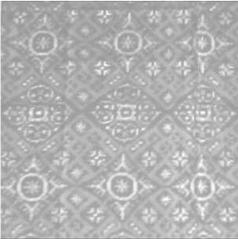
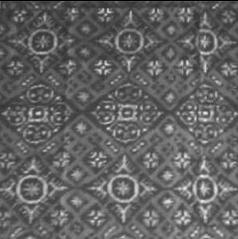


Gambar 5.3 Citra hasil *grayscale*

3. Pengujian proses normalisasi. Proses ini bertujuan untuk mengetahui bahwa sistem telah mengubah citra *grayscale* menjadi citra yang telah dinormalisasi. Dilakukan percobaan menggunakan parameter yang berbeda. Dilakukan percobaan dengan $M_0 = 180$ dan $V_0 = 180$ [13]. Kemudian dengan

$M_0 = 100$ dan $V_0 = 500$ [12]. Hasil percobaan dapat dilihat pada Tabel 5.2.

Tabel 5.2 Hasil Percobaan normalisasi dengan parameter berbeda

M_0	V_0	Waktu Komputasi	Hasil
180	180	0,0300 detik	
100	500	0,0320 detik	

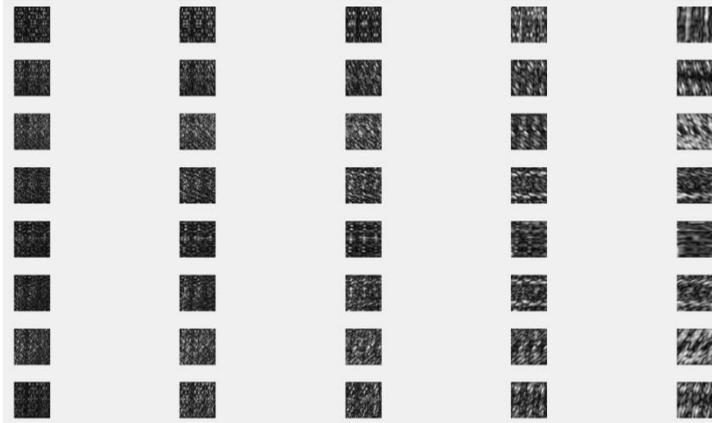
Pada normalisasi dengan $M_0 = 180$ dan $V_0 = 180$, citra telah mengalami keseragaman cahaya, namun motif tenun yang terdapat pada citra menjadi kurang terlihat jelas. Sedangkan pada normalisasi dengan $M_0 = 100$ dan $V_0 = 500$, citra juga mengalami keseragaman cahaya, dan motif tenun pada citra terlihat jelas. Diantara dua percobaan tersebut terdapat perbedaan waktu komputasi sebesar 0,0020 detik.

5.4 Pengujian Tahap Ekstraksi Ciri

5.4.1 Pengujian Hasil Implementasi Proses Filterisasi

Program hasil implementasi proses filterisasi mendapatkan input berupa citra tenun yang telah dinormalisasi. Citra tenun yang telah di normalisasi kemudian dikonvolusi dengan Gabor kernel. Output yang dihasilkan pada proses ini berupa 40 buah

citra tenun. Gambar 5.4 menunjukkan hasil pengujian program implementasi proses filterisasi citra tenun.



Gambar 5.4 Hasil Filterisasi Citra Tenun

5.4.2 Pengujian Hasil Implementasi Proses Perhitungan Vektor Ciri

Input untuk proses perhitungan vektor ciri ini berupa 40 buah citra tenun yang telah difilter pada proses sebelumnya. Kemudian dicari nilai maksimum dengan menempatkan window W_0 dengan ukuran 4×4 . Nilai maksimum tersebut merupakan *feature points*. Output yang dihasilkan berupa *feature points* berukuran 5625×40 dengan jenis data berupa *double*. Kemudian *feature points* ini di hitung meannya sehingga dari satu citra akhirnya menghasilkan 1×40 fitur vektor, sehingga dalam dataset terdapat 60×40 kumpulan fitur vektor. Kemudian fitur vektor ini disimpan dalam dataset untuk kemudian digunakan pada proses selanjutnya.

5.5 Pengujian Tahap Klasifikasi dengan Fuzzy C-Means

Pada proses ini menggunakan 60 data citra latihan dengan rincian 20 citra tenun Bali, 20 citra tenun NTT, dan 20 citra tenun Jawa tengah. Data yang diklasifikasi berasal dari hasil

perhitungan vektor ciri. Sehingga matriks data berukuran 60×40 dengan jumlah data 60 dan jumlah variabel 40.

Sebelum melakukan proses Fuzzy C-Means, terlebih dahulu menentukan parameter Fuzzy. Parameter yang dibutuhkan ditunjukkan pada Tabel 5.3.

Tabel 5.3 Tabel Parameter *Fuzzy C-means*

Jumlah klaster	K	3
Pembobot	w	2
Maksimum iterasi	maksiter	100
Kriteria Penghentian	ε	0,00001
Fungsi objektif awal	J_0	0
Iterasi awal	t	1

Kemudian dibentuk matriks derajat keanggotaan awal dengan matriks random. Matriks awal U ini berukuran 60×3 . Selanjutnya hitung pusat kelas dengan persamaan (3). Setelah itu, perbaiki derajat keanggotaan tiap citra dengan persamaan (4). Kemudian hitung kriteria perhentian iterasi dengan persamaan (5). Jika belum memenuhi syarat atau belum kurang dari ε , maka proses akan dilanjutkan kembali dari perbaikan pusat kelas. Hasil klasifikasi pada proses *training* ditunjukkan pada Tabel 5.4.

Tabel 5.4 Tabel Hasil Klasifikasi

	Bali	NTT	Jawa Tengah
Bali	18	0	2
NTT	0	16	4
Jawa Tengah	0	0	20

Dari Tabel 5.4 menunjukkan bahwa citra tenun Bali yang benar diklasifikasikan pada kelas Bali terdapat 18 citra. Citra tenun NTT yang benar diklasifikasikan pada kelas NTT terdapat 16 citra. Dan citra tenun Jawa Tengah yang benar diklasifikasikan pada kelas Jawa Tengah yang benar diklasifikasikan pada kelas Jawa Tengah terdapat 20 citra. Rincian derajat keanggotaan tiap citra akan ditunjukkan pada lampiran G.1.

Pada proses klasifikasi ini, juga menghasilkan output data berupa vektor pusat. Vektor pusat merupakan vektor yang menjadi pusat kelas tersebut. Vektor pusat ini nantinya akan digunakan untuk proses selanjutnya. Vektor pusat tiap kelas ditunjukkan pada lampiran G.2.

5.6 Perbandingan Hasil Klasifikasi dengan *Magnitude* Gabor Filter

Ekstraksi Gabor Filter dilakukan dengan proses konvolusi antara piksel citra dengan Gabor Kernel. Jumlah *Magnitude* bergantung pada jumlah frekuensi dan orientasi. Telah dilakukan proses klasifikasi dengan beberapa jumlah *Magnitude* pada Gabor yang berbeda. Pada percobaan ini dilakukan dengan 12 *Magnitude*[12], 20 *Magnitude*[13], dan 40 *Magnitude*[8]. Akurasi hasil klasifikasi dapat dilihat pada tabel 5.5.

Tabel 5.5 Hasil Klasifikasi dengan *Magnitude* Filter yang Berbeda

Jumlah <i>Magnitude</i>	Akurasi Hasil Klasifikasi	Waktu Komputasi
12 <i>Magnitude</i>	60%	3,0150 detik
20 <i>Magnitude</i>	58,33%	8,2544 detik
40 <i>Magnitude</i>	90%	27,9660 detik

5.7 Pengujian Tahap Perhitungan Kemiripan Citra

Pada proses ini menggunakan data yang telah di *training* atau citra *database*, dan 15 data diluar data *training* atau citra data uji. Data uji akan melewati pre-proses dan proses ekstraksi. Sehingga menghasilkan vektor fitur berukuran 1x40. Dari vektor fitur data uji tersebut, akan dihitung kemiripan citra nya dengan vektor pusat tiap kelas. menghitung kemiripan citra menggunakan *Euclidean Distance*.

Dilakukan pengujian citra dengan pengulangan sebanyak sepuluh kali. Data yang digunakan, diambil secara acak dari data *database*, kemudian diambil lagi secara acak dari data uji. Diambil sebanyak sepuluh citra dan dilakukan pengulangan

pengujian sebanyak sepuluh kali. Kemudian dilanjutkan dengan perhitungan akurasi tiap pengujian.

Cara menghitung tingkat akurasi sistem yaitu dengan menggunakan persamaan:

$$Akurasi = \frac{B + N + J}{jumlah\ citra\ uji} \times 100$$

dengan :

B = Jumlah citra Bali benar

N = Jumlah citra NTT benar

J = Jumlah citra Jawa Tengah benar

Sehingga menghasilkan akurasi yang berbeda pada citra database dan citra uji. Hasil dari pengujian dapat dilihat pada Tabel 5.6 untuk pengujian pada citra database, dan Tabel 5.7 untuk citra pada data uji. Kemudian untuk hasil lebih rincinya dapat dilihat pada lampiran G.3 untuk pengujian pada citra database, dan lampiran G.4 untuk pengujian pada citra uji. Dari hasil pengujian, dihitung akurasi dari masing-masing pengujian dan diambil rata-rata dari akurasi tersebut. Hasil rata-rata akurasi pengujian pada citra database dan pada citra uji dapat dilihat pada Tabel 5.8.

Tabel 5.6 Tabel Pengujian dengan Pengulangan Sepuluh Kali pada Citra Database

Uji ke-	Benar	Salah
1	10	0
2	10	0
3	10	0
4	10	0
5	10	0
6	10	0
7	10	0
8	10	0
9	10	0
10	10	0

Tabel 5.7 Tabel Pengujian dengan Pengulangan Sepuluh Kali pada Citra Uji

Uji ke-	Benar	Salah
1	9	1
2	9	1
3	10	0
4	9	1
5	10	0
6	10	0
7	9	1
8	9	1
9	10	0
10	9	1

Tabel 5.8 Tabel Rata-rata Akurasi Pengujian dengan Pengulangan

Citra yang diuji	Rata-rata Akurasi
Citra Database	100%
Citra Uji	94%

5.8 Pengujian dengan Perubahan Parameter Fuzzy C-Means

Pada proses klasifikasi menggunakan Fuzzy C-Means, terdapat beberapa parameter yang dibutuhkan. Salah satu parameter yang akan dilakukan uji coba adalah pembobot (w) dan kriteria penghentian (ϵ).

Pada saat pembobot berukuran 2, hasil klasifikasi lebih akurat. Klasifikasi dapat mengelompokkan citra sesuai dengan kelas yang diinginkan. Sehingga, citra diklasifikasi menjadi 3 kelas. Sedangkan pada saat pembobot berukuran lebih dari 2, hasil klasifikasi tidak sesuai dengan data seharusnya. Hasil dari pembobot lebih dari 2, mengelompokkan data pada 2 kelas saja.

Kemudian dilakukan percobaan dengan mengubah kriteria penghentian (ϵ). Pada hasil percobaan, menunjukkan

semakin kecil kriteria penghentian, semakin besar akurasi yang dihasilkan. Hasil percobaan dapat dilihat pada Tabel 5.9.

Tabel 5.9 Hasil Percobaan Klasifikasi dengan Kriteria Penghentian yang Berbeda

ε	Banyak Iterasi	Akurasi Hasil
10^{-1}	4	86,67 %
10^{-2}	22	88,33 %
10^{-3}	25	88,33 %
10^{-4}	35	88,33 %
10^{-5}	51	90 %

5.9 Pembahasan Penyebab Citra tidak Teridentifikasi Benar Sesuai Kelasnya

Penyebab utama citra tidak teridentifikasi benar sesuai kelasnya antara lain :

1. Citra yang diakuisisi memiliki motif yang sangat mirip dengan citra lain yang berasal dari daerah lain. Terdapat tenun Bali yang memiliki motif hampir mirip dengan motif Jawa Tengah. Sehingga hasil ekstraksi cirinya juga menjadi mirip.
2. Kualitas citra yang diakuisisi kurang begitu baik. Motif atau tekstur pada citra kurang terlihat, sehingga fitur ciri menjadi mirip dan dianggap sama.
3. Terdapat data pada citra latih yang memiliki permasalahan sama dengan poin nomor 1 dan 2 sehingga pada training klasifikasi menjadi berantakan dan tidak sesuai dengan kelasnya.

BAB VI PENUTUP

Bab ini berisi tentang beberapa kesimpulan yang dihasilkan berdasarkan penelitian yang telah dilaksanakan. Di samping itu, pada bab ini juga dimasukkan beberapa saran yang dapat digunakan jika penelitian ini ingin dikembangkan.

6.1 Kesimpulan

Berdasarkan analisis terhadap hasil pengujian sistem klasifikasi tenun dengan menggunakan metode *Gabor Filter* dan *Fuzzy C-Means*, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Presentase tingkat keberhasilan sistem klasifikasi tenun pada penelitian ini dapat mencapai 100% untuk citra pada database.
2. Presentase tingkat keberhasilan sistem klasifikasi tenun pada penelitian ini dapat mencapai 94% untuk citra diluar database atau citra uji.
3. Ekstraksi *Gabor Filter* menggunakan 40 magnitudo lebih memberikan hasil yang akurat daripada 12 magnitudo dan 20 magnitudo. Namun waktu komputasi yang dibutuhkan lebih lama.
4. Semakin kecil kriteria penghentian, akurasi hasil klasifikasi akan semakin tinggi.

6.2 Saran

Dengan melihat hasil yang dicapai pada penelitian ini, ada beberapa hal yang penulis sarankan untuk pengembangan selanjutnya yaitu:

1. Pemilihan citra harus memiliki kualitas yang baik dengan menunjukkan motif yang jelas pada citra sehingga tekstur citra lebih terlihat
2. Pada penelitian ini hanya menggunakan tiga jenis motif yang terkenal karena keterbatasan data, sehingga dapat dikembangkan dengan menggunakan lebih dari tiga motif.

3. Metode yang digunakan dalam penelitian ini dapat dikembangkan untuk aplikasi sistem klasifikasi citra yang berbasis tekstur lainnya seperti batik.
4. Ekstraksi *Gabor Filter* menggunakan 40 *magnitude* lebih memberikan hasil yang akurat namun waktu komputasi lebih lama. Maka bisa dimodifikasi dengan metode lain untuk menangani lamanya waktu komputasi.
5. Penelitian dapat dikembangkan dengan menambahkan kebebasan menggunakan jenis *file* selain JGP dan JPEG. Juga menambahkan pada sistem, algoritma *cropping* secara otomatis.

DAFTAR PUSTAKA

- [1] Hartanti, G. 2011. *Tenun dan Penerapannya pada Desain Interior Sebagai Warisan Budaya yang memiliki Nilai Jual yang Tinggi*. Jurusan Desain Interior, Fakultas Komunikasi Multimedia, Bina Nusantara University.
- [2] Nguyen, T. Khosravi, A. Creighton, D. Nahavandi, S. *Medical Data Classification Using Interval Type-2 Fuzzy Logic System and Wavelets*. Centre for Intelligent Systems Research (CISR), Deakin University, Geelong Waurm Ponds Campus, Victoria 3216, Australia.
- [3] Wikipedia. Januari 2016. Kain Tenun NTT. <URL: https://id.wikipedia.org/wiki/Kain_tenun_NTT>. (diakses 20 Maret 2016).
- [4] Kartiwa, S. 1987. *Ragam Hias Indonesia: Tenun Ikat*. Djambatan : Jakarta.
- [5] Gonzales, C. R. Woods, R. E. 2002. *Digital Image Processing*. Prentice-Hall Inc, New Jersey, USA.
- [6] Susilo, A. 2006. *Web Image Retrieval Untuk Identifikasi Bunga Dengan Pengelompokan Content Menggunakan Ciri Warna dan Bentuk*. Jurusan Teknologi Informasi Politeknik Elektronika Negeri Surabaya.
- [7] Carmona, R. Hwang, W. L. Torresani, B. 1998. *Practical Time Frequency Analysis: Gabor and Wavelet Transforms with an Implementation in S (Wavelet Analysis and Its Application)*. Academic Press, Cambirdge, USA.
- [8] Kepenekci, B. 2001. *Face recognition using gabor wavelet transform*. Middle East Technical University, Department of Electrical and Electronics Engineering, Ankara, Turkey.
- [9] Legind, H. 2002. *Possibility Theory and Fuzzy Pattern Matching*. University Esbjerg, Denmark.
- [10] Bandemer, H. Gottwald, S. 1996. *Fuzzy Sets, Fuzzy Logic, Fuzzy Methods with Applications*. West Sussex, England.
- [11] Du, G. Miao, F. Tian, S. Liu, Y. 2009. *A Modified Fuzzy C-means Algorithm in Remote Sensing Image Segmentation*. International Conference on Environmental Science and

Information Application Technology, IEEE (No. 2008DTKF102).

- [12] Fahmi, M. K. Ratna, D. ***Deteksi Cacat Pada Selongsong Peluru Berbasis Citra Menggunakan Gabor Filter***. Jurusan Matematika, Institut Teknologi Sepuluh Nopember.
- [13] Putra, I K. G. D. Bhuana, W. Erdiawan. ***Pembentukan Kode Telapak Tangan (Palm Code) Berbasis Metode Gabor 2D***. Jurusan Teknik Elektro, Fakultas Teknik, Universitas Udayana, Bandung.

LAMPIRAN A

A. Kode Prosedur Proses Akuisisi Citra Tenun

```

folder_name = uigetdir(pwd, 'Select the
directory of images');
if ( folder_name ~= 0 )
    handles.folder_name = folder_name;
    guidata(hObject, handles);
else
    return;
end

if (~isfield(handles, 'folder_name'))
    errordlg('Silahkan pilih dataset citra
terlebih dahulu');
    return;
end

pngImagesDir = fullfile(handles.folder_name,
'*.*png');
jpgImagesDir = fullfile(handles.folder_name,
'*.*jpg');
bmpImagesDir = fullfile(handles.folder_name,
'*.*bmp');

num_of_png_images = numel( dir(pngImagesDir) );
num_of_jpg_images = numel( dir(jpgImagesDir) );
num_of_bmp_images = numel( dir(bmpImagesDir) );
totalImages = num_of_png_images +
num_of_jpg_images + num_of_bmp_images;

jpg_files = dir(jpgImagesDir);
png_files = dir(pngImagesDir);
bmp_files = dir(bmpImagesDir);

if ( ~isempty( jpg_files ) ||
~isempty( png_files ) || ~isempty( bmp_files ) )
    jpg_counter = 0;

```

LAMPIRAN A (LANJUTAN)

```

png_counter = 0;
bmp_counter = 0;
for k = 1:totalImages

    if ( (num_of_jpg_images - jpg_counter) >
0)
        imgInfoJPG =
imfinfo( fullfile( handles.folder_name,
jpg_files(jpg_counter+1).name ) );
        if
( strcmp( lower(imgInfoJPG.Format), 'jpg') == 1 )
            sprintf('%s \n',
jpg_files(jpg_counter+1).name)

                I =
imread( fullfile( handles.folder_name,
jpg_files(jpg_counter+1).name ) );
                [pathstr, name, ext] =
fileparts( fullfile( handles.folder_name,
jpg_files(jpg_counter+1).name ) );
                end

                jpg_counter = jpg_counter + 1;

elseif ( (num_of_png_images - png_counter) > 0)
        imgInfoPNG =
imfinfo( fullfile( handles.folder_name,
png_files(png_counter+1).name ) );
        if
( strcmp( lower(imgInfoPNG.Format), 'png') == 1 )
            sprintf('%s \n',
png_files(png_counter+1).name)
                I =
imread( fullfile( handles.folder_name,
png_files(png_counter+1).name ) );

```

LAMPIRAN A (LANJUTAN)

```

                                [pathstr, name, ext] =
fileparts( fullfile( handles.folder_name,
png_files(png_counter+1).name ) );
                                end

                                png_counter = png_counter + 1;

                                elseif ( (num_of_bmp_images -
bmp_counter) > 0)
                                imgInfoBMP =
iminfo( fullfile( handles.folder_name,
bmp_files(bmp_counter+1).name ) );
                                if
( strcmp( lower(imgInfoBMP.Format), 'bmp') == 1 )
                                sprintf('%s \n',
bmp_files(bmp_counter+1).name)
                                I =
imread( fullfile( handles.folder_name,
bmp_files(bmp_counter+1).name ) );
                                [pathstr, name, ext] =
fileparts( fullfile( handles.folder_name,
bmp_files(bmp_counter+1).name ) );
                                end

                                bmp_counter = bmp_counter + 1;

                                end

```

LAMPIRAN A (LANJUTAN)

“Halaman ini sengaja dikosongkan”

LAMPIRAN B

B. 1 Kode Fungsi Pra-Pemrosesan

```
if ~isgray(I)
    I=rgb2gray(I);
end
```

B. 2 Kode Fungsi Proses Normalisasi

```
function [nor] = fg_norm(I)
jml = sum(sum(I));
ukuran = (size(I,1)*size(I,2));
rata=jml/ukuran;
var=0;

for x=1:size(I,1)
    for y=1:size(I,2)
        var=var+((double(I(x,y))-
double(rata))^2)/(ukuran-1));
    end
end

nor=uint8(zeros(size(I)));
for k=1:size(I,1)
    for j=1:size(I,2)
        if I(k,j)<rata
            nor(k,j)=uint8(100-
double(sqrt((500*(double(I(k,j))-
double(rata))^2)/var)));
        end
        if I(k,j)>rata
            nor(k,j)=uint8(100+double(sqrt((500*(double(I(k,
j))-double(rata))^2)/var)));
        end
    end
end
```

LAMPIRAN B (LANJUTAN)

end

end

LAMPIRAN C

C.1 Kode Fungsi Proses Filterisasi

```

function [V, Xj] = featurevector(image, matrik)
I = image ;
if ~isgray(I)
    I=rgb2gray(I);
end

[I] = fg_norm(I);

theta=[0 22.5 45 67.5 90 112.5 135 157.5];
lambda=[4 5.6 8 11.3 16];
sigma=lambda;

[filter, res]=gabor_filter(I, sigma(1), sigma(1),
(2*pi/lambda(1)), theta(1));

[row, col]=size(res);
maxM=titik(res, row, col, matrik);
[Mx, My]=koord(maxM, res, row, col, matrik);

[isimaxM, saveg]=IsimaxM(I, matrik, Mx, My);
for i=1:length(Mx)
    V(i, :)= [Mx(i) My(i) isimaxM(i, :)];
    Xj(i, :)= [isimaxM(i, :)];
end
end

```

C.2 Kode Fungsi Proses Pembuatan Filter Gabor

```

function [gf , res]= gabor_filter (I, sigx, sigy,
omega, teta)

teta = (pi/8)*teta;
k=1;

```

LAMPIRAN C (LANJUTAN)

```

l=1;

for x=-3*sigx:3*sigx
    for y=-3*sigy:3*sigy
        gf(k,l) = (1/(2*pi*sigx*sigy))*exp(-
0.5*((x/sigx).^2+(y/sigy).^2)+(2*j*omega*(x*cos(
teta)+y*sin(teta))));
        l=l+1;
    end
    l=1;
    k=k+1;
end

re_gf=real(gf);
im_gf=imag(gf);

I1=conv2(double(I), re_gf, 'same');
I2=conv2(double(I), im_gf, 'same');
res=(sqrt((I1.*I1)+(I2.*I2)));
end

```

C.3 Kode Fungsi Proses Penghitungan Vektor Ciri

```

function [isimaxM, saveg]=IsimaxM(image, matrik,
Mx, My)

I=image;
if ~isgray(I)
    I=rgb2gray(I);
end

[I] = fg_norm(I);

theta=[0 22.5 45 67.5 90 112.5 135 157.5];
lambda=[4 5.6 8 11.3 16];

```

LAMPIRAN C (LANJUTAN)

```

sigma=lambda;

isimaxM=[];
ii=1;
z=1;
for j=1:5
    for k=1:8
        [filt, res]=gabor_filter(I, sigma(j),
sigma(j), (2*pi/lambda(j)), theta(k));
saveg(:, :, z)=res;
z=z+1;

    for i=1:length(Mx)
        isimaxM(i, ii)=res(My(i), Mx(i));
    end

    ii=ii+1;
end
end

function maxM=titik(M, row, col, matrik)

a=1;
b=matrik;
aa=1;
bb=matrik;
for r=1:row/matrik
    for c=1:col/matrik
        i=a:b;
        j=aa:bb;
        maxM(r, c)=max(max(M(i, j)));
        aa=aa+matrik;
        bb=bb+matrik;
        if (aa==row+1)
            aa=1;

```

LAMPIRAN C (LANJUTAN)

```

        bb=matrik;
    end
end
    a=a+matrik;
    b=b+matrik;
    aa=1;
    bb=matrik;
end
end
function [Mx,My]=koord(maxM, M, row, col,
matrik)

    Mx=[];
    My=[];
    a=1;
b=matrik;
aa=1;
bb=matrik;
brs=1;
kol=1;
for r=1:row/matrik
    for c=1:col/matrik
        i=a:b;
        j=aa:bb;
        MMM=M(i,j);
        for ii=1:matrik
            for jj=1:matrik
                if (MMM(ii,jj)==maxM(r,c))
                    Mx=[Mx jj+(matrik*(c-1))];
                    My=[My ii+(matrik*(r-1))];
                    kol=kol+1;
                    aa=aa+matrik;
                    bb=bb+matrik;
                    if (aa==row+1)
                        aa=1;
                        bb=matrik;
                    end
                end
            end
        end
    end
end

```

LAMPIRAN C (LANJUTAN)

```
                                end
                            end
                        end
                    end
                a=a+matrik;
                b=b+matrik;
                aa=1;
                bb=matrik;
                brs=brs+1;
                kol=1;
            end
        end
```

LAMPIRAN C (LANJUTAN)

“Halaman ini sengaja dikosongkan”

LAMPIRAN D

D. Kode Prosedur Proses Klasifikasi

```

data=importdata('fiturvektor.mat', 'data');
    [center,U,obj_fcn] = fcm(data, 3);
    puskel1=center(1,:);
    puskel2=center(2,:);
    puskel3=center(3,:);

    save('pusat1.mat','puskel1');
    save('pusat2.mat','puskel2');
    save('pusat3.mat','puskel3');
    k=1;l=1;m=1;
    for(i=1:size(U,2))
        if(U(1,i)>U(2,i) && U(1,i)>U(3,i))
            indeks(i)=1;

prev_list=cellstr(get(handles.list1,'string'));
    vars=sprintf('tenun%d',i);
    new_list=[prev_list;{vars}];
    set(handles.list1,'string',new_list)
    elseif(U(2,i)>U(1,i) && U(2,i)>U(3,i))
        indeks(i)=2;

prev_list=cellstr(get(handles.list2,'string'));
    vars=sprintf('tenun%d',i);
    new_list=[prev_list;{vars}];
    set(handles.list2,'string',new_list)
    else
        indeks(i)=3;

prev_list=cellstr(get(handles.list3,'string'));
    vars=sprintf('tenun%d',i);
    new_list=[prev_list;{vars}];
    set(handles.list3,'string',new_list)
    end
end
save('indeks.mat','indeks');

```

LAMPIRAN D (LANJUTAN)

“Halaman ini sengaja dikosongkan”

LAMPIRAN E

E. Kode Prosedur Proses Perhitungan Kemiripan

```
[V, Xj] = featurevector (handles.Citra.RGB, 4);
vtes=mean(Xj)

vtes

p1=importdata('pusat1.mat','puskel1');
p2=importdata('pusat2.mat','puskel2');
p3=importdata('pusat3.mat','puskel3');

D1 = sqrt(sum((p1 - vtes) .^ 2));
D2 = sqrt(sum((p2 - vtes) .^ 2));
D3 = sqrt(sum((p3 - vtes) .^ 2)); global nama1,
global nama2, global nama3
Im=rgb2gray(handles.Citra.RGB);
[V, Xj] = featurevector (Im, 4);
vtes=mean(Xj);

vtes

p1=importdata('pusat1.mat','puskel1');
p2=importdata('pusat2.mat','puskel2');
p3=importdata('pusat3.mat','puskel3');

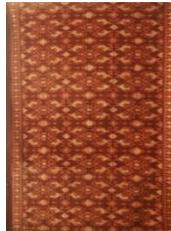
D1 = sqrt(sum(power(minus(p1,vtes),2)));
D2 = sqrt(sum(power(minus(p2,vtes),2)));
D3 = sqrt(sum(power(minus(p3,vtes),2)));

D1
D2
D3

if D1<D2 && D1<D3
    set(handles.edit1,'String',nama1);
```

LAMPIRAN E (LANJUTAN)

```
elseif D2<D1 && D2<D3
    set(handles.edit1,'String',nama2);
else
    set(handles.edit1,'String',nama3);
end
```

LAMPIRAN F**F.1 Data Citra Tenun dari Bali****Tenun 1****Tenun 2****Tenun 3****Tenun 4****Tenun 5****Tenun 6****Tenun 7****Tenun 8****Tenun 9****Tenun 10****Tenun 11****Tenun 12**

LAMPIRAN F (LANJUTAN)



Tenun 13



Tenun 14



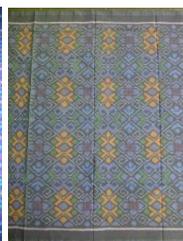
Tenun 15



Tenun 16



Tenun 17



Tenun 18



Tenun 19



Tenun 20

LAMPIRAN F (LANJUTAN)

F.2 Data Citra Tenun dari Jawa Tengah



Tenun 21



Tenun 22



Tenun 23



Tenun 24



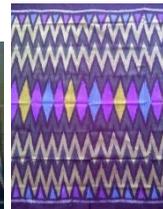
Tenun 25



Tenun 26



Tenun 27



Tenun 28



Tenun 29



Tenun 30



Tenun 31



Tenun 32



Tenun 33



Tenun 34



Tenun 35

LAMPIRAN F (LANJUTAN)



Tenun 36



Tenun 37



Tenun 38



Tenun 39



Tenun 40

LAMPIRAN F (LANJUTAN)**F.3 Data Citra Tenun dari NTT****Tenun 41****Tenun 42****Tenun 43****Tenun 44****Tenun 45****Tenun 46****Tenun 47****Tenun 48****Tenun 49****Tenun 50****Tenun 52****Tenun 53**

LAMPIRAN F (LANJUTAN)



Tenun 54



Tenun 55



Tenun 56



Tenun 57



Tenun 58



Tenun 59

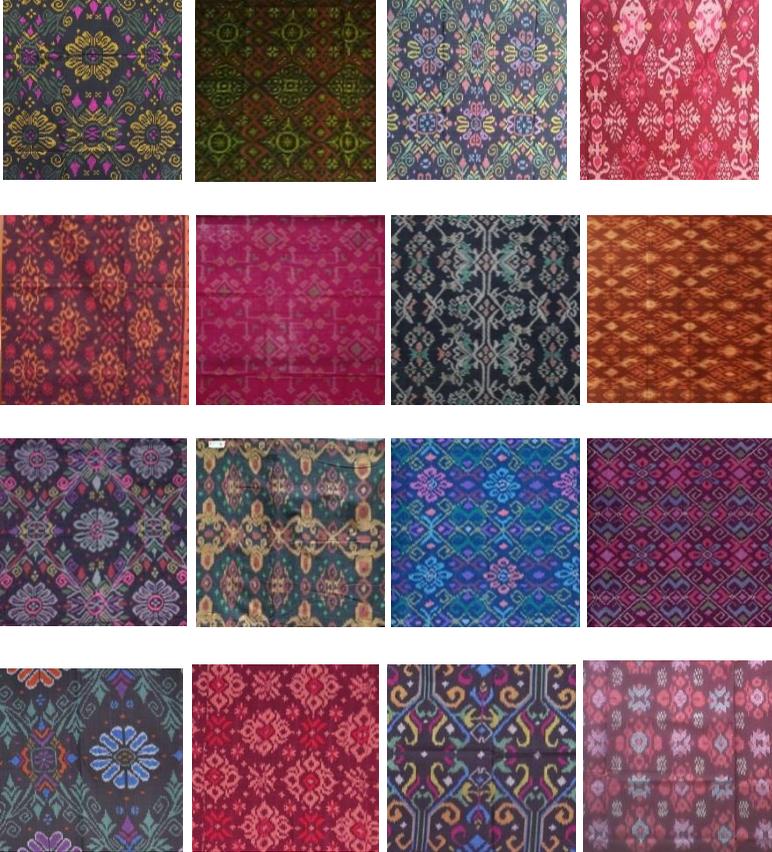


Tenun 60

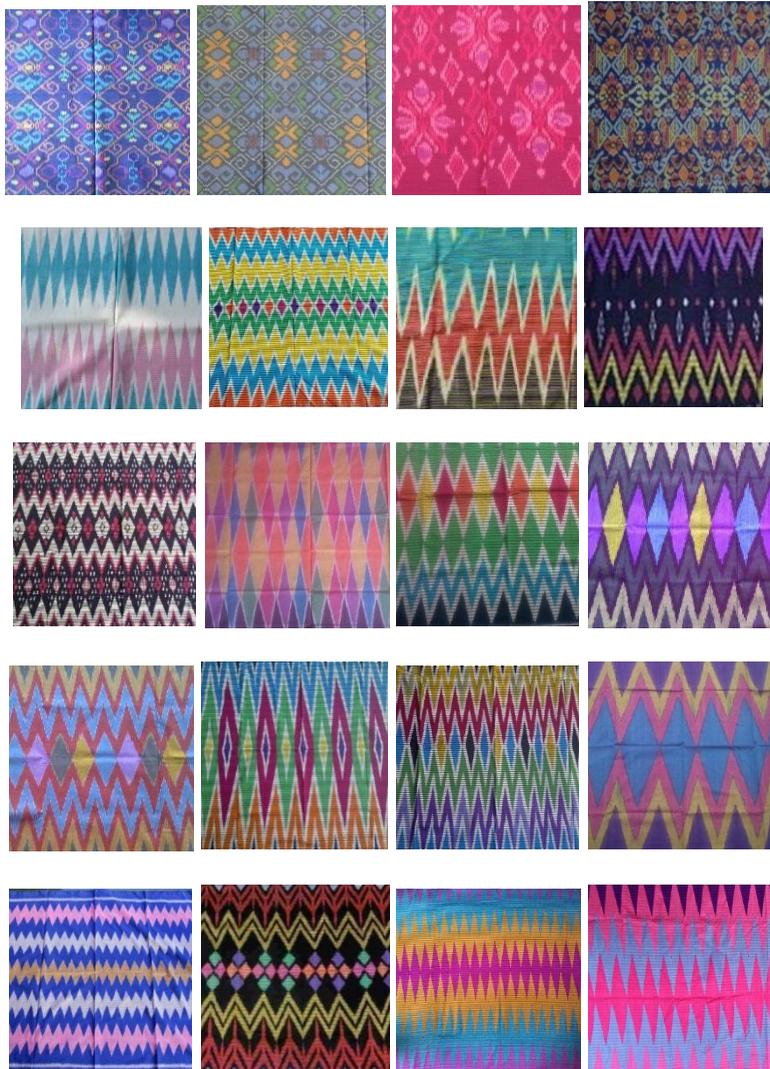
LAMPIRAN F (LANJUTAN)

F.4 Data Citra Tenun yang Telah Melalui Proses *Cropping*

Semua Citra di *cropping* menjadi berukuran 300×300 piksel



LAMPIRAN F (LANJUTAN)



LAMPIRAN G

G.1 Tabel Nilai Keanggotaan Masing-masing Citra pada Tiap Kelas

Data Citra	Derajat Keanggotaan Pada Kelas			Hasil
	Kelas Bali	Kelas NTT	Kelas Jawa Tengah	
Tenun 1	0,6226	0,1842	0,1932	Kelas Bali
Tenun 2	0,5914	0,1253	0,2834	Kelas Bali
Tenun 3	0,7071	0,1306	0,1624	Kelas Bali
Tenun 4	0,7303	0,1010	0,1687	Kelas Bali
Tenun 5	0,5080	0,1079	0,3841	Kelas Bali
Tenun 6	0,5767	0,1433	0,2800	Kelas Bali
Tenun 7	0,4170	0,1708	0,4122	Kelas Bali
Tenun 8	0,7533	0,0795	0,1672	Kelas Bali
Tenun 9	0,5413	0,0707	0,3880	Kelas Bali
Tenun 10	0,6618	0,1296	0,2086	Kelas Bali
Tenun 11	0,5239	0,2117	0,2644	Kelas Bali
Tenun 12	0,7006	0,0955	0,2040	Kelas Bali
Tenun 13	0,8482	0,0497	0,1021	Kelas Bali
Tenun 14	0,8306	0,0460	0,1234	Kelas Bali
Tenun 15	0,5069	0,2736	0,2195	Kelas Bali
Tenun 16	0,4389	0,1125	0,4486	Kelas Jawa Tengah
Tenun 17	0,6980	0,1095	0,1925	Kelas Bali
Tenun 18	0,5314	0,2277	0,2410	Kelas Bali
Tenun 19	0,2405	0,0806	0,6789	Kelas Jawa Tengah
Tenun 20	0,6042	0,1097	0,2861	Kelas Bali
Tenun 21	0,1881	0,1231	0,6888	Kelas Jawa Tengah
Tenun 22	0,2114	0,1369	0,6517	Kelas Jawa Tengah
Tenun 23	0,2747	0,1125	0,6128	Kelas Jawa Tengah
Tenun 24	0,3542	0,2074	0,4384	Kelas Jawa Tengah

LAMPIRAN G (LANJUTAN)

Data Citra	Derajat Keanggotaan Pada Kelas			Hasil
	Kelas Bali	Kelas NTT	Kelas Jawa Tengah	
Tenun 25	0,3091	0,3284	0,3625	Kelas Jawa Tengah
Tenun 26	0,1953	0,1080	0,6967	Kelas Jawa Tengah
Tenun 27	0,2433	0,1625	0,5942	Kelas Jawa Tengah
Tenun 28	0,2338	0,1103	0,6559	Kelas Jawa Tengah
Tenun 29	0,2996	0,0952	0,6052	Kelas Jawa Tengah
Tenun 30	0,2985	0,3483	0,3532	Kelas Jawa Tengah
Tenun 31	0,2761	0,3217	0,4022	Kelas Jawa Tengah
Tenun 32	0,1899	0,0751	0,7350	Kelas Jawa Tengah
Tenun 33	0,1617	0,0778	0,7605	Kelas Jawa Tengah
Tenun 34	0,3046	0,1528	0,5426	Kelas Jawa Tengah
Tenun 35	0,2734	0,3547	0,3719	Kelas Jawa Tengah
Tenun 36	0,2841	0,2291	0,4867	Kelas Jawa Tengah
Tenun 37	0,1967	0,1103	0,6929	Kelas Jawa Tengah
Tenun 38	0,3097	0,2288	0,4615	Kelas Jawa Tengah
Tenun 39	0,2788	0,2136	0,5076	Kelas Jawa Tengah
Tenun 40	0,1978	0,1077	0,6945	Kelas Jawa Tengah
Tenun 41	0,2167	0,5605	0,2228	Kelas NTT
Tenun 42	0,2367	0,5536	0,2098	Kelas NTT
Tenun 43	0,2528	0,5203	0,2269	Kelas NTT
Tenun 44	0,2627	0,5059	0,2314	Kelas NTT
Tenun 45	0,3725	0,2366	0,3909	Kelas Jawa Tengah
Tenun 46	0,2487	0,5148	0,2364	Kelas NTT
Tenun 47	0,2778	0,4876	0,2347	Kelas NTT
Tenun 48	0,2826	0,4727	0,2446	Kelas NTT
Tenun 49	0,3919	0,1556	0,4525	Kelas Jawa Tengah
Tenun 50	0,2636	0,2721	0,4643	Kelas Jawa Tengah

LAMPIRAN G (LANJUTAN)

Data Citra	Derajat Keanggotaan Pada Kelas			Hasil
	Kelas Bali	Kelas NTT	Kelas Jawa Tengah	
Tenun 51	0,2893	0,4034	0,3073	Kelas NTT
Tenun 52	0,1953	0,5905	0,2141	Kelas NTT
Tenun 53	0,2243	0,5113	0,2644	Kelas NTT
Tenun 54	0,3214	0,4191	0,2596	Kelas NTT
Tenun 55	0,3300	0,4947	0,1753	Kelas NTT
Tenun 56	0,2302	0,1145	0,6553	Kelas Jawa Tengah
Tenun 57	0,2450	0,5093	0,2456	Kelas NTT
Tenun 58	0,3276	0,4191	0,2534	Kelas NTT
Tenun 59	0,2569	0,5566	0,1865	Kelas NTT
Tenun 60	0,2504	0,4943	0,2552	Kelas NTT

G.2 Tabel Vektor Pusat Kelas

Pusat Kelas Bali	Pusat Kelas NTT	Pusat Kelas Jawa Tengah
1,5241	2,3449	1,6173
0,7902	1,1284	0,5834
1,1544	1,5344	0,7735
1,9583	2,6208	1,3727
0,8090	1,2403	0,5501
0,7912	1,1211	0,5839
0,7902	1,1284	0,5834
0,8102	1,2339	0,5494
1,8603	3,8920	2,1407
0,9136	1,2044	0,6517
1,4046	1,3576	0,8977
2,2123	3,2203	1,5775
1,0355	1,2582	0,6285

LAMPIRAN G (LANJUTAN)

Pusat Kelas Bali	Pusat Kelas NTT	Pusat Kelas Jawa Tengah
0,9218	1,2162	0,6584
0,9136	1,2044	0,6517
1,0514	1,2590	0,6311
1,6329	1,7641	2,1389
0,9747	0,9924	0,6761
1,6074	1,0919	1,0804
2,5098	4,0791	1,8052
1,3629	1,1015	0,6862
0,9747	0,9643	0,6905
0,9747	0,9924	0,6761
1,3237	1,0780	0,6675
1,7332	1,6257	1,4176
1,2500	0,9330	0,6855
1,7728	1,0898	1,2644
2,5930	4,5935	2,0246
1,8274	0,9124	0,7568
1,2635	0,9453	0,7079
1,2500	0,9330	0,6855
1,8563	0,8952	0,7519
1,9360	1,2492	1,3922
1,4497	0,7175	0,6858
1,8099	1,0934	1,5198
2,8261	5,7938	2,5378
2,1384	0,7849	0,8155
1,4528	0,7117	0,7025
1,4497	0,7175	0,6858
2,1225	0,8042	0,8114

LAMPIRAN G (LANJUTAN)**G.3 Tabel Hasil Pengujian Pengulangan dengan Pengambilan Data Secara Acak pada Database**

Uji ke-	citra 1	citra 2	citra 3	citra 4	citra 5	citra 6	citra 7	citra 8	citra 9	citra 10	Jumlah Benar	Akurasi
1	b	b	b	b	b	b	b	b	b	b	10	100%
2	b	b	b	b	b	b	b	b	b	b	10	100%
3	b	b	b	b	b	b	b	b	b	b	10	100%
4	b	b	b	b	b	b	b	b	b	b	10	100%
5	b	b	b	b	b	b	b	b	b	b	10	100%
6	b	b	b	b	b	b	b	b	b	b	10	100%
7	b	b	b	b	b	b	b	b	b	b	10	100%
8	b	b	b	b	b	b	b	b	b	b	10	100%
9	b	b	b	b	b	b	b	b	b	b	10	100%
10	b	b	b	b	b	b	b	b	b	b	10	100%
Rata-rata											100%	

LAMPIRAN G (LANJUTAN)**G.4 Tabel Hasil Pengujian Pengulangan dengan Pengambilan Data Secara Acak pada Data Uji**

Uji ke-	citra 1	citra 2	citra 3	citra 4	citra 5	citra 6	citra 7	citra 8	citra 9	citra 10	Jumlah Benar	Akurasi
1	b	s	b	b	b	b	b	b	b	b	9	90%
2	s	b	b	b	b	b	b	b	b	b	9	90%
3	b	b	b	b	b	b	b	b	b	b	10	100%
4	b	b	s	b	b	b	b	b	b	b	9	90%
5	b	b	b	b	b	b	b	b	b	b	10	100%
6	b	b	b	b	b	b	b	b	b	b	10	100%
7	s	b	b	b	b	b	b	b	b	b	9	90%
8	s	b	b	b	b	b	b	b	b	b	9	90%
9	b	b	b	b	b	b	b	b	b	b	10	100%
10	b	b	s	b	b	b	b	b	b	b	9	90%
Rata-rata												94%

BIODATA PENULIS



Penulis bernama lengkap 'Isyi Bifawa'idati, yang biasa dipanggil Isyi/Faid. Penulis dilahirkan di Gresik pada tanggal 6 Maret 1994. Penulis lulus dari SD NU 1 Gresik, melanjutkan pendidikan di SMP N 1 Gresik, selanjutnya meneruskan pendidikan di SMA N 1 Gresik dan melanjutkan pendidikan di Matematika ITS pada tahun 2012.

Semasa menempuh jenjang pendidikan S-1, penulis juga aktif dalam kegiatan non-akademis diantaranya aktif di beberapa organisasi kemahasiswaan ITS yaitu di Departemen KESMA HIMATIKA ITS periode 2013/2014, Tim kestari BEM FMIPA ITS periode 2013/2014, Sekretaris Umum BEM FMIPA ITS periode 2015/2016, dan Anggota UKM IFLS periode 2012/2013. Untuk mendapatkan informasi yang berhubungan dengan Tugas Akhir ini dapat ditunjukkan ke alamat email : isyifaid@gmail.com