



INSTITUT TEKNOLOGI DEL

Pembangunan Modul untuk Penentuan Kriteria Kualitas Gambar pada Aplikasi JTenun

Tugas Akhir

Oleh :

11315010 Abed Nego Lubis
11315023 Hartanti Saragih
11315032 Grace Naomi Damanik

**FAKULTAS TEKNIK INFORMATIKA DAN ELEKTRO
PROGRAM STUDI DIII TEKNIK INFORMATIKA
LAGUBOTI
September 2018**



INSTITUT TEKNOLOGI DEL

Pembangunan Modul untuk Penentuan Kriteria Kualitas Gambar pada Aplikasi JTenun

Tugas Akhir

Diajukan sebagai salah satu syarat untuk memperoleh gelar Diploma III

Oleh :

11315010 Abed Nego Lubis
11315023 Hartanti Saragih
11315032 Grace Naomi Damanik

**FAKULTAS TEKNIK INFORMATIKA DAN ELEKTRO
PROGRAM STUDI DIII TEKNIK INFORMATIKA
LAGUBOTI
2017/2018**

HALAMAN PERNYATAAN ORISINALITAS

**Tugas Akhir ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Abed Nego Lubis

NIM : 11315020

Tanda Tangan :

Tanggal :

Nama : Hartanti Saragih

NIM : 11315023

Tanda Tangan :

Tanggal :

Nama : Grace Naomi Damanik

NIM : 11315032

Tanda Tangan :

Tanggal :

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh :

- | | |
|-------------------|---|
| 1. Nama | : Abed Nego Lubis |
| NIM | : 11315020 |
| Program studi | : D3 Teknik Informatika |
| 2. Nama | : Hartanti Saragih |
| NIM | : 11315023 |
| Program studi | : D3 Teknik Informatika |
| 3. Nama | : Grace Naomi Damanik |
| NIM | : 11315032 |
| Program studi | : D3 Teknik Informatika |
| Judul Tugas Akhir | : Pembangunan Modul untuk Penentuan
Kriteria Kualitas Gambar pada Aplikasi
Jtenun |

Telah berhasil dipertahankan dihadapannya Dewan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Diploma III, pada program studi Teknik Informatika, Fakultas Teknik Informatika dan Elektro Institut Teknologi Del.

DEWAN PENGUJI

Pembimbing 1	: Dr. Arlinta Christy Barus, S.T., M.InfoTech	()
Pembimbing 2	: Yaya Setiadi, S.Si.,M.T.	()
Penguji 1	: Anthon Roberto Tampubolon, S.Kom, M.T	()
Penguji 2	: Arie Satia Dharma, S.T, M.Kom	()

Ditetapkan di : Sitoluama

Tanggal :

KATA PENGANTAR

Puji dan syukur kepada Tuhan Yang Maha Esa atas rahmat-Nya yang menyertai penulis selama pengerjaan Tugas Akhir ini, sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik. Laporan Tugas Akhir ini ditulis sebagai syarat kelulusan Diploma III Institut Teknologi Del. Laporan Tugas Akhir ini bertujuan untuk mendokumentasikan hasil dari Tugas Akhir mengenai ***“Pembangunan Modul untuk Penentuan Kriteria Kualitas Gambar pada Aplikasi JTenun”***

Penulis menyampaikan terima kasih kepada pembimbing, Ibu Dr Arlinta Barus, S.T., M. Infotech dan juga kepada Bapak Yaya Setiyadi S.Si., M.T yang telah memberikan ide, arahan, dan bimbingan selama pengerjaan Tugas Akhir ini. Penulis juga mengucapkan terima kasih kepada Bapak Yohanssen Pratama, S.Si, M.T dan Ibu Hilda Deborah, PhD yang telah membantu memberikan arahan dan bimbingan selama pengerjaan Tugas Akhir ini. Serta kepada Bapak Anthon Roberto Tampubolon, S.Kom, M.T sebagai koordinator TA 2017/2018, sekaligus sebagai ketua penguji dan juga kepada Bapak Arie Satia Dharma, S.T, M.Kom sebagai anggota penguji pada saat Seminar Proposal, Seminar TA1, PraSidang, dan Sidang TA 2 serta semua pihak yang telah membantu dan mendukung penulis selama proses pengerjaan Tugas Akhir ini sehingga dapat berjalan dengan baik.

Penulis berharap laporan Tugas Akhir ini dapat bermanfaat bagi semua pihak yang memerlukannya. Penulis menyadari bahwa laporan ini masih memiliki banyak kekurangan, untuk itu penulis mengharapkan kritik dan saran yang membangun untuk perbaikan dokumen ini sehingga kedepannya dokumen ini menjadi dokumen yang lebih baik.

Sitoluama, 30 Agustus 2018

11315020 Abed Nego Lubis
11315023 Hartanti Saragih
11315032 Grace Naomi Damanik

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai civitas akademik Institut Teknologi Del, saya yang bertanda tangan di bawah ini:

Nama / NIM : Abed Nego Lubis / 11315020
Hartanti Saragih / 11315023
Grace Naomi Damanik / 11315032

Program Studi : D3 Teknik Informatika

Fakultas : Teknik Informatika & Elektro

Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Institut

Teknologi Del **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul:

**Pembangunan Modul untuk Penentuan Kriteria Kualitas Gambar pada
Aplikasi JTenun**

beserta perangkat yang ada (jika di perlukan). Dengan Hak Bebas Royalti Noneksklusif ini Institut Teknologi Del berhak menyimpan, mengalih/media-format dalam bentuk pangkalan data (database), merawat,dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Sitoluama

Pada tanggal : 30 Agustus 2018

Yang menyatakan:

Abed Nego Lubis

Hartanti Saragih

Grace Naomi Damanik

ABSTRAK

Nama : Abed Nego Lubis / Hartanti Saragih / Grace Naomi Damanik
Program Studi : D3 Teknik Informatika
Judul : Pembangunan Modul untuk Penentuan Kriteria Kualitas Gambar pada Aplikasi JTenun

Ulos adalah selembar kain yang ditenun sebagai kerajinan oleh wanita dengan berbagai motif, pola dan aturan-aturan. Namun variasi motif yang dihasilkan masih terbatas sehingga banyak potensi ekonomi yang belum dapat dikembangkan. Oleh karena itu, dikembangkanlah sebuah aplikasi yang akan membantu para penenun untuk menghasilkan motif Ulos yang baru yang dikenal dengan nama aplikasi JTenun. Pada pengembangan aplikasi ini dibutuhkan sebuah fitur untuk menangkap gambar ulos secara langsung dari kamera. Gambar yang ditangkap secara langsung tidak selalu menghasilkan kualitas gambar yang baik, maka dari itu peneliti juga akan mengklasifikasikan gambar Ulos untuk menentukan kualitas gambar Ulos yang dijadikan sebagai gambar masukan. Penentuan kualitas gambar masukan dapat dilihat dari dua parameter yang mempengaruhi kualitas suatu gambar yakni parameter *blur* dan parameter *noise*. Kedua parameter yang digunakan akan dideteksi dengan menggunakan dua algoritma yang berbeda yaitu algoritma FFT (*fast fourier transform*) dan PCA (*principal component analysis*). Algoritma ini akan bekerja untuk mendeteksi kedua parameter tersebut pada gambar masukan sehingga aplikasi ini akan mampu menghasilkan kualitas gambar yang lebih baik.

Kata Kunci : Tenun, Ulos, JTenun, Kualitas gambar, *Blur*, *Noise*, *FFT*, *PCA*

Ulos is a piece of cloth woven as a craft by women with various motives, patterns and rules. But the variety of motives produced is still limited so that a lot of economic potential has not been developed. Therefore, an application was developed that would help the weavers to produce new Ulos motifs known as JTenun applications. In developing this application, a feature is needed to capture ulos images directly from the camera. Images captured directly do not always produce good image quality, so researchers will also classify Ulos images to determine the image quality of Ulos that is used as an input image. Determination of input image quality can be seen from two parameters that affect the quality of an image, namely the blur parameters and noise parameters. The two parameters used will be detected using two different algorithms namely FFT (fast fourier transform) algorithm and PCA (principal component analysis). This algorithm will work to detect both parameters in the input image so that this application will be able to produce better image quality.

Keywords: Weaving, Ulos, JTenun, Image Quality, Blur, Noise, FFT, PCA

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	iii
HALAMAN PENGESAHAN	iv
KATA PENGANTAR	v
ABSTRAK	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
1. Pendahuluan	1
1.1 Latar Belakang	1
1.2 Tujuan.....	2
1.3 Lingkup	2
1.4 Pendekatan.....	2
1.5 Sistematika Penyajian.....	3
2. Tinjauan Pustaka	5
2.1 Tenun Ulos	5
2.1.1 Klasifikasi Berdasarkan Nilai Adat	5
2.2 Kualitas Gambar	6
2.3 Blur	7
2.4 Noise.....	9
2.5 Fast Fourier Transform (FFT)	10
2.6 Laplacian Operator	11
2.7 Principle Component Analysis (PCA)	11
2.8 Weak Textured Patched (WTP).....	13
2.9 Tools Terkait	13
2.9.1 Matlab	14
2.9.2 Open CV	14
2.9.3 Python	14
2.9.4 JTenun.....	15
2.10 Modul Pendukung	16
2.10.1 Modul Enhancement	16
2.10.2 Modul Classification.....	17
2.11 Penelitian Terkait	17
2.12 Kesimpulan Tinjauan Pustaka	19
3. Analisis	20
3.1 Tahapan Analisis	20

3.2	Analisis Algoritma Fast Fourier Transform dan Laplacian Operator	20
3.3	Analisis Algoritma PCA.....	23
3.4	Analisis kaitan parameter blur dan noise dengan karakteristik motif Ulos	24
3.5	Kesimpulan Analisis.....	25
4.	Desain Eksperimen	26
4.1	Objek Eksperimen	26
4.2	Desain Eksperimen.....	27
4.2.1	Parameter Blur	28
4.2.2	Parameter Noise	31
4.2.3	Parameter Blur dan Noise	34
4.3	Tools/perangkat yang digunakan untuk eksperimen	35
4.3.1	Hardware.....	35
4.3.2	Software	36
5.	Hasil Eksperimen dan Pembahasan	37
5.1	Hasil Eksperimen Parameter Blur	37
5.2	Hasil Eksperimen Parameter Noise	41
5.3	Hasil Eksperimen Parameter Blur dan Noise	44
5.4	Hasil Eksperimen Evaluasi Gambar	45
5.5	Hasil Implementasi GUI.....	46
5.6	Pembahasan Hasil.....	47
6.	Kesimpulan dan Saran	48
6.1	Kesimpulan.....	48
6.2	Saran	48
	Daftar Pustaka.....	50

DAFTAR TABEL

Tabel 1. Ringkasan Penelitian Terkait	17
Tabel 2. Spesifikasi Hardware	35
Tabel 3. Sampel gambar Ulos kategori bad berdasarkan parameter blur	37
Tabel 4. Sampel gambar Ulos kategori improve berdasarkan parameter blur	39
Tabel 5. Sampel gambar Ulos kategori good berdasarkan parameter blur	40
Tabel 6. Sampel gambar Ulos kategori improve berdasarkan parameter noise	42
Tabel 7. Sampel gambar Ulos kategori good berdasarkan parameter noise	43
Tabel 8. Thresold parameter blur	45
Tabel 9. Thresold parameter noise	45

DAFTAR GAMBAR

Gambar 1 Contoh Gambar Tidak Blur	7
Gambar 2 Contoh Gambar Blur	7
Gambar 3 Contoh gambar efek blur yang diinginkan	8
Gambar 4 Contoh gambar efek blur yang tidak diinginkan	8
Gambar 5 Tahapan pengolahan citra dengan PCA	12
Gambar 6 kernel 3x3	21
Gambar 7 Ilustrasi kernel	21
Gambar 8 Contoh matriks gambar masukan	22
Gambar 9 Rumus Varians	22
Gambar 10 Patches dengan nilai eigen	23
Gambar 11 Proses evaluasi gambar Ulos berdasarkan parameter blur	29
Gambar 12 Proses evaluasi gambar Ulos berdasarkan parameter noise	32
Gambar 13 Proses Evaluasi Gambar blur dan noise	34
Gambar 14 Tampilan fitur pendeteksi blur pada gambar Ulos	46
Gambar 15 Tampilan fitur pendeteksi noise pada gambar Ulos	46

1. Pendahuluan

Pada Bab I dijelaskan mengenai latar belakang pemilihan topik Tugas Akhir, tujuan, lingkup pengerjaan, pendekatan dan sistematika penyajian yang digunakan selama pengerjaan dan penyelesaian persoalan yang menjadi objek kajian Tugas Akhir.

1.1 Latar Belakang

Indonesia adalah salah satu negara yang menghasilkan seni tenun yang terbesar terutama dalam hal keanekaragaman hiasannya [18]. Tenun merupakan teknik dalam pembuatan kain yang dibuat dengan asas (prinsip) yang sederhana yaitu dengan menggabungkan benang secara memanjang dan melintang [3]. Salah satu bentuk tenun yang berasal dari Sumatera adalah Ulos. Ulos adalah selembur kain yang ditenun sebagai kerajinan oleh wanita dengan berbagai pola dan aturan-aturan [14]. Ulos dibedakan atas proses pembuatan gorga atau motif beserta coraknya [13]. Setiap Ulos memiliki ciri khas dan makna tersendiri dari setiap motif yang dihasilkan, namun variasi motif yang dihasilkan masih terbatas sehingga banyak potensi ekonomi yang belum dapat dikembangkan. Berdasarkan hal tersebut, dikembangkanlah sebuah aplikasi yang akan membantu para penenun untuk menghasilkan motif Ulos yang baru yang dikenal dengan nama aplikasi *mobile JTenun* [1].

Aplikasi JTenun memiliki 4 (empat) modul utama, yaitu: 1. *Tenun Editor*, 2. *Core Learning System*, 3. *Tenun Catalogue*, 4. *Data Collecting Interface*. Pada Tugas Akhir ini, peneliti akan berfokus pada pengerjaan salah satu modul yaitu *Tenun Editor*. Modul ini merupakan modul utama dimana penenun dan piranti berinteraksi [1]. Salah satu fungsi dari modul ini adalah sebagai antarmuka tempat pengguna memasukkan gambar Ulos ke dalam aplikasi yang dijadikan sebagai acuan atau sumber gambar untuk menghasilkan motif Ulos baru.

Pada Tugas Akhir ini, peneliti akan membangun sebuah modul yang membuat para penenun mampu secara langsung menangkap gambar Ulos dengan kamera *smartphone* dan menjadikannya sebagai masukan untuk menghasilkan motif baru. Gambar yang ditangkap secara langsung tidak selalu menghasilkan

kualitas gambar yang baik, maka dari itu peneliti juga akan mengklasifikasikan gambar Ulos untuk menentukan kualitas gambar Ulos yang dimasukkan.

Pengklasifikasian ini adalah salah satu upaya untuk menentukan kualitas pada sebuah gambar masukan. Penentuan kualitas gambar masukan dapat dilihat dari beberapa parameter yang mempengaruhi [2]. Kemudian, peneliti mengambil 2 (dua) bentuk parameter paling umum yaitu efek *blur* dan *noise*. Efek *blur* adalah *distorsi* gambar yang umum terjadi di bidang fotografi [6] dan *noise* adalah gambar atau piksel yang mengurangi kualitas citra dalam pengolahan citra. Pendeteksian tingkat *blur* akan dilakukan menggunakan *OpenCV*, *Python*, dan *Laplacian operator* [15] sedangkan untuk pendeteksian *noise* peneliti akan menggunakan algoritma *principal component analysis* dari *weak textured patches* [22].

1.2 Tujuan

Tujuan dari Tugas Akhir ini adalah membangun sebuah modul pada aplikasi JTenun untuk menentukan klasifikasi kualitas gambar Ulos yaitu gambar diterima untuk diproses ke tahapan selanjutnya, gambar diperbaiki agar mampu menuju tahapan selanjutnya, atau gambar ditolak dan tidak bisa menuju tahapan selanjutnya.

1.3 Lingkup

Lingkup masalah yang akan dicakup pada Tugas Akhir ini:

1. Membangun sebuah modul untuk menentukan kualitas gambar Ulos yang dijadikan sebagai gambar masukan aplikasi *mobile* JTenun. Dalam penentuan kualitas gambar, peneliti memilih *blur* dan *noise* dijadikan sebagai parameter kualitas gambar Ulos.
1. Metode yang diusulkan pada Tugas Akhir ini hanya bertujuan untuk menentukan kualitas gambar yang baik untuk tahapan selanjutnya namun tidak untuk memperbaiki gambar tersebut.
2. Hasil dari Tugas Akhir ini berupa modul yang mampu mengklasifikasikan gambar masukan berupa gambar Ulos yang diterima (*good*), diperbaiki (*improve*), atau ditolak (*bad*).

1.4 Pendekatan

Metodologi penelitian yang akan dilakukan pada Tugas Akhir ini adalah:

1. Studi Literatur

Studi Literatur yang dimaksud dalam konteks ini adalah mengumpulkan informasi terkait dengan menentukan kualitas gambar dengan pengklasifikasian untuk dijadikan sebagai gambar masukan, serta mencari informasi mengenai kedua parameter yang mempengaruhi kualitas gambar yaitu *blur* dan *noise*.

2. Analisis dan Perancangan

Pada tahap analisis dan perancangan dilakukan analisis mengenai informasi teknik atau algoritma pendeteksian terhadap parameter *blur* dan *noise* pada gambar masukan serta dilakukan juga eksplorasi *tools* yang terkait dengan pendeteksian kedua parameter

3. Implementasi

Pada tahapan implementasi dilakukan eksperimen di *tools* berupa program yang digunakan untuk menjalankan algoritma dari hasil tahapan perancangan dan analisis.

4. Evaluasi

Pada tahap ini merupakan tahapan untuk memberikan penilaian dan pengklasifikasian terhadap hasil eksperimen yang sudah dilakukan, apakah kualitas gambar yang diklasifikasikan sudah baik untuk dijadikan sebagai gambar masukan.

1.5 Sistematika Penyajian

Sistematika penulisan laporan Tugas Akhir adalah sebagai berikut:

1. Bab I berisi latar belakang, tujuan, lingkup, pendekatan, dan sistematika penyajian selama pengerjaan Tugas Akhir.
2. Bab II dijelaskan mengenai rangkuman informasi yang telah dihimpun dari berbagai pustaka dan kesimpulan dari studi literatur yang telah dilakukan.
3. Bab III menjelaskan analisis terhadap beberapa algoritma serta *tools* yang akan digunakan untuk penentuan kualitas gambar Ulos serta proses evaluasi gambar Ulos.
4. Bab IV berisi penjelasan tahapan desain eksperimen yang akan dilakukan untuk penentuan kualitas gambar Ulos berdasarkan parameter *blur* dan *noise* pada gambar.
5. Bab V memaparkan pelaksanaan eksperimen dan hasil serta pembahasan mengenai klasifikasi gambar Ulos berdasarkan kedua parameter yaitu *blur* dan *noise*.
6. Bab VI merupakan kesimpulan dan saran yang dapat diambil dari penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.

2. Tinjauan Pustaka

Pada bab ini, dijelaskan mengenai *study literature* beserta hasil observasi terhadap algoritma yang akan digunakan untuk mengembangkan Pembangunan Modul untuk Penentuan Kriteria Kualitas Gambar pada Aplikasi JTenun.

2.1 Tenun Ulos

Secara harfiah, Ulos berarti selimut yang menghangatkan tubuh dan melindunginya dari terpaan udara dingin. Menurut kepercayaan leluhur suku Batak ada tiga sumber yang memberi panas kepada manusia, yaitu matahari, api, dan Ulos. Dari ketiga sumber kehangatan tersebut Ulos dianggap paling nyaman dan akrab dengan kehidupan sehari-hari [23]. Ulos sebagai produk budaya asli suku Batak menjadi kebutuhan primer, karena pemakaian dari Ulos semakin meluas, tidak hanya sekedar penghangat badan. Motif-motif Ulos yang selama ini telah ada masih terus dikembangkan tanpa menghilangkan corak motif Ulos yang lama. Selain itu, Ulos memiliki makna yang lebih penting atau pun sakral dalam adat ketika mulai dipakai oleh tetua-tetua adat dan para pemimpin kampung dalam pertemuan-pertemuan adat resmi. Ditambah lagi dengan kebiasaan para leluhur suku Batak yang selalu memilih Ulos untuk dijadikan hadiah atau pemberian kepada orang-orang yang mereka sayangi.

2.1.1 Klasifikasi Berdasarkan Nilai Adat

Salah satu pengklasifikasian Ulos adalah klasifikasi berdasarkan nilai adat. Klasifikasi Ulos berdasarkan daerah dijelaskan sebagai berikut (Takari, 2009):

1. Ragi idup

Ulos Ragi idup menggambarkan garis hidup seseorang, terutama dalam hidup berkeluarga.

2. Ragi Hotang

Ulos Ragi Hotang atau corak rotan, menggambarkan keeratan hubungan kekerabatan bagaikan kuatnya ikatan rotan.

3. Sadum

Ulos Sadum menggambarkan sukacita dan dukacita.

4. Bintang Maratur

Ulos Bintang Maratur secara harfiah artinya adalah bintang yang teratur. Makna dari kain ini adalah melambangkan harapan kehidupan yang teratur dan bersahaja. Kehidupan yang rukun dan berbahagia

5. Mangiring

Ulos Mangiring melambangkan kesuburan dan kesepakatan.

6. Sibolang

Ulos Sibolang merupakan Ulos yang digunakan untuk acara berkabung menunjukkan duka.

7. Harungguan

Ulos Harungguan merupakan dokumentasi beberapa corak ikat yang dibuat oleh penenun masing-masing.

8. Sitolutuho

Ulos Sitolutuho digunakan untuk pemakaian sehari-hari, biasanya digunakan untuk para petani.

2.2 Kualitas Gambar

Pengukuran pada kualitas gambar adalah ukuran penting untuk mengetahui kualitas gambar. Karena sebagian besar gambar dapat mengalami banyak distorsi, jadi diperlukan pemrosesan pada distorsi ini untuk meningkatkan kualitas pada gambar. Kualitas gambar dapat diukur dengan menggunakan metode objektif dan subjektif. Berikut akan dijelaskan keuntungan dan kerugian dari setiap metodenya [4]:

1. Metode objektif

Metode objektif adalah proses memperkirakan kualitas gambar menggunakan algoritma berbasis komputer, tanpa bantuan pengamat manusia. Adapun dari keuntungan ini adalah metode ini dapat secara otomatis memprediksi kualitas gambar yang dirasakan. Sedangkan kerugiannya adalah kualitas gambar yang diprediksi melalui metode objektif tidak sesuai dengan kualitas gambar seperti yang dirasakan oleh manusia.

2. Metode subjektif

Proses memperkirakan kualitas gambar menggunakan pengamat manusia dikenal sebagai metode subjektif dari estimasi kualitas gambar. Keuntungan

dari metode ini ialah metode ini akurat dalam memperkirakan kualitas visual suatu gambar karena mereka dilakukan oleh subjek manusia dan kerugiannya adalah proses ini membutuhkan biaya mahal yang membutuhkan sejumlah besar pengamat dan juga membutuhkan lebih banyak waktu.

2.3 Blur

Salah satu masalah yang sering muncul dalam dunia fotografi adalah efek *blur* yang dapat diakibatkan baik oleh objek yang bergerak maupun gerakan kamera yang berhubungan dengan kecepatan rana (*shutter speed*) ketika gambar akan diambil [6]. Alasan utama yang ada di balik hasil gambar *blur* adalah karena pergerakan lensa yang dinamis selama proses pengambilan, dimana lensa tidak dapat menetapkan sudut dan fokus yang tepat dan oleh karena itu ada keluaran gambar *blur* [15]. Untuk memperjelas perbedaan gambar *blur* dengan tidak *blur* akan diperlihatkan perbedaan gambar *blur* dengan gambar tidak *blur* pada Gambar 1 dan Gambar 2.



Gambar 1 Contoh Gambar Tidak Blur



Gambar 2 Contoh Gambar Blur

Gambar yang *blur* (kabur) bisa juga muncul karena berbagai alasan lainnya, seperti sengaja ditambahkan pada gambar untuk menimbulkan sensasi gerak cepat dari objek gambar. Fotografer sering menggunakan efek ini untuk menghasilkan gambar yang dramatis sehingga gambar yang dihasilkan akan dianggap memiliki daya tarik lebih. Gambar 3 dan 4 menampilkan beberapa contoh efek gerak *blur* (*motion blur*). Gambar 3 adalah contoh efek gerak *blur* yang diinginkan untuk memberikan sensasi gerakan cepat. Gambar ini dibuat dengan menggunakan kecepatan rana yang lambat pada kamera apabila dibandingkan dengan pergerakan objek ketika gambar diambil. Efek *blur* pada Gambar 4 adalah distorsi yang tidak diinginkan karena membuat detail gambar menjadi kabur. Distorsi ini mungkin terjadi karena proses pemindaian gambar yang kurang sempurna atau karena kegoyangan kamera pada saat pemotretan (*camera shake*) [6].



Gambar 3 Contoh gambar efek blur yang diinginkan



Gambar 4 Contoh gambar efek blur yang tidak diinginkan

Dalam gambar digital ada tiga jenis efek *blur* [19]:

1. *Average Blur*

Average Blur adalah salah satu dari beberapa jenis efek *blur* yang dapat digunakan untuk menghilangkan *noise* dan bintik-bintik dalam gambar. Gunakan saat *noise* hadir di seluruh gambar. Jenis pengaburan ini dapat didistribusikan dalam arah *horizontal* dan *vertikal* dan dapat melingkar rata-rata dengan radius yang dievaluasi.

2. *Gaussian Blur*

Efek *Gaussian Blur* adalah filter yang memadukan sejumlah piksel tertentu secara bertahap, mengikuti kurva berbentuk lonceng. Pengaburannya padat di bagian tengah dan bulu-bulu di tepi. *Gaussian Blur* dapat diterapkan ke gambar saat ingin lebih mengontrol efek *blur*.

3. *Motion Blur*

Efek *Motion Blur* adalah filter yang membuat gambar tampak bergerak dengan menambahkan *blur* ke arah tertentu. Gerakan dapat dikontrol oleh sudut atau arah (0 hingga 360 derajat atau -90 hingga $+90$) dan atau dengan jarak atau intensitas dalam piksel (0 hingga 999), berdasarkan pada perangkat lunak yang digunakan.

2.4 Noise

Noise adalah variasi acak dari intensitas gambar dan terlihat sebagai butiran pada gambar [4]. Secara umum, *noise* bisa muncul dikarenakan oleh beberapa faktor seperti proses *capture* yang tidak sempurna, pencahayaan yang tidak merata yang mengakibatkan intensitas tidak seragam, kontras citra terlalu rendah, pemakaian ISO tinggi atau gangguan fisik (optik), maupun disengaja karena proses pengolahan yang tidak sesuai dan lain sebagainya. Dijelaskan bahwa *noise* muncul biasanya sebagai akibat dari pembelokan yang tidak bagus (*sensor noise*, *photographic gain noise*). Maka dari itu, *noise* berarti bahwa gambar yang memiliki piksel yang nilai intensitasnya berbeda, bukan lagi nilai piksel yang benar. Dilihat dari karakteristik dan bentuknya, *noise* pada citra dibagi menjadi beberapa macam yaitu *Gaussian noise*, *Speckle noise*, *Salt and Pepper noise*, *Poisson Noise*. Contohnya adalah bintik hitam atau putih yang muncul secara acak yang tidak diinginkan di dalam citra. Bintik acak ini disebut dengan *Salt and Pepper noise*. Tergantung pada jenis

gangguan, *noise* dapat mempengaruhi gambar ke tingkat yang berbeda. Gangguan tersebut umumnya berupa variasi intensitas suatu piksel yang tidak berkorelasi dengan piksel-piksel tetangganya. Secara visual, gangguan mudah dilihat oleh mata karena tampak berbeda dengan piksel tetangganya. Piksel yang mengalami gangguan umumnya memiliki frekuensi tinggi. Komponen citra yang berfrekuensi rendah umumnya mempunyai nilai piksel konstan atau berubah sangat lambat. Operasi *denoise* dilakukan untuk menekan komponen yang berfrekuensi tinggi dan meloloskan komponen yang berfrekuensi rendah [24]. *Noise* akan tampak mengganggu pada hasil foto dan muncul berupa titik-titik warna yang tidak enak untuk dilihat. Penyebab lain yang memicu *noise* dapat terjadi yaitu dari proses pengolahan citra yang tidak sesuai atau terdapat penyimpangan data digital oleh alat penerima data gambar.

2.5 Fast Fourier Transform (FFT)

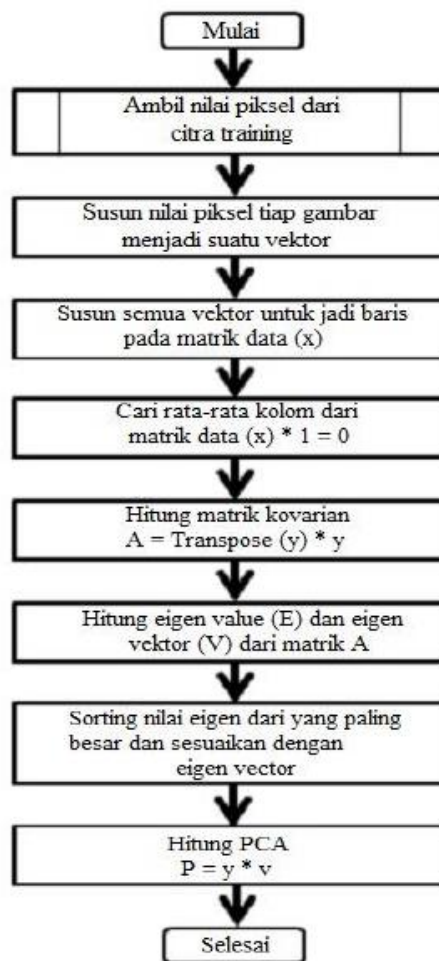
Fast Fourier Transform (FFT) adalah suatu transformasi yang mengubah data *digital* ke domain frekuensi. Inti dari transformasi *Fourier* adalah memecah *signal* (citra) menjadi gelombang-gelombang *sinusoidal* dimana jumlahnya sama dengan *signal* asalnya. *Fast Fourier Transform* (FFT) merupakan salah satu algoritma yang paling sering digunakan dalam menganalisis dan manipulasi data *digital* [7]. Penelitian (Rockmore, 2000) menunjukkan bahwa FFT dapat diterapkan untuk banyak hal, seperti *electroacoustic music* dan pengolahan sinyal audio, pengolahan citra, *medical imaging*, *pattern recognition*, *computational chemistry*, dan lain-lain [17]. Transformasi *fourier* mempunyai peranan penting dalam aplikasi pengolahan citra, dapat memberikan kontribusi untuk peningkatan mutu citra, analisis data citra seperti menganalisis kelakuan restorasi citra dan kompresi data citra [31]. Dalam transformasi *Fourier*, algoritma menghitung frekuensi pada gambar di titik-titik yang berbeda dan berdasarkan tingkat frekuensi itu, algoritma akan memutuskan apakah gambar buram atau tidak. Ketika *High Frequency* berjumlah rendah maka akan dinyatakan sebagai gambar *blur* dan keputusan untuk angka yang dijadikan sebagai *threshold* ditentukan berdasarkan *programmer* [9]. Pada pembelajaran *signal processing*, *high frequency* yang dimaksud adalah frekuensi tinggi pada sinyal yang cepat atau transisi tajam di diagram sinyal pemrosesan citra.

2.6 Laplacian Operator

Operator *Laplacian* diimplementasikan untuk menemukan tepi dalam gambar. Operator *Laplacian* selanjutnya dipisahkan menjadi dua klasifikasi lebih lanjut yaitu operator Negatif *Laplacian* dan operator Positif *Laplacian* [22]. Positif dan negatif tidak dapat diterapkan pada gambar yang sama. Pada operator *laplacian*, ada yang dinamakan dengan operator *resultan*. Dalam *Laplacian* negatif, *operator resultant* gambar ditambahkan ke gambar asli untuk mendapatkan efek dipertajam pada gambar. Sedangkan di *Laplacian* positif, *operator resultant* dikurangkan dari gambar asli untuk mendapatkan efek dipertajam gambar. Semakin tinggi nilai *output* dari *laplacian operator* maka gambar akan dianggap sebagai gambar yang tidak *blur*, begitu sebaliknya.

2.7 Principle Component Analysis (PCA)

Metode *Principal Component Analysis* (PCA) dibuat pertama kali oleh para ahli statistik dan ditemukan Karl Pearson pada tahun 1901 yang memakainya pada bidang biologi. PCA adalah transformasi linier untuk menentukan sistem koordinat yang baru dari dataset. Teknik PCA dapat mengurangi dimensi dari dataset tanpa menghilangkan informasi penting dari dataset. *Principal Component Analysis* (PCA) relatif mudah menangani sejumlah data yang cukup besar serta kemampuannya menangani data-data dimensi yang kompleks. Salah satunya, PCA berperan dalam mengolah citra. Pada Gambar 5 ditampilkan tahapan pengolahan citra dengan PCA, adalah:



Gambar 5 Tahapan pengolahan citra dengan PCA

1. *Input* gambar

Pada tahap awal, *input* semua citra *training* yang akan diproses. Citra *training* adalah citra yang akan digunakan untuk proses pelatihan.

2. Proses *threshold* citra

Pada proses ini, citra yang sudah dimasukkan akan diuji coba dengan nilai *threshold* pada rentang keabuan antara 0-255. Merujuk pada penelitian falasev et al. (2011), penelitian ini menggunakan nilai ambang atau rentang keabuan untuk mengetahui intensitas keabuannya. Pemberian nilai *threshold* ini dimaksudkan untuk membatasi intensitas keabuan yang akan dipakai. Selain itu, dengan nilai *threshold* yang berbeda-beda akan mendapatkan hasil analisis yang bervariasi pula karena rentang minimalnya mulai dari 0 dan rentang maksimalnya adalah 255.

3. *Segmentasi* matrik

Setelah dilakukan pemberian nilai *threshold*, dilanjutkan dengan segmentasi ukuran citra menjadi beberapa bagian. *Segmentasi* matriks ini bertujuan untuk memperkecil area perhitungan dan detail nilai citra yang akan diproses. Pada *segmentasi*, citra akan otomatis dibagi menjadi 4 bagian sama rata. Jika citra berukuran (250x250) piksel, maka tiap *segmen* bernilai (25x25) piksel. Dimana pada tiap bagian nanti akan dilanjutkan ekstraksi ciri menggunakan PCA.

4. Ekstraksi ciri menggunakan PCA

Proses ekstraksi ciri dilakukan setelah mendapatkan citra *grayscale* untuk selanjutnya diimplementasikan dalam metode PCA. Tahapan yang harus dilakukan adalah pertama, mengambil nilai piksel dari citra *training*. Tahapan selanjutnya adalah menyusun nilai piksel tiap gambar menjadi suatu vektor. Vektor yang disusun bisa dalam bentuk kolom ataupun baris. Maksudnya, dari matriks baris dan kolom diubah menjadi matriks baris saja atau matriks kolom saja sehingga setiap gambar hanya punya satu nilai saja. Tahap ini bertujuan untuk mempermudah dalam mencari rata-rata. Selanjutnya, dilakukan proses perhitungan matriks kovarian A. Tahap berikutnya adalah menghitung nilai *eigen*. Setelah itu akan dilakukan pengurutan nilai *eigen* dari yang paling besar ke kecil. Kemudian langkah selanjutnya adalah mencari nilai PCA.

2.8 Weak Textured Patched (WTP)

Weak Texture Patched erat kaitannya dengan algoritma PCA. Penelitian sebelumnya menerapkan teknik PCA dengan menganalisis WTP pada citra untuk memperkirakan tingkat *noise* [10]. Peneliti menggunakan nilai minimum *eigen* yang diambil dari matriks *gradien* kovariansi gambar pada daerah *weak textured patch* dan menganalisis perubahan tingkat *noise* yang berbeda.

2.9 Tools Terkait

Pada bagian ini dibahas *tools* terkait dalam proses mengevaluasi gambar pada Tugas Akhir ini. Terdapat empat *tools* yang dibahas yaitu *MatLab*, *Open CV*, *Python* dan *JTenun*.

2.9.1 Matlab

Pada bagian ini akan dibahas mengenai *tools* terkait dalam proses penentuan evaluasi gambar yang dirancang pada Tugas Akhir ini. Terdapat empat *tools* yang akan dibahas. Salah satunya adalah MATLAB (*Matrix Laboratory*) adalah sistem interaktif yang mempunyai basis data *array* yang tidak membutuhkan dimensi. Pemakaian MATLAB meliputi:

1. Matematika dan komputasi
2. Pembentukan algoritma
3. Akusisi data
4. Pemodelan, simulasi dan pembuatan *prototype*
5. Analisis data, explorasi, dan visualisasi
6. Grafik keilmuan dan bidang rekayasa

Elemen dasar data yang digunakan adalah matriks, sehingga sebuah bilangan bulat dianggap sebagai matriks satu baris dan satu kolom. Sehingga dengan demikian *tools* ini mempermudah penggunaanya untuk melakukan pengolahan gambar. Pengolahan gambar pada tugas akhir yang dilakukan penulis banyak berhubungan dengan matriks dan *vector* dari sebuah gambar tenun.

2.9.2 Open CV

OpenCV atau *Open Source Computer Vision* adalah *library* dari fungsi-fungsi pemrograman untuk *Realtime Computer Vision* (OpenCV, Opencv 2015). *Library* ini dapat digunakan untuk kegiatan akademik ataupun komersial secara gratis. Hingga saat ini sudah terdapat *interface* adalah bahasa C, C++, *Python* dan *Java* (*Android*) pada *library* dan dapat berjalan di *Windows*, *Linux*, *Android* dan *iOS platform*. Terdapat 2500 (dua ribu lima ratus) algoritma yang telah dioptimasi dan digunakan di seluruh dunia.

2.9.3 Python

Tools yang digunakan oleh peneliti pada pengerjaan Tugas Akhir ini yaitu *python*. *Python* adalah bahasa pemrograman interpretatif multiguna. Tidak seperti bahasa lain yang susah untuk dibaca dan dipahami, *python* lebih menekankan pada keterbacaan kode agar lebih mudah untuk memahami sintaks. Hal ini membuat *python* sangat mudah dipelajari baik untuk pemula maupun untuk yang sudah

menguasai bahasa pemrograman lain. Bahasa ini muncul pertama kali pada tahun 1991, dirancang oleh seorang bernama Guido van Rossum. Sampai saat ini *python* masih dikembangkan oleh *Python Software Foundation*. Bahasa *python* mendukung hampir semua sistem operasi, bahkan untuk sistem operasi *linux*, hampir semua distronya sudah menyertakan *python* di dalamnya. Dengan kode yang simpel dan mudah diimplementasikan, seorang *programmer* dapat lebih mengutamakan pengembangan aplikasi yang dibuat.

2.9.4 JTenun

Menurut Saragih et al., (2005: 41-42) [1] JTenun merupakan perangkat lunak yang dapat digunakan untuk menghasilkan motif atau pola tenun yang baru tanpa melupakan informasi

dasar dari pola tenun yang sudah ada sebelumnya. JTenun juga merupakan bagian terbesar dari pengerjaan Tugas Akhir ini. JTenun terdiri dari 4 (empat) modul utama, yaitu *Tenun Editor/Generator*, *Core Learning System*, *Tenun Catalogue*, dan *Data Collecting Interface*.

Berikut penjelasan dari tiap-tiap modul:

1. *Tenun Editor* atau *Generator*

Modul ini merupakan modul utama dimana penenun dan piranti berinteraksi. Fungsi-fungsi dari modul ini adalah sebagai berikut:

- a. Antarmuka tempat pengguna memasukkan *input* dan menerima *output* dari piranti cerdas - *output* utama berupa hasil desain pola tenun yang baru atau hasil translasi desain ke alat tenun.
- b. Sebagai *editor* tempat pengguna/penenun untuk memanipulasi gambar pola tenun.
- c. Sebagai salah satu alat pengoleksi desain tenun yang digunakan pada modul *data collecting interface* untuk memperkaya data latih modul *core learning system*.

2. *Core Learning System*

Modul ini memungkinkan modul *Tenun Editor/Generator* untuk dapat menggenerasi atau mensintesis pola tenun baru dengan memanfaatkan *Machine Learning*.

3. Tenun *Catalogue*

Modul ini digunakan sebagai *viewer* untuk melihat arsip dari data tenun yang telah dikoleksi. *Viewer* ini dirancang dalam bentuk *online catalogue* yang dapat diakses *via web browser*.

4. Data *Collection Interface*

Modul ini merupakan antarmuka untuk pengoleksian data tenun secara masif. Semua data yang dimasukkan melalui modul ini akan disimpan pada *database*. Kumpulan data tersebut digunakan oleh *Core Learning System* untuk menghasilkan tenun *generator* dan juga sebagai sumber informasi bagi tenun *Catalogue*.

2.10 Modul Pendukung

Dalam pengerjaan Tugas Akhir ini, peneliti memerlukan modul dari kelompok Tugas Akhir yang lain untuk melakukan pengklasifikasian dan melakukan perbaikan pada gambar masukan. Berikut penjelasan ringkas mengenai modul yang dimaksud.

2.10.1 Modul Enhancement

Seperti yang telah dijelaskan pada sub bab sebelumnya pada latar belakang pengerjaan Tugas Akhir ini, gambar yang ditangkap secara langsung tidak selalu menghasilkan kualitas gambar yang baik. Maka dari itu, gambar masukan yang kualitasnya kurang baik dikarenakan mengalami gangguan perlu untuk diperbaiki agar dapat dengan mudah diinterpretasikan baik oleh manusia atau mesin. Perbaikan gambar atau *image enhancement* merupakan proses memperbaiki kualitas gambar untuk memperoleh gambar yang lebih berkualitas dari gambar sebelum diolah oleh komputer. Proses yang termasuk dalam *image enhancement* yang dimaksud yaitu perubahan kecerahan gambar (*contrast enhancement*), pengurangan derau (*noise reduction*), *noise smoothing*, dan penajaman gambar (*image sharpening*) [8]. Dari Tugas Akhir ini peneliti menggunakan *noise reduction* dan *image sharpening*. Modul *image sharpening* akan digunakan dalam meningkatkan kualitas gambar masukan dengan cara meningkatkan nilai *blur* pada gambar masukan. Sementara itu, modul *noise reduction* akan digunakan dalam

meningkatkan nilai *noise* gambar dengan parameter *noise* dengan tujuan untuk meningkatkan kualitas gambar.

2.10.2 Modul Classification

Modul *classification* ini menggunakan pendekatan *supervised learning*. Proses *classification* ini bertujuan untuk mengelompokkan gambar Ulos berdasarkan jenis Ulos. Klasifikasi atau pengelompokan gambar Ulos dilakukan dengan menggunakan algoritma *Convolutional Neural Networks* (CNN). Berdasarkan penelitian implementasi *image classification* menggunakan algoritma CNN diketahui bahwa rata-rata akurasi yang didapatkan diatas 85% dalam melakukan proses klasifikasi terhadap gambar. Oleh karena itu, penerapan algoritma CNN memiliki peluang untuk diimplementasikan pada Ulos *classification* [21]. Data yang digunakan untuk *training* dan *test* pada model yang akan dihasilkan yaitu data gambar Ulos Batak Toba. Data gambar ulos Batak Toba tersebut terbagi ke dalam 8 (delapan) kategori seperti Ulos Sadum, Ulos Ragi Hidup, Ulos Harungguan, Ulos Bintang Maratur, Ulos Sibolang, Ulos Sitolutuho, Ulos Mangiring, dan Ulos Ragi Hotang.

2.11 Penelitian Terkait

Pada bagian ini dibahas mengenai penelitian terkait yang berhubungan dengan proses evaluasi gambar dalam pengerjaan Tugas Akhir. Penelitian lain terkait tentang Tugas Akhir ini selain yang dituliskan pada Tugas Akhir tahun lalu dapat dilihat pada Tabel 1.

Tabel 1. Ringkasan Penelitian Terkait

No.	Objek Penelitian dan Peneliti	Keterangan Singkat Penelitian yang dilakukan
1.	Xinhao Liu, Masayuki Tanaka, Masatoshi Okutomi dari Department of Mechanical and Control Engineering, Tokyo Institute of Technology dengan judul penelitian “ Noise Level Estimation Using Weak Textured Patches Of A Single Noisy Image “[22]	Tujuan dari penelitiannya adalah untuk mengetahui estimasi <i>noise</i> dari suatu gambar yang rusak oleh <i>Gaussian</i> dengan menggunakan Algoritma <i>Principle Component Analysis</i> untuk memperkirakan tingkat <i>noise</i> berdasarkan daerah <i>weak textured patch</i> .

No.	Objek Penelitian dan Peneliti	Keterangan Singkat Penelitian yang dilakukan
2.	Suta Wijaya, Hendri, Gasim dari Jurusan Teknik Informatika, STMIK GI MDP, Palembang dengan judul penelitian “Penerapan Algoritma Principle Component Analysis (PCA) Dan Fitur RGB Untuk Pelacakan Jenis Dan Warna Buah” [20]	Tujuan dari penelitiannya adalah untuk melacak jenis buah dan warna buah dengan menggunakan Algoritma <i>Principle Component Analysis</i> . Berdasarkan hasil perhitungan, algoritma PCA dapat mengenali jenis dengan tingkat akurasi kemiripan jenis buah sebesar 86.7 % dari citra yang diuji.
3.	Ahmad Saikhu, Nanik Suciati, Widhiantantri S dari Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember dengan judul penelitian “Perbaikan Citra Ber-Noise Menggunakan Switching Median Filter Dan Boundary Discriminative Noise Detection” [12]	Tujuan dari penelitiannya adalah untuk mendeteksi sekaligus memperbaiki <i>noise</i> dengan algoritma BDND. Algoritma <i>Boundary Discriminative Noise Detection</i> sangat tangguh dalam mendeteksi <i>noise</i> . Karena dari uji coba yang telah dilakukan <i>miss detection</i> mencapai nilai 0 untuk <i>noise density</i> mencapai 90%.
4.	Renting Liu, Zhaorong Li, Jiaya Jia Department of Computer Science and Engineering. The Chinese University of Hong Kong dengan judul penelitian “Image Partial Blur Detection and Classification” [16]	Tujuan dari penelitiannya adalah mengusulkan deteksi gambar parsial-buram dan kerangka analisis untuk klasifikasi secara otomatis apakah satu gambar berisi wilayah yang buram dan apa jenis <i>blur</i> terjadi tanpa perlu melakukan gambar <i>deblurring</i> .
5.	Jianping Shi, Li Xu, Jiaya Jia The Chinese University of HongKong. Image & Visual Computing Lab, Lenovo R&T dengan judul penelitian “Discriminative Blur Detection Features” [10]	Tujuan dari penelitiannya adalah untuk memaparkan pembeda fitur deteksi <i>blur</i> . Ada beberapa fitur <i>blur</i> lokal yang efektif. Penelitiannya menggambarkan sifat <i>blur</i> yang berbeda dan terintegrasi menjadi kerangka inferensi multi-skala untuk menangani variasi skala.

2.12 Kesimpulan Tinjauan Pustaka

Berdasarkan hasil analisis dari penelitian-penelitian sebelumnya serta studi literatur yang berkaitan dengan topik Tugas Akhir, dapat disimpulkan peneliti akan menggunakan 2 (dua) parameter dalam menentukan kualitas gambar Ulos, yaitu dengan menggunakan parameter *blur* dan *noise* karena kedua parameter ini merupakan bagian dari parameter penentuan kualitas gambar. Kedua parameter ini akan diuji dengan menggunakan dua algoritma yaitu FFT (*fast fourier transform*) dan PCA (*principal component analysis*).

3. Analisis

Pada bab ini diuraikan mengenai proses analisis kedua parameter sebagai salah satu cara penentuan kualitas gambar masukan pada aplikasi JTenun serta metode yang akan digunakan untuk mendeteksi kedua parameter tersebut.

3.1 Tahapan Analisis

Pada tahapan analisis, dilakukan berbagai pengamatan dan analisis terhadap hasil studi *literature* dan eksplorasi. Tahapan analisis yang dilakukan adalah sebagai berikut.

1. Analisis algoritma FFT dan *Laplacian Operator* dalam mendeteksi *blur*.
2. Analisis algoritma PCA dalam mendeteksi *noise*.
3. Analisis kaitan *blur* dan *noise* dengan karakteristik Ulos.
4. Kesimpulan analisis.

3.2 Analisis Algoritma Fast Fourier Transform dan Laplacian Operator

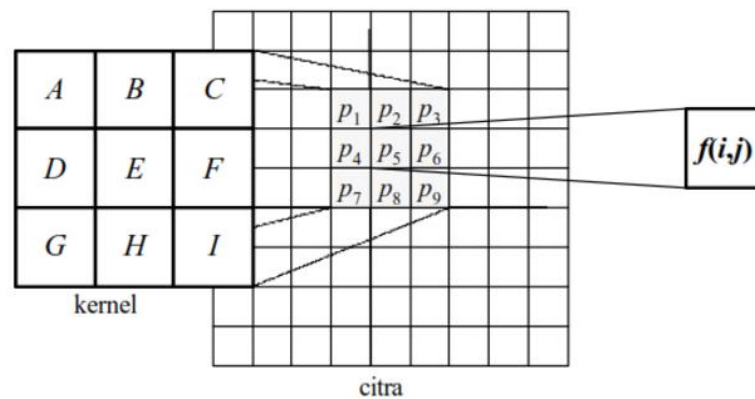
Parameter *blur* akan dihitung dengan menggunakan dua algoritma yang memiliki peranan berbeda yaitu, algoritma *Fast Fourier Transform* dan *laplacian operator*. Awalnya *laplacian operator* yang merupakan operator turunan kedua akan melakukan pendeteksian tepi dengan tujuan untuk memperlihatkan detail gambar. Berikut alur dari pendeteksian yang dilakukan oleh *laplacian* pada gambar masukan:

1. *Input* gambar ke dalam program kemudian gambar atau citra masukan akan diubah dan dibaca dalam bentuk matriks. Pembacaan ini dilakukan berdasarkan warna piksel pada gambar masukan.
2. Selanjutnya, citra masukan akan diubah kedalam gambar berskala keabuan (*grayscale*).
3. Ketika suatu gambar masukan diubah kedalam bentuk matriks, maka terdapat ukuran matriks gambar yang berbeda-beda untuk setiap gambar. Untuk gambar dengan warna RGB maka bentuk matriks akan diubah dalam bentuk matriks biasa (piksel). Setelah itu matriks yang dihasilkan oleh gambar akan dikonvolusi dengan *kernel* 3 x 3, seperti matriks di Gambar 6:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Gambar 6 kernel 3x3

4. Dalam ranah diskrit *kernel*, konvolusi dinyatakan dalam bentuk matriks (umumnya 3 x 3, namun ada juga yang berukuran 2 x 2 atau 2 x 1 atau 1 x 2). Ukuran matriks ini biasanya lebih kecil dari ukuran citra masukan. Setiap elemen matriks disebut koefisien konvolusi.
5. Berikut ilustrasi konvolusi ditunjukkan pada Gambar 7 di bawah ini:



Gambar 7 Ilustrasi kernel

6. Operasi konvolusi dilakukan dengan menggeser *kernel* konvolusi piksel per piksel. Hasil konvolusi disimpan di dalam matriks yang baru. Namun piksel-piksel pinggir diabaikan yang artinya tidak dilakukan konvolusi, maka piksel-piksel pinggir nilainya tetap sama seperti citra asal. Konvolusi dalam pengolahan citra dapat dilakukan untuk banyak hal. Pada Tugas Akhir ini, peneliti mengkonvolusi citra masukan untuk mendeteksi tepi dalam citra.
7. Kemudian, proses ini akan dilanjutkan dengan perhitungan FFT. Pada FFT akan dilakukan perhitungan varians dan standar deviasi kuadrat. Standar deviasi adalah nilai statistik yang dimanfaatkan untuk menentukan bagaimana sebaran data pada citra masukan.

8. Untuk menghitung nilai varians, pertama-tama harus diketahui nilai rata-rata dari data yang ada. Berikut diberikan contoh matriks gambar masukan pada Gambar 8:

	0	1	2	3	4
3	180	179	182	191	192
4	177	174	171	177	183
5	177	169	167	164	165
6	173	169	175	172	162
7	169	174	175	165	159
8	173	178	177	179	175
9	178	179	182	180	177
10	181	179	176	177	175
11	181	186	184	179	170
12	172	165	168	167	156
13	168	157	156	157	145
14	160	158	154	151	147

Gambar 8 Contoh matriks gambar masukan

Matriks pada Gambar 8 dengan ukuran 700 x 533 dan dimulai dari indeks 0. Dari matriks inilah akan dilakukan perhitungan varians dan standar deviasi. Sebelum itu, harus diketahui jumlah rata-rata pada keseluruhan matriks. Setelah rata-rata pada matriks diketahui maka akan dihitung variansnya. Rumus varians dapat dilihat pada Gambar 9:

$$= \frac{\sum (x - \bar{x})^2}{n - 1}$$

Gambar 9 Rumus Varians

Keterangan:

X = nilai matriks per piksel

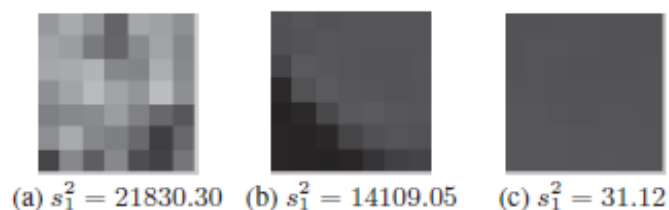
\bar{X} = nilai rata-rata piksel

n = jumlah data

Perhitungan varians pada gambar masukan akan menggunakan rumus varians seperti pada Gambar 9, dimana perhitungan ini didasarkan pada matriks keluaran. Sedangkan perhitungan standar deviasi adalah nilai kuadrat dari hasil perhitungan varians. Penelitian ini memiliki asumsi bahwa sebuah citra masukan yang memiliki jumlah persebaran data yang rendah pada titik-titik frekuensi yang luas maka akan disebut dengan daerah *high frequency* yang hasilnya adalah gambar *blur*, begitu juga sebaliknya. Citra masukan dengan persebaran data yang kecil pada daerah *high frequency* akan dianggap sebuah gambar yang *blur*. Semakin tinggi nilai *output* dari *laplacian* maka gambar akan dianggap sebagai gambar yang tidak *blur* dan sebaliknya semakin rendah nilai *output* maka gambar akan dianggap *blur*.

3.3 Analisis Algoritma PCA

Analisis Algoritma PCA (Komponen Utama) banyak digunakan untuk memproyeksikan atau mengubah suatu kumpulan data berukuran besar menjadi bentuk sajian data dengan ukuran yang lebih kecil. PCA digunakan menghitung nilai *noise* pada suatu gambar dari pemilihan WTP (*weak textured patch*). Zhu dan Milanfar mendemostrasikan bahwa struktur dari gambar dapat diukur dengan efektif melalui matriks kovarian [31]. Nilai *noise* pada gambar dapat diukur apabila *weak texture patch* pada gambar dipilih terlebih dahulu. *Patch* pada gambar akan direpresentasikan sebagai matriks yang merepresentasikan operator turunan *vertical* dan *horizontal*. Dalam pemilihan *patch* ada asumsi bahwa *weak textured patch* merupakan “*patch* atau potongan gambar yang memiliki tekstur relatif *smooth* (lembut)”.



Gambar 10 Patches dengan nilai eigen

Dapat dilihat dari Gambar 10, bahwa *weak textured patches* adalah gambar yang paling kanan karena memiliki nilai *eigen* yang paling kecil. Jika berbicara mengenai *threshold*, maka *threshold* akan diberikan secara manual oleh *programmer*

misalnya, 0.99. Berikut penjelasan mengenai pemilihan *patch* dan penghitungan nilai *noise* dari gambar:

1. Sebuah gambar masukan akan diproses untuk perhitungan nilai *noise*. Gambar Ulos yang sudah dimasukkan akan diuji coba dengan nilai *threshold* pada rentang keabuan antara 0-255 (*grayscale*).
2. Dilanjutkan dengan segmentasi ukuran citra menjadi beberapa bagian.
3. Selanjutnya, proses ekstraksi ciri dilakukan setelah mendapatkan citra *grayscale*.
4. Kemudian akan dilakukan proses perhitungan nilai *noise* dari matriks kovarian, dimana telah dihasilkan oleh semua *patch* pada gambar masukan. Tahapan berikutnya adalah menghitung nilai *eigen* (E) dari matriks kovarian, setelah itu dilakukan pengurutan nilai *eigen* kemudian dicari *patches* dengan nilai *eigen* paling kecil. Tahap ini dilakukan sebanyak tiga kali iterasi agar didapatkan nilai *noise* yang stabil, namun hal ini bisa ditentukan jumlahnya pada parameter fungsi. Tahap terakhir identifikasi *noise* pada gambar didapat dari gambar masukan. Untuk penghitungan nilai *eigen* melibatkan perhitungan aljabar linear, dapat dilihat dari penelitian Xinhao Liu, 2012 [22].

3.4 Analisis kaitan parameter blur dan noise dengan karakteristik motif Ulos

Beberapa motif Ulos memiliki karakteristik atau corak yang timbul seperti Ulos Sadum. Hal ini menyebabkan corak pada gambar Ulos terdeteksi sebagai *noise* dan akan mempengaruhi program pendeteksi *noise* yang akan meningkatkan nilai *noise* pada gambar.

3.5 Kesimpulan Analisis

Berdasarkan hasil analisis yang dilakukan, terdapat beberapa hal yang menjadi kesimpulan dan pertimbangan dalam tahap implementasi yang akan dilakukan:

1. Proses evaluasi gambar akan dilakukan dengan tiga eksperimen yaitu memeriksa dan menghitung nilai dari 1 (satu) parameter terlebih dahulu yaitu evaluasi gambar untuk parameter *blur* dan evaluasi gambar untuk parameter *noise*, dan yang ketiga dilakukan secara sekuensial yaitu sebagai berikut, pertama dilakukan pemeriksaan dan penghitungan nilai dari parameter *blur* kemudian sebagian hasil dari evaluasi *blur* akan dilakukan pengecekan parameter *noise*.
2. Parameter *blur* dideteksi dengan algoritma *laplacian operator* dan FFT.
3. Parameter *noise* dideteksi dan dihitung dengan algoritma PCA (*Principal Component Analysis*).

4. Desain Eksperimen

Pada bab ini dijelaskan mengenai desain eksperimen yang telah diimplementasikan pada berbagai Ulos sesuai dengan metode yang telah dijelaskan pada bab sebelumnya serta menjelaskan desain detail untuk setiap percobaan yang dilakukan.

4.1 Objek Eksperimen

Pada eksperimen ini, objek yang menjadi kasus pengujian adalah gambar Ulos yang telah dikumpulkan oleh tim peneliti. Ulos yang digunakan sebanyak 8 (delapan) jenis, sebagai berikut:

1. Bintang Maratur
2. Harurungan
3. Mangiring
4. Ragihidup
5. Ragihotang
6. Sadum
7. Sibolang
8. Sitoluntuho

Keseluruhan gambar Ulos yang digunakan dalam eksperimen ini sebanyak 541 (lima ratus empat puluh satu) gambar Ulos. Beberapa gambar yang digunakan sebagai data masukan ini mengalami perulangan sehingga menyebabkan 1 (satu) Ulos dapat memiliki lebih dari 1 (satu) gambar Ulos yang ditangkap. Peneliti mendapatkan gambar Ulos yang akan digunakan sebagai data masukan eksperimen dengan cara mengambil foto Ulos milik masyarakat yang berada di daerah Tobasa, milik dosen dan mahasiswa IT Del, dan juga dengan cara mengunjungi pabrik penghasil maupun toko penjual Ulos di daerah Tobasa. Adapun teknik pengambilan gambar Ulos yang dilakukan peneliti ialah dengan Ulos yang dibentangkan secara lurus pada permukaan datar misalnya meja, lantai, dan lain sebagainya kemudian peneliti mengambil gambar seluruh Ulos dari atas dengan *angle* (arah pengambilan) maupun *smartphone* yang berbeda-beda untuk memperoleh keragaman piksel pada gambar ulos.

4.2 Desain Eksperimen

Eksperimen ini dirancang untuk melakukan pengevaluasian terhadap gambar Ulos yang dijadikan sebagai data masukan. Menurut Calongesi (1995), evaluasi adalah suatu keputusan tentang nilai berdasarkan hasil pengukuran yang dilakukan [5]. Berdasarkan tujuannya, terdapat pengertian evaluasi sumatif dan evaluasi formatif. Evaluasi formatif dinyatakan sebagai upaya untuk memperoleh *feedback* perbaikan program, sementara evaluasi sumatif merupakan upaya menilai manfaat program dan mengambil keputusan [11]. Proses Evaluasi Gambar ini masuk ke evaluasi sumatif. Pada sub bab ini akan dijelaskan secara rinci mengenai tahapan ketiga eksperimen yang telah diputuskan oleh peneliti untuk dilaksanakan. Eksperimen ini dilakukan sebanyak tiga kali karena peneliti juga ingin mengamati bahwa eksperimen manakah yang lebih akurat untuk pengklasifikasian kualitas gambar masukan. Adapun ketiga eksperimen tersebut adalah:

1. Evaluasi parameter *blur*

Mendeteksi hanya pada parameter *blur* pada gambar Ulos. Ada 3 kategori Ulos pada evaluasi ini yaitu *good*, *improve*, dan *bad*.

2. Evaluasi parameter *noise*

Mendeteksi hanya pada parameter *noise* pada gambar Ulos. Ada 2 kategori Ulos pada evaluasi ini yaitu *good* dan *improve*.

3. Evaluasi *blur* dan *noise*

Mendeteksi secara sekuensial antara *blur* dengan *noise*. Pertama-tama, gambar akan dievaluasi melalui parameter *blur* yang memiliki 3 kategori, kemudian gambar Ulos yang kategori *good* dan *improve* akan dievaluasi melalui parameter kedua, *noise*.

Berikut penjelasan untuk 3 (tiga) kategori yang ditetapkan oleh peneliti:

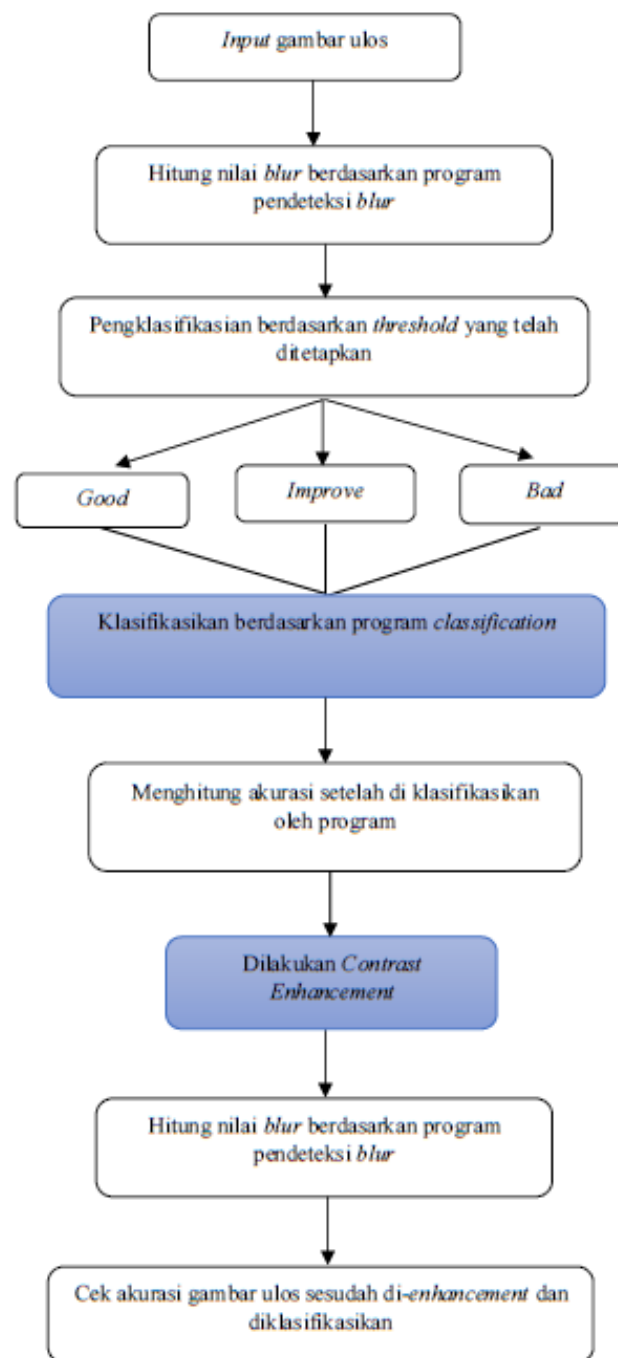
1. Kategori *bad* akan secara langsung ditolak oleh aplikasi karena jika gambar Ulos di-*improve* maka kemungkinan tetap berada di dalam kategori *bad* besar,
2. Kategori *improve* akan di-*enhancement* dengan *contrast enhancement* agar dapat menuju proses selanjutnya yaitu pengecekan *noise* dan gambar,
3. Sedangkan, Ulos yang kategori *good* akan secara langsung dilakukan pengecekan *noise*.

Adapun hal hal yang telah dipersiapkan untuk ekperimen ini adalah:

1. Program pendeteksi *blur* pada IDE *spyder*.
2. Program pendeteksi *noise* pada IDE MATLAB.
3. Gambar Ulos yang ditangkap dari *smartphone* sebanyak 541 (lima ratus empat puluh satu) gambar.
4. Program *enhancement* yaitu *contrast enhancement* dan *denoising*.
5. Program *classification* Ulos.

4.2.1 Parameter Blur

Pada sub bab ini dibahas terkait pengamatan evaluasi gambar Ulos untuk parameter *blur*. Pengamatan evaluasi tersebut berupa pemahaman terkait proses untuk mengetahui dan semakin memastikan rentang *threshold* pada gambar agar dapat menentukan klasifikasinya. Proses analisis parameter *blur* pada gambar Ulos yang akan dilakukan oleh peneliti akan tampak seperti Gambar 11 di bawah ini.



Gambar 11 Proses evaluasi gambar Ulos berdasarkan parameter blur

Pada Gambar 11 ditampilkan tahapan evaluasi gambar Ulos berdasarkan parameter *blur*. Berikut tahapan yang dilakukan pada proses ini:

1. *Input* keseluruhan gambar Ulos yang diambil dari *smartphone* ke dalam IDE *spyder* yang telah berisi program pendeteksi *blur*.

2. Jalankan program pendeteksi *blur* yang telah dimodifikasi oleh peneliti, maka program ini akan mengeluarkan nilai *blur* pada saat dijalankan untuk setiap gambar Ulos.
3. Selanjutnya, gambar Ulos yang telah memiliki nilai akan diklasifikasikan ke dalam tiga klasifikasi yaitu *good*, *improve*, dan *bad*. Peneliti yang akan menentukan *threshold* pada tiap klasifikasinya dengan mengamati nilai *output* gambar Ulos secara manual. Pengklasifikasian manual yang dimaksud ialah dengan mengamati secara kasat mata gambar masukan Ulos dan menilainya secara subyektif.
4. Langkah selanjutnya adalah memasukkan keseluruhan gambar ulos ke dalam program *classification* dan gambar ulos akan diklasifikasikan berdasarkan jenis. Misalnya, Ulos sadum, Ulos mangiring, Ulos harungguan, dan lain sebagainya.
5. Setelah pengklasifikasian selesai maka peneliti akan menghitung akurasi gambar masukan tiap kategori. Perhitungan akurasi ini dilakukan per kategori, berikut pada persamaan (i), (ii), dan (iii) akan dijelaskan rumus untuk perhitungan ini:
 - (i)
$$\text{Akurasi data bad} = \frac{\text{jumlah gambar ulos yang klasifikasinya salah}}{\text{total gambar ulos keseluruhan pada kategori bad}} \times 100\%$$
 - (ii)
$$\text{Akurasi data improve} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$
 - (iii)
$$\text{Akurasi data good} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$
6. Untuk meningkatkan keakuratan dalam memperoleh *threshold* yang telah ditetapkan, peneliti pun akan memasukkan keseluruhan gambar masukan ke dalam program *enhancement* yaitu *contrast enhancement*. Program ini bertujuan untuk meningkatkan nilai *blur* pada gambar Ulos.
7. Setelah dilakukan *contrast enhancement*, maka gambar harus diklasifikasikan ulang berdasarkan nilai yang baru diperoleh.

8. Setelah kedua proses ini selesai, peneliti akan menghitung akurasi gambar masukan tiap kategori setelah dilakukan *enhancement*. Perhitungan akurasi ini dilakukan per kategori, berikut pada persamaan (i), (ii), dan (iii) akan dijelaskan rumus untuk perhitungan ini:

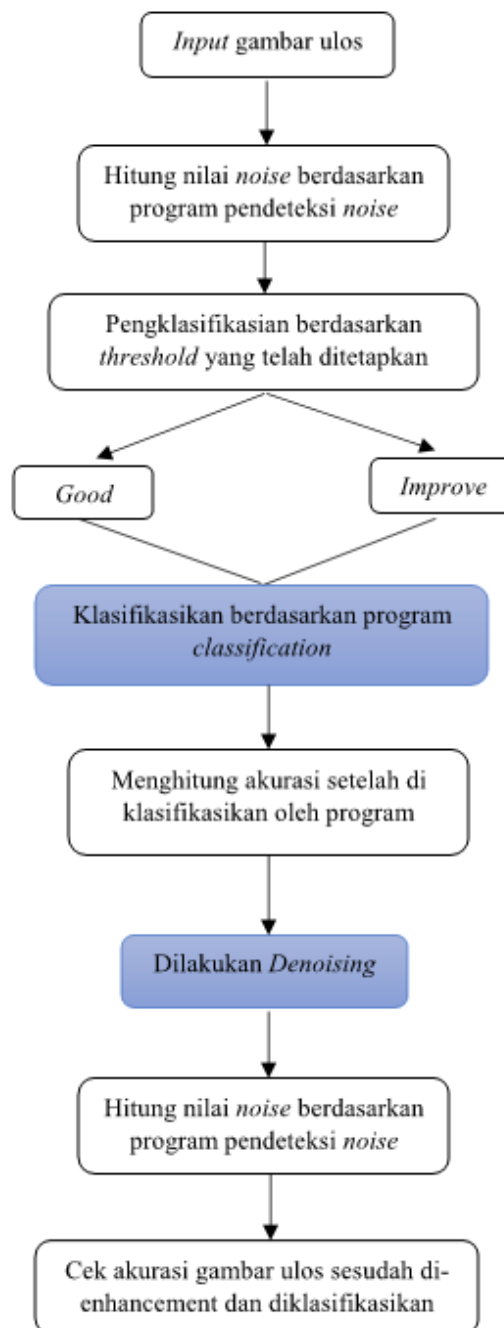
$$(i) \text{Akurasi data bad} = \frac{\text{jumlah gambar ulos yang klasifikasinya salah}}{\text{total gambar ulos keseluruhan pada kategori bad}} \times 100\%$$

$$(ii) \text{Akurasi data improve} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$

$$(iii) \text{Akurasi data good} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$

4.2.2 Parameter Noise

Pada sub bab ini dibahas terkait pengamatan evaluasi gambar untuk parameter *noise*. Pengamatan tersebut berupa pemahaman terkait proses mengetahui *threshold* suatu gambar dinyatakan memiliki *noise*. Pada Gambar 12 ditampilkan tahapan evaluasi gambar Ulos berdasarkan parameter *noise*.



Gambar 12 Proses evaluasi gambar Ulos berdasarkan parameter noise

Berikut penjelasan Gambar 12 mengenai proses evaluasi gambar berdasarkan parameter *noise*:

1. *Input* gambar Ulos yang diambil dari *smartphone* ke dalam IDE MATLAB yang telah berisi program pendeteksi *noise*.

2. Jalankan program pendeteksi *noise* yang telah dimodifikasi oleh peneliti, maka program ini akan mengeluarkan nilai *noise* sesuai dengan gambar yang telah dimasukkan.
3. Selanjutnya, gambar Ulos yang telah memiliki nilai akan diklasifikasikan ke dalam dua klasifikasi yaitu *good* dan *improve*. Peneliti yang akan menentukan *threshold* dengan mengamati gambar Ulos serta nilai *output* secara manual.
4. Langkah selanjutnya adalah memasukkan keseluruhan gambar ulos ke dalam program *classification* dan gambar ulos akan diklasifikasikan berdasarkan jenis. Misalnya, Ulos sadum, Ulos mangiring, Ulos harungguan, dan lain sebagainya.
5. Setelah pengklasifikasian selesai maka peneliti akan menghitung akurasi gambar masukan tiap kategori. Perhitungan akurasi ini dilakukan per kategori, berikut pada persamaan (i), (ii), dan (iii) akan dijelaskan rumus untuk perhitungan ini:

$$(i) \quad \text{Akurasi data bad} = \frac{\text{jumlah gambar ulos yang klasifikasinya salah}}{\text{total gambar ulos keseluruhan pada kategori bad}} \times 100\%$$

$$(ii) \quad \text{Akurasi data improve} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$

$$(iii) \quad \text{Akurasi data good} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$

6. Untuk meningkatkan keakuratan dalam memperoleh *threshold* yang telah ditetapkan, peneliti pun akan memasukkan keseluruhan gambar masukan ke dalam program *enhancement* yaitu *contrast enhancement*. Program ini bertujuan untuk meningkatkan nilai *blur* pada gambar Ulos.
7. Setelah dilakukan *contrast enhancement*, maka gambar harus diklasifikasikan ulang berdasarkan nilai yang baru diperoleh.
8. Setelah kedua proses ini selesai, peneliti akan menghitung akurasi gambar masukan tiap kategori setelah dilakukan *enhancement*. Perhitungan akurasi ini dilakukan per kategori, berikut pada persamaan (i), (ii), dan (iii) akan dijelaskan rumus untuk perhitungan ini:

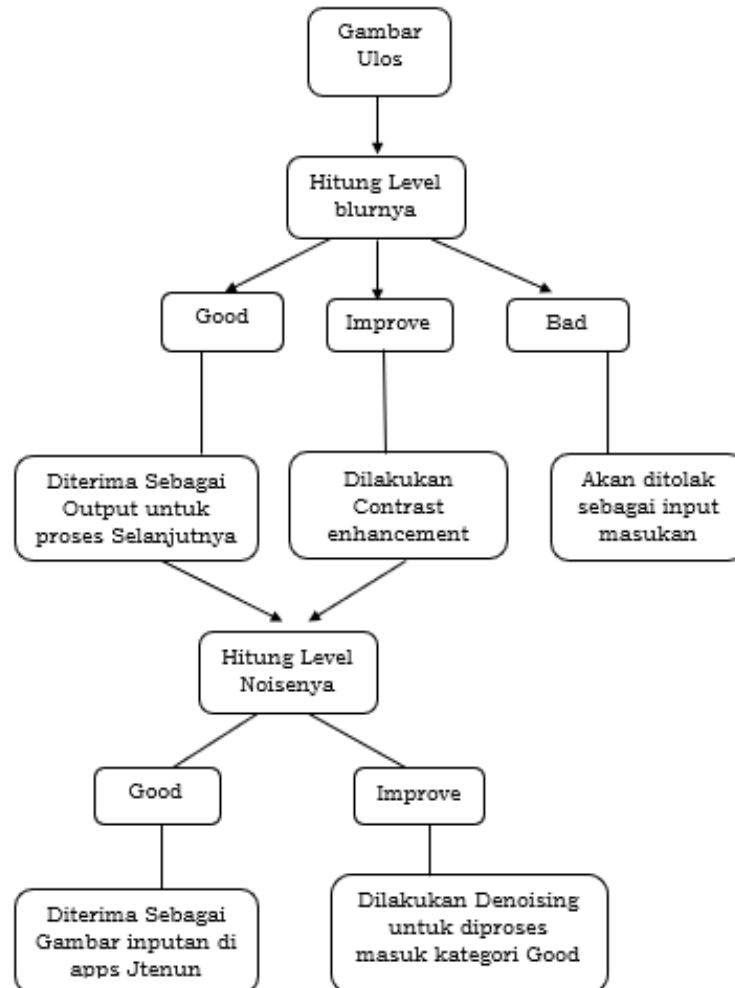
$$(i) \quad \text{Akurasi data bad} = \frac{\text{jumlah gambar ulos yang klasifikasinya salah}}{\text{total gambar ulos keseluruhan pada kategori bad}} \times 100\%$$

$$(ii) \quad \text{Akurasi data improve} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$

$$(iii) \quad \text{Akurasi data good} = \frac{\text{jumlah gambar ulos yang klasifikasinya benar}}{\text{total gambar ulos keseluruhan pada kategori improve}} \times 100\%$$

4.2.3 Parameter Blur dan Noise

Pada sub bab ini dibahas terkait pengamatan evaluasi gambar untuk parameter *blur* dan *noise*. Pengamatan tersebut berupa pemahaman terkait proses mengetahui *threshold* suatu gambar dinyatakan memiliki *blur* dan *noise*. Pada Gambar 13 ditampilkan tahapan evaluasi gambar Ulos berdasarkan parameter *blur* dan *noise*:



Gambar 13 Proses Evaluasi Gambar blur dan noise

Berdasarkan Gambar 13, proses evaluasi gambar akan dimulai dengan penghitungan nilai *blur* pada semua gambar Ulos berdasarkan program pendeteksi *blur*, kemudian gambar akan dikategorikan berdasarkan *threshold* yang telah ditetapkan oleh peneliti. Kategori yang dimaksud yaitu, *bad*, *improve*, dan *good*. Berikut penjelasan untuk 3 (tiga) kategori yang ditetapkan oleh peneliti:

1. Kategori *bad* akan secara langsung ditolak oleh aplikasi karena jika gambar Ulos di-*improve* maka kemungkinan tetap berada di dalam kategori *bad*-nya besar,

2. Kategori *improve* akan di *enhancement* dengan *contrast enhancement* agar dapat menuju proses selanjutnya yaitu pengecekan *noise* dan gambar
3. Sedangkan, Ulos yang kategori *good* akan secara langsung dilakukan pengecekan *noise*.

Untuk pengecekan parameter kedua yaitu *noise* hanya akan ada dua kategori yaitu *good* dan *improve*. Berikut penjelasannya:

1. Untuk kategori *good* akan secara langsung diterima oleh aplikasi untuk dilanjut ke proses selanjutnya.
2. Sedangkan kategori *improve* akan dikirim menuju proses *enhancement* yaitu, *noise reduction*. Kemudian akan diterima oleh aplikasi untuk dilanjut ke proses selanjutnya.

Proses evaluasi gambar ini menjadi penentuan kualitas terkait gambar masukan yang akan diproses untuk tahap selanjutnya.

4.3 Tools/perangkat yang digunakan untuk eksperimen

Berikut dijelaskan terkait *tools*/perangkat yang digunakan saat melakukan eksperimen yaitu sebuah *personal computer* dan laptop dengan spesifikasi dapat dilihat pada tabel di sub bab ini.

4.3.1 Hardware

Berikut dijelaskan perangkat keras yang digunakan peneliti dalam melakukan uji coba eksperimen adalah sebuah *personal computer* dan laptop dengan spesifikasi dapat dilihat pada Tabel 2 berikut.

Tabel 2. Spesifikasi Hardware

<i>Operating System (OS)</i>	<i>Windows</i>
Windows Edition	Windows 8
Processor	Intel ® Core™ i7-6700 CPU @3.40GHz 3.40 GHz
Installed Memory (RAM)	8.00 GB
System Type	64-bit Operating System
Windows Edition	Windows 10 Education
Processor	Intel ® Core™ i5-4200 CPU @2.50GHz 2.50 GHz
Installed Memory (RAM)	8.00 GB
System Type	64-bit Operating System

4.3.2 Software

Perangkat Lunak (*Software*) yang digunakan dalam pelaksanaan eksperimen evaluasi gambar dapat dilihat pada Tabel 2 berikut.

IDE	Edition
Phyton 3.6	Spyder 3.2.6, Spyder 3.2.8, Spyder 3.0.0
Matlab R2016b	R2016b (9.1.0,441655) 64 – bit (win 64)
OpenCV	OpenCV 3.4.1 – bit (win 64)

5. Hasil Eksperimen dan Pembahasan

Pada Sub bab ini dijelaskan hasil eksperimen yang telah didapatkan dari eksperimen yang ada dalam evaluasi gambar dengan metode yang telah ditetapkan.




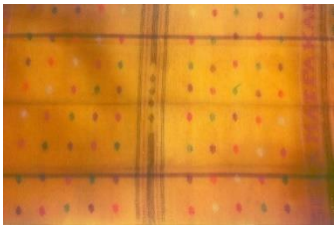

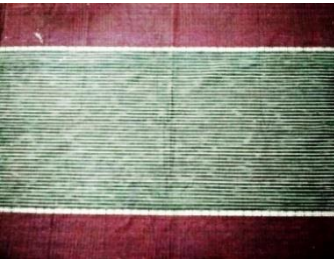
5.1 Hasil Eksperimen Parameter Blur





Pada sub bab ini berisi tentang hasil eksperimen yang dicapai dalam melakukan proses evaluasi gambar Ulos berdasarkan satu parameter yaitu *blur*. Berikut beberapa sampel gambar yang telah didapatkan nilai serta klasifikasinya:

1. Kategori *bad*

Tabel 4 berisi gambar Ulos kategori *bad* sebelum dan sesudah *enhancement* disertai dengan nilai *blur* dan klasifikasi masing-masing.

Tabel 3. Sampel gambar Ulos kategori bad berdasarkan parameter blur

Gambar Ulos sebelum Enhancement	Hasil Klasifikasi	Gambar Ulos sesudah Enhancement	Hasil Klasifikasi
 <p>Nilai blur: 147.79</p>	Salah	 <p>Nilai blur: 3206.74</p>	Salah
 <p>Nilai blur: 37.87</p>	Salah	 <p>Nilai blur: 206.42</p>	Salah
 <p>Nilai blur: 167.10</p>	Salah	 <p>Nilai blur: 9137.53</p>	Salah









Gambar Ulos sebelum Enhancement	Hasil Klasifikasi	Gambar Ulos sesudah Enhancement	Hasil Klasifikasi
 Nilai blur: 365.62	Salah	 Nilai blur: 2922.14	Salah
 Nilai blur: 121.55	Salah	 Nilai blur: 6888.18	Salah

Tabel 4 berisi beberapa sampel gambar ulos berdasarkan parameter *blur*. Pada tabel ini, peneliti memberikan contoh gambar ulos yang memiliki nilai *blur* rendah dan terkategori *bad* oleh peneliti. Hal ini dapat dilihat dari nilai yang disediakan tepat berada di bawah gambar ulos. Seperti yang telah dijelaskan pada bab sebelumnya, gambar masukan Ulos akan diklasifikasikan berdasarkan jenis motif ulos seperti ulos sadum, mangiring dan lain sebagainya menggunakan program *classification* yang dikerjakan oleh kelompok Tugas Akhir lain baik sebelum maupun sesudah *contrast enhancement*. Peneliti juga menyediakan hasil klasifikasi pada saat diklasifikasikan, pada kategori ini gambar masukan Ulos memiliki hasil klasifikasi yang sebagian besar salah.

2. Kategori *improve*

Tabel 5 berisi gambar Ulos kategori *improve* sebelum dan sesudah *enhancement* serta sudah memiliki nilai *blur* dan klasifikasi masing-masing.

Tabel 4. Sampel gambar Ulos kategori improve berdasarkan parameter blur

Gambar Ulos	Hasil Klasifikasi	Gambar Ulos setelah Enhancement	Hasil Klasifikasi
 Nilai blur: 1496.39	Benar	 Nilai blur: 2098.09	Benar
 Nilai blur: 485.24	Benar	 Nilai blur: 6247.23	Benar
 Nilai blur: 3802.60	Benar	 Nilai blur: 12138.96	Benar
 Nilai blur: 2769.14	Salah	 Nilai blur: 3077.29	Salah


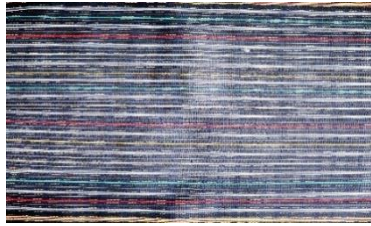






Tabel 5 berisi beberapa sampel gambar ulos berdasarkan parameter *blur*. Pada tabel ini, peneliti memberikan contoh gambar ulos yang terkategori *improve* oleh peneliti. Hal ini dapat dilihat dari nilai yang disediakan tepat berada di bawah gambar ulos. Seperti yang telah dijelaskan pada bab sebelumnya, gambar masukan Ulos akan diklasifikasikan berdasarkan jenis motif ulos seperti ulos sadum, mangiring dan lain sebagainya menggunakan program *classification* yang dikerjakan oleh kelompok Tugas Akhir lain baik sebelum maupun sesudah *contrast enhancement*. Peneliti juga menyediakan hasil klasifikasi pada saat diklasifikasikan, pada kategori ini gambar masukan Ulos memiliki hasil klasifikasi ada yang salah dan ada yang benar.

3. Kategori *good*

Tabel 6 berisi gambar Ulos kategori *good* sebelum dan sesudah *enhancement* serta

sudah memiliki nilai *blur* dan klasifikasi masing-masing.

Tabel 5. Sampel gambar Ulos kategori *good* berdasarkan parameter *blur*

Gambar Ulos sebelum Enhancement	Hasil Klasifikasi	Gambar Ulos setelah Enhancement	Hasil Klasifikasi
 Nilai blur: 4927.27	Benar	 Nilai blur: 15104.54	Benar
 Nilai blur: 8747.33	Benar	 Nilai blur: 16789.09	Benar
 Nilai blur: 2917.82	Benar	 Nilai blur: 11044.04	Benar
 Nilai blur: 18850.24	Benar	 Nilai blur: 14543.00	Benar

Tabel 6 berisi beberapa sampel gambar ulos berdasarkan parameter *blur*. Pada tabel ini, peneliti memberikan contoh gambar ulos yang memiliki nilai *blur* tinggi dan terkategori *good* oleh peneliti. Hal ini dapat dilihat dari nilai yang disediakan tepat berada di bawah gambar ulos. Seperti yang telah dijelaskan pada bab sebelumnya, gambar masukan Ulos akan diklasifikasikan

berdasarkan jenis motif ulos seperti ulos sadum, mangiring dan lain sebagainya menggunakan program *classification* yang dikerjakan oleh kelompok Tugas Akhir lain baik sebelum maupun sesudah *contrast enhancement*. Peneliti juga menyediakan hasil klasifikasi pada saat diklasifikasikan, pada kategori ini gambar masukan Ulos memiliki hasil klasifikasi yang sebagian besar benar. Pada tahap pengujian eksperimen satu parameter yakni hanya parameter *blur* dapat ditarik kesimpulan sebagai berikut:

1. Data *bad* yang telah terklasifikasikan melalui program dihitung tingkat akurasi pada pengklasifikasian gambar Ulos dan memperoleh akurasi sebesar 67.72%
2. Data *Improve* yang terklasifikasikan melalui program dihitung tingkat akurasi pada pengklasifikasian gambar Ulos dan diperoleh akurasi sebesar 16.03%
3. Data *Good* yang terklasifikasikan melalui program dihitung tingkat akurasi pada pengklasifikasian gambar Ulos dan diperoleh akurasi sebesar 26,41%
4. Data Ulos dari kategori Ulos yaitu *Improve* yang telah melalui proses *Contrast Enhancement* dihitung tingkat akurasinya pada pengklasifikasian gambar Ulos diperoleh akurasi sebesar 88.31%
5. Data Ulos dari kategori Ulos yaitu *Improve* yang telah melalui proses *Contrast Enhancement* dihitung tingkat akurasinya pada pengklasifikasian gambar Ulos diperoleh akurasi sebesar 15.38%
6. Data Ulos dari kategori Ulos yaitu *Good* yang telah melalui proses *Contrast Enhancement* dihitung tingkat akurasinya pada pengklasifikasian gambar Ulos diperoleh akurasi sebesar 14.67%







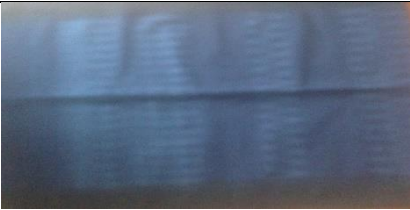

5.2 Hasil Eksperimen Parameter Noise

Pada sub bab ini berisi tentang hasil eksperimen yang dicapai dalam melakukan proses evaluasi gambar Ulos berdasarkan satu parameter yaitu *noise*. Berikut beberapa sampel foto yang telah ditentukan rentang *threshold* oleh peneliti dan didapatkan nilainya.

1. Kategori *improve*

Tabel 7 berisi gambar Ulos kategori *improve* sebelum dan sesudah *enhancement* serta sudah memiliki nilai *noise* dan klasifikasi masing-masing.

Tabel 6. Sampel gambar Ulos kategori *improve* berdasarkan parameter *noise*

Gambar Ulos sebelum Noise Reduction	Hasil Klasifikasi	Gambar Ulos Setelah Noise Reduction	Hasil Klasifikasi
 Nilai noise: 3,443	Salah	 Nilai noise: 0.571	Benar
 Nilai noise: 2,727	Salah	 Nilai noise: 0,125	Benar
 Nilai noise: 1,505	Salah	 Nilai noise: 0,235	Salah
 Nilai noise : 1,428	Salah	 Nilai noise: 0,206	Salah







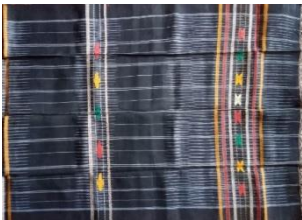

Tabel 7 berisi beberapa sampel gambar ulos berdasarkan parameter *noise*. Pada tabel ini, peneliti memberikan contoh gambar ulos yang memiliki nilai *noise* terkategori *improve* oleh peneliti. Hal ini dapat dilihat dari nilai yang disediakan tepat berada di bawah gambar ulos. Seperti yang telah dijelaskan pada bab sebelumnya, gambar masukan Ulos akan diklasifikasikan berdasarkan jenis motif ulos seperti ulos sadum, mangiring dan lain

sebagainya menggunakan program *classification* yang dikerjakan oleh kelompok Tugas Akhir lain baik sebelum maupun sesudah *denoising*. Peneliti juga menyediakan hasil klasifikasi pada saat diklasifikasikan, pada kategori ini gambar masukan Ulos memiliki hasil klasifikasi yang sebagian besar benar.

2. Kategori *good*

Tabel 8 berisi gambar Ulos kategori *good* sebelum dan sesudah *enhancement* serta sudah memiliki nilai *noise* dan klasifikasi masing-masing.

Tabel 7. Sampel gambar Ulos kategori *good* berdasarkan parameter noise

Gambar Ulos Sebelum Noise Reduction	Hasil Klasifikasi	Gambar Ulos Setelah Noise Reduction	Hasil Klasifikasi
 Nilai noise : 0,274	Benar	 Nilai noise : 0,045	Benar
 Nilai noise : 0,432	Benar	 Nilai noise: 0,040	Benar
 Nilai noise : 0,574	Benar	 Nilai noise: 0,154	Benar
 Nilai noise: 0,789	Benar	 Ulosgood132 : 0,121	Benar

Tabel 8 berisi beberapa sampel gambar ulos berdasarkan parameter *noise*. Pada tabel ini, peneliti memberikan contoh gambar ulos yang memiliki nilai *noise* terkategori *good* oleh peneliti. Hal ini dapat dilihat dari nilai yang disediakan tepat berada di bawah gambar ulos. Seperti yang telah dijelaskan pada bab sebelumnya, gambar masukan Ulos akan diklasifikasikan berdasarkan jenis motif ulos seperti ulos sadum, mangiring dan lain sebagainya menggunakan program *classification* yang dikerjakan oleh kelompok Tugas Akhir lain baik sebelum maupun sesudah *denoising*. Peneliti juga menyediakan hasil klasifikasi pada saat diklasifikasikan, pada kategori ini gambar masukan Ulos memiliki hasil klasifikasi yang sebagian besar benar.

Pada tahap pengujian eksperimen satu parameter yakni hanya parameter *noise* dapat ditarik kesimpulan sebagai berikut:

1. Data *Improve* yang terklasifikasikan melalui program dihitung tingkat akurasi pada pengklasifikasian gambar Ulos dan diperoleh akurasi sebesar 30,79%
2. Data *Good* yang terklasifikasikan melalui program dihitung tingkat akurasi pada pengklasifikasian gambar Ulos dan diperoleh akurasi sebesar 29,68%
3. Data Ulos dari kategori Ulos yaitu *Improve* yang telah melalui proses *denoising* dihitung tingkat akurasinya pada pengklasifikasian gambar Ulos diperoleh akurasi sebesar 14.62%
4. Data Ulos dari kategori Ulos yaitu *Good* yang telah melalui proses *denoising* dihitung tingkat akurasinya pada pengklasifikasian gambar Ulos diperoleh akurasi sebesar 20.67%

5.3 Hasil Eksperimen Parameter Blur dan Noise

Pada eskperimen terakhir yaitu evaluasi dengan menggabungkan antara pengecekan melalui parameter *blur* dan pengecekan parameter *noise* diperoleh tingkat akurasinya sebagai berikut:

1. Data Ulos dari 3 kategori Ulos yaitu *Bad*, *Improve*, dan *Good* yang telah melalui proses pengecekan pertama yaitu, *blur* diperiksa tingkat akurasinya pada pengklasifikasian gambar Ulos kategori *improve* diperoleh akurasi 17.62%

2. Data Ulos dari 3 kategori Ulos yaitu *Bad*, *Improve*, dan *Good* yang telah melalui proses pengecekan pertama yaitu, *noise* diperiksa tingkat akurasi pada pengklasifikasian gambar Ulos kategori *good* diperoleh akurasi 22.16%

5.4 Hasil Eksperimen Evaluasi Gambar

Dari analisis, eksperimen serta evaluasi yang telah dilakukan oleh peneliti untuk mendapatkan ketetapan *threshold* pada parameter *blur* dan *noise*, maka dapat dilihat pada Tabel 9 untuk parameter *blur* dan Tabel 10 untuk parameter *noise* sebagai ketetapan *threshold* yang final.

Tabel 8. Thresold parameter blur

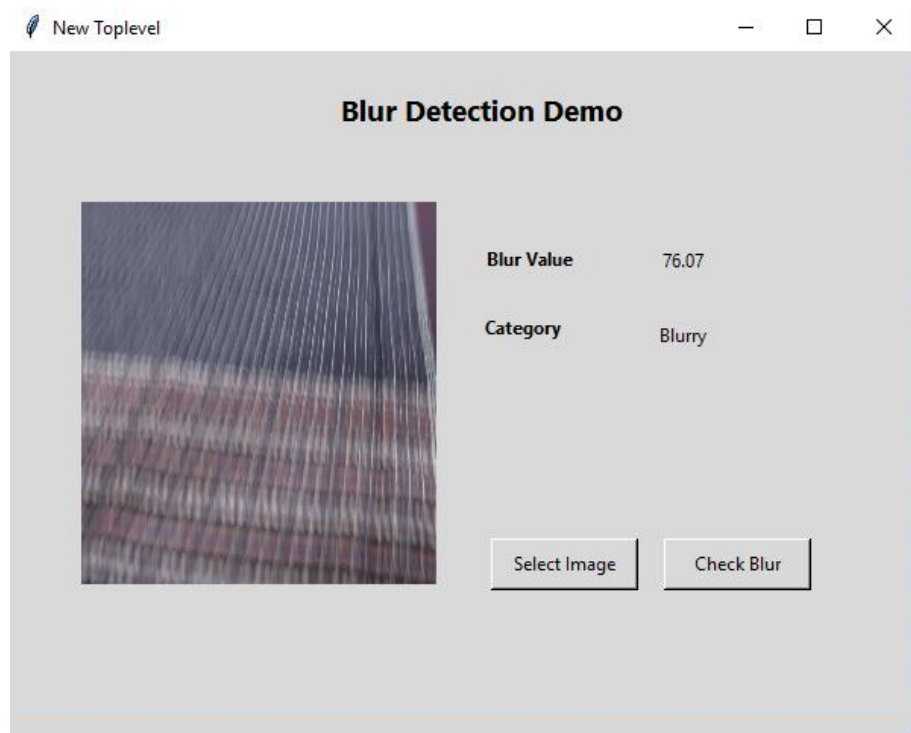
Kategori	<i>Threshold</i>	Akurasi
<i>Bad</i>	0 – 400	67,72%
<i>Improve</i>	400 – 1500	16,03%
<i>Good</i>	> 1500	26,41%

Tabel 9. Thresold parameter noise

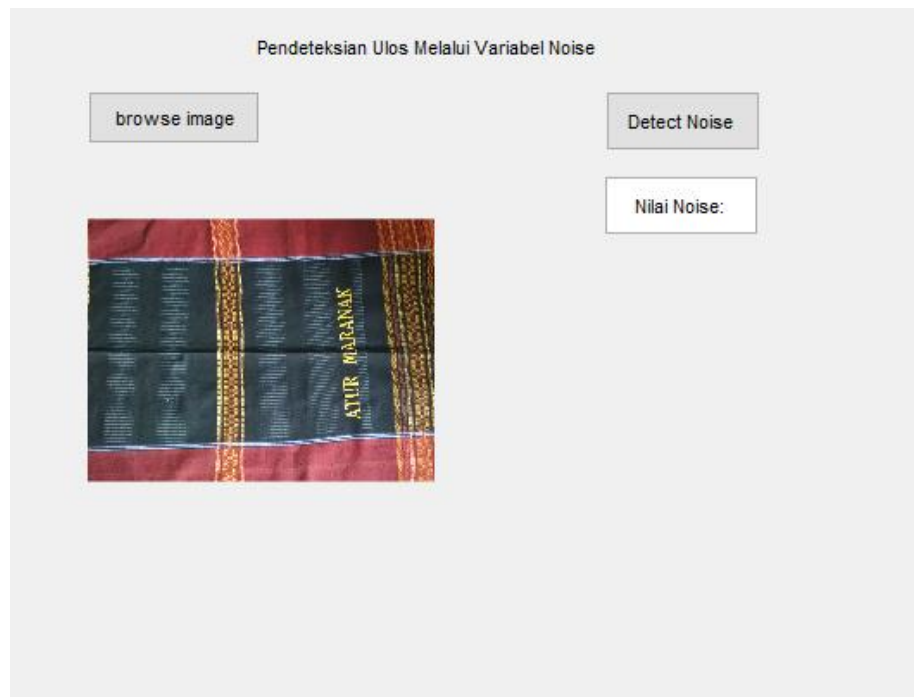
Kategori	<i>Threshold</i>	Akurasi
<i>Improve</i>	0	14,62%
<i>Good</i>	≥ 1	22,16%

5.5 Hasil Implementasi GUI

Berikut pada Gambar 14 dan Gambar 15 adalah hasil implementasi GUI yang telah dikerjakan oleh peneliti. GUI dibuat untuk tujuan mempermudah proses mendeteksi *noise* dan *blur* dari suatu gambar Ulos.



Gambar 14 Tampilan fitur pendeteksi blur pada gambar Ulos



Gambar 15 Tampilan fitur pendeteksi noise pada gambar Ulos

5.6 Pembahasan Hasil

Berikut ini dibahas mengenai hasil yang telah diperoleh dari eksperimen serta evaluasi yang dilakukan, yaitu:

1. Pada parameter *blur* dan *noise* kategori *improve* dan *good* belum memperoleh akurasi yang tinggi dikarenakan beberapa hal seperti:
 - 1.1 Pengklasifikasian secara manual yang dilakukan oleh peneliti masih memiliki keterbatasan pada kategori *improve* dan *good*, yaitu peneliti masih sulit mengkategorikan gambar masukan yang dapat diperbaiki (*improve*) dengan kategori yang bagus (*good*) secara kasat mata
 - 1.2 Ketergantungan modul ini dengan modul lain seperti modul *enhancement* dan modul *classification* menyebabkan hasil eksperimen memiliki nilai akurasi yang rendah dikarenakan tingkat akurasi pada modul lain yang digunakan juga masih terkategori rendah dan belum akurat. Untuk modul *enhancement* tidak terdapat modul yang mampu memperbaiki *blur*, jadi peneliti menggunakan proses *contrast enhancement* untuk lebih mencerahkan gambar masukan sehingga hal ini menjadi salah satu hal yang mempengaruhi tingkat keakuratan dalam mengevaluasi gambar. Sedangkan, pada parameter *noise* digunakan proses *denoising*. Proses ini memang sesuai digunakan untuk menurunkan nilai *noise* pada gambar masukan. Sedangkan pada modul *classification* yang mempengaruhi tingkat akurasinya ialah adanya beberapa keterbatasan modul tersebut seperti jumlah data dan variasi data yang masih digunakan dalam jumlah yang minim, serta kemiripan data gambar Ulos pada modul ini yang menyebabkan data yang dipelajari dianggap mirip oleh mesin.
2. Pada parameter *blur* dengan kategori *bad* memiliki tingkat akurasi yang tinggi didukung oleh pengkategorian manual yang lebih mudah karena secara kasat mata lebih dapat dikategorikan.
3. Dari hasil eksperimen dapat disimpulkan bahwa pengklasifikasian kualitas gambar masukan dengan menggunakan dua parameter *blur* dan *noise* lebih baik dikarenakan pemrosesan gambar masukan lebih .

6. Kesimpulan dan Saran

Pada bab ini dijelaskan berisi kesimpulan dan saran-saran pada penelitian selanjutnya yang diperoleh dari keseluruhan proses evaluasi gambar yang akan digunakan di Aplikasi *mobile JTenun*.

6.1 Kesimpulan

Dari pengerjaan penelitian Tugas Akhir ini, peneliti telah memaparkan hasil-hasil percobaan serta evaluasi penelitian. Walaupun memiliki beberapa kekurangan, peneliti mampu menarik kesimpulan bahwa:

1. Penelitian ini menggunakan dua parameter yaitu *blur* dan *noise* untuk mengukur kualitas gambar masukan, namun dari hasil eksperimen menunjukkan bahwa pengklasifikasian gambar Ulos lebih baik menggunakan dua parameter *blur* dan *noise* dikarenakan akurasi yang lebih tinggi dibandingkan dengan menggunakan salah satu parameter *blur* atau *noise* saja.
2. Penelitian ini telah menghasikan modul yang mampu mengklasifikasikan gambar masukan ke dalam tiga kategori yaitu kategori *good*, *improve*, dan *bad*.
3. Dari ketiga kategori tersebut, kategori *bad* memperoleh nilai akurasi tertinggi dan untuk dua kategori lainnya, *improve* dan *good* masih rendah dikarenakan dua hal yaitu pengkategorian secara manual masih sulit dilakukan dan masih ada ketergantungan pada modul penelitian lain yang belum tuntas yaitu modul *enhancement* dan modul *classification*.

6.2 Saran

Pada sub bab ini dipaparkan mengenai saran untuk melakukan penelitian selanjutnya. Saran tersebut adalah sebagai berikut:

1. Untuk penelitian lanjutan dari penelitian ini, diharapkan peneliti lebih melakukan eksplorasi terhadap gambar Ulos sehingga peneliti mampu mendapatkan nilai piksel pada gambar masukan melalui pendekatan *machine learning* karena melalui *machine learning*, *threshold* akan mengikuti gambar masukan sebagai data yang ada.

2. Melakukan penelitian menggunakan gambar Ulos yang ditangkap dari kamera digital bukan lagi gambar yang ditangkap oleh kamera *smartphone* karena piksel pada gambar Ulos digital lebih bagus dibanding dengan gambar yang ditangkap oleh kamera *smartphone*.
3. Penelitian selanjutnya juga lebih baik dianalisis berdasarkan jenis Ulos karena pada penelitian ini didapat beberapa jenis Ulos yang berbeda seperti Ulos Sadum yang memiliki banyak bintik-bintik sehingga dianggap memiliki banyak *noise*.

Daftar Pustaka

- [1] A. C. Barus, "Piranti Cerdas Penghasil Motif Tenun Nusantara," p. 23, 2015.
- [2] Ahmed, M. G. a. M., 2016. Image Quality Parameter Detection : A Study. *International Journal of Computer Sciences and Engineering*, IV(7), pp. 110-116.
- [3] Budiyono and dkk, "Eksplorasi Tenun Sambas dengan Style Edgy Pada Busana," in *Kriya Tekstil Jilid 1*, Jakarta, Direktorat, 2008, p. 421.
- [4] C.Sasi varnan, A. Jagan, Jaspreet Kaur, Divya Jyoti and Dr. D. S. Rao, "Image Quality Assessment Techniques pn Spatial Domain", ISSN: 2229-4333(Print) ISSN, 0976-8491 (Online), IJCST VOL.2, Issue 3, September 2011
- [5] Calongesi. J, S. 1995. Merancang Tes untuk Menilai Prestasi Siswa. Bandung: ITB.
- [6] E. K. d. K. U. Irwan Prasetya Gunawan, "IDENTIFIKASI DISTORSI BLUR PADA GAMBAR DIGITAL," p. 8, 2012.
- [7] HIDAYAH, N., 2016. IMPLEMENTASI PERBAIKAN SISI CITRA MENGGUNAKAN METODE TRANSFORMASI FOURIER DAN FAST FOURIER TRANSFORM. *INFOTEK*, Volume III, pp. 94-99.
- [8] Hutapea Oppir, P. M. T. D., 2018. Image Enhancement with Histogram Equalization and Median Filtering Approach. p. 110.
- [9] Ikhlas AbdelQader, Osama Abudayyeh, Michael E .Kelly (2003), "Analysis o f Edge-Detection Techniques for Crack Identification in Bridges" on Fast f ouri es transform.
- [10] Jianping Shi, L. X. a. J. J., 2014 . Discriminative Blur Detection Features. pp. 2965 - 2972 .
- [11] Lehman, H. (1990). The Systems Approach to Education. Special Presentatiom Conveyed in The International Seminar on Educational Innovation and Technology Manila. Innotech Publications-Vol 20 No. 05.
- [12] N. S. d. W. S. Ahmad Saikhu, "PERBAIKAN CITRA BER-NOISE MENGGUNAKAN SWITCHING MEDIAN FILTER DAN BOUNDARY DISCRIMINATIVE NOISE DETECTION," *Seminar Nasional Aplikasi Teknologi Informasi*, pp. 1-46, 2009.
- [13] Ngarap Im. Manik, M., 2012. PENGGUNAAN MODEL FRAKTAL UNTUK PENGEMBANGAN MOTIF ULOS. *Mat Stat*, Volume 12, pp. 143-151.
- [14] Pardosi, J., 2008. MAKNA SIMBOLIK UMPASA, SINAMOT, DAN ULOS PADA ADAT PERKAWINAN BATAK TOBA. Volume IV, pp. 101-107.
- [15] Raghav Bansal, G. R. a. T. C., 2016. Blur Image Detection using Laplacian Operator and Open-CV. *IEEE*, pp. 1-5.

- [16] Renting Liu, Z. L. a. J. J., n.d. Image Partial Blur Detection and Classification*. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8.
- [17] Rockmore, D. (2000). The FFT - an algorithm the whole family can use. *Computing in Science & Engineering* (pp. 60-64).
- [18] S. Kartiwa, "Ragam Kain Tradisional Indonesia," in *Tenun ikat = Indonesian ikats*, S. Kartiwa, Ed., Gramedia Pustaka Utama, 2007, p. 156.
- [19] Sonu Jain, A. D. D. S. C. P. K. S., 2014. Image Deblurring from Blurred Images. *International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014)*, II(3), pp. 41-45.
- [20] Suta Wijaya, H. d. G., 2012. PENERAPAN ALGORITMA PRINCIPLE COMPONENT ANALYSIS (PCA) DAN FITUR RGB UNTUK PELACAKAN JENIS DAN WARNA BUAH. pp. 1-13
- [21] Waruhu Johannes, H. B. P. K., 2018. Image Classification using Deep Learning Case study: Ulos Batak Toba. *Tugas Akhir*, p. 123.
- [22] Xinhao Liu, M. T. a. M. O., 2012. NOISE LEVEL ESTIMATION USING WEAK TEXTURED PATCHES OF A SINGLE NOISY IMAGE. *IEEE*, pp. 1-4.
- [23] Y. Erlyana, "KAJIAN VISUAL KERAGAMAN CORAK," p. 46, 01 April 2016.
- [24] Yuda Permadi, M., 2015. APLIKASI PENGOLAHAN CITRA UNTUK IDENTIFIKASI KEMATANGAN MENTIMUN BERDASARKAN TEKSTUR KULIT BUAH MENGGUNAKAN METODE EKSTRAKSI CIRI STATISTIK. *JURNAL INFORMATIKA*, Volume IX, pp. 1028-1038.
- [25] Zhu Xiang and Milanfar Peyman, "Automatic parameter selection for denoising algorithms using a no reference measure of image content," *IEEE trans. on image processing*, vol. 19, pp.3116–32, 2010.