

**APLIKASI PENGOLAHAN CITRA DIGITAL PENGHITUNG
JUMLAH PENGUNJUNG MALL MENGGUNAKAN METODE
MORPHOLOGY**



**Disusun Oleh:
CITRA MULIANA BARAGAIN
10.18.178**

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2014**

LEMBAR PERSETUJUAN DAN PENGESAHAN

**APLIKASI PENGOLAHAN CITRA DIGITAL PENGHITUNG
JUMLAH PENGUNJUNG MALL MENGGUNAKAN METODE
MORPHOLOGY**

SKRIPSI

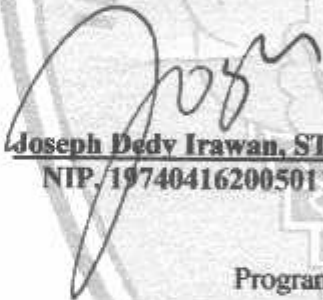
*Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan guna
mencapai Gelar Sarjana Komputer Strata Satu (S-1)*

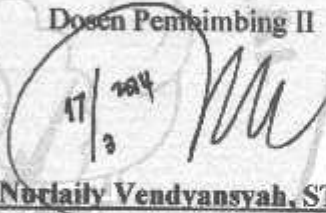
Disusun Oleh :
Citra Muliana Baragain
10.18.178

Diperiksa dan Disetujui,


Dosen Pembimbing I

Dosen Pembimbing II


Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002


Norlaily Vendvansyah, ST

Program Studi Teknik Informatika S-1


Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2014


MILIK
PERPUSTAKAAN
ITN MALANG

APLIKASI PENGOLAHAN CITRA DIGITAL PENGHITUNG JUMLAH PENGUNJUNG MALL MENGGUNAKAN METODE MORPHOLOGY

Citra Muliana Baragain

(10.18.178)

Jurusan Teknik Informatika S-1

Fakultas Teknologi Industri, Institut Teknologi Nasional Malang

Kampus ITN II Jalan Raya Karanglo KM 2 Malang

Email: liana.dewi77@yahoo.com

Dosen Pembimbing : **1. Joseph Dedy Irawan, ST., MT.**

2. Nurlaily Vendyansyah, ST

Abstrak

Mall sebagai salah satu pusat perbelanjaan yang sering dikunjungi oleh banyak orang untuk berbelanja bahkan untuk sekedar melepaskan penat dalam kesibukan sehari-hari. Banyaknya pengunjung yang datang ke Mall tidak dapat di hitung secara manual. Berdasarkan permasalahan tersebut maka penulis merancang sebuah Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pengunjung Mall Menggunakan Metode Morphology untuk mengetahui dan menghitung jumlah pengunjung Mall. Hasil dari aplikasi ini akan membantu pihak yang akan membuat even Di Mall tetapi ingin mengetahui terlebih dahulu jumlah pengunjung.

Aplikasi dibangun menggunakan Metode Morphology sebagai alat untuk proses mengekstrak komponen-komponen citra yang berguna dalam representasi dan destripsi dari suatu bentuk wilayah dalam citra. Operasi-operasi Morphology yang digunakan yaitu operasi Erosi dan operasi Dilasi.

Objek yang bergerak dihitung dan dikenali sebagai mamusia. Aplikasi mampu menghitung objek pada Mall tanpa mengenali objek masuk maupun keluar. Pengujian yang dilakukan adalah Aplikasi mampu mendeteksi objek yang nampak pada citra, seperti manusia, tas, keranjang, hewan dan benda hergerak, deteksi dapat dilakukan apabila jarak setiap objek pada citra tidak terlalu dekat dan berdasarkan pengujian menggunakan Metode Morhology mamusia yang saling berdekatan dihitung satu.

Kata kunci : Morphology, Erosi, Dilasi, Pengolahan Citra Digital, Filtering.

Lembar Persembahan

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dengan Rahmat Allah yang Maha Pengasih Lagi Maha Penyayang. Dengan rasa syukur yang mendalam saya persembahkan karya sederhana ini, untuk cahaya hidup, yang senantiasa ada saat suka maupun duka, selalu setia mendampingi, selalu memberikan saya semangat Ayah dan Ibu tercinta (Ibu Mukjirah & Bapak, Butje Baragain) yang selalu memanjatkan do'a untuk saya dalam setiap sujudnya. Terima kasih untuk semuanya. Terima kasih atas limpahan do'a dan kasih sayang yang tak terhingga dan selalu memberikan yang terbaik,

Terima kasih kepada kakak saya (Rizki Feri Baragain) dan Adik saya (Ridho Sadevo Baragain) yang telah memberikan saya semangat dan do'a untuk dapat menyelesaikan karya sederhana ini, dan untuk seluruh keluarga saya yang selalu mendukung dan mendo'akan saya.

Terima kasih kepada sahabat tercinta Kristin Adityas Oprantina Kasih Pattinama, Mega Putri Darliana, Yessy Paradipa, Wahyuni dan semuanya yang sudah menemani saya selama Di Malang dan sudah memberikan banyak warna dalam hidup saya, memberikan semangat kepada saya, yang setia mendengarkan keluh kesah saya. Terima kasih untuk kebersamaannya selama ini, kebersamaan yang tidak akan pernah terlupakan.

Terima kasih kepada Ibu & Bapak Dosen yang sudah mengajar dan memberikan saya banyak ilmu sehingga saya dapat menyelesaikan karya sederhana ini, dan terima kasih untuk seluruh Staf Institut Teknologi Nasional Malang khususnya Teknik Informatika yang telah membantu saya dari awal sampai lulus.

Terima kasih kepada teman-teman Teknik Informatika Institut Teknologi Nasional Malang Angkatan 2010 senasib, sepejuangan dan sepenanggungan, terima kasih atas gelak tawa dan solidaritas yang luar biasa sehingga membuat hari-hari semasa kuliah lebih berarti. Semoga tak ada lagi duka di dada tapi suka dan bahagia juga tawa dan canda.

Semoga Allah SWT membalas jasa budi kalian semua dikemudian hari dan memberikan kemudahan dalam segala hal, semoga kesuksesan selalu menghampiri kita semua. Amin...

~Citra Muliana Baragain~

Akhir kata penulis mohon maaf yang sebesar-besarnya apabila dalam penyusunan Skripsi ini terdapat kekurangan serta kesalahan. Semoga Skripsi ini bermanfaat bagi semua.

Malang, 11 Februari 2014

Penulis

DAFTAR ISI

Halaman Judul	i
Lembar Persetujuan dan Pengesahan	ii
Surat Orisinalitas	iii
Abstrak	iv
Lembar Persembahan	v
Kata Pengantar	vi
Daftar Isi	viii
Daftar Gambar.....	xi
Daftar Tabel	xii
Bab I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Penulisan	3
BAB II Landasan Teori	5
2.1 Pengolahan Citra Digital	5
2.2 Komponen Citra Digital	7
2.2.1 Representasi Citra Digital	9
2.2.2 Jenis-Jenis Citra Digital	10
2.2.2.1 Citra Biner	10
2.2.2.2 Citra Grayscale	10
2.2.2.3 Citra Warna (<i>True Color</i>)	11
2.2.3 Elemen-elemen Citra Digital	11
2.2.4 Format Citra Digital	13
2.2.5 Representasi Warna	13
2.3 Metode Morphology	14
2.4 Java	17
2.5 NetBeans IDE 6.5	19
2.6 MySQL	20

BAB III Perancangan Aplikasi	22
3.1 Perancangan Sistem	22
3.2 Desain Sistem	22
3.3 Perancangan Aplikasi Sistem	23
3.4 Perancangan Diagram Alir	24
3.5 Proses Pengolahan Citra	25
3.5.1 Mengubah Citra Awal Menjadi Citra Grayscale	25
3.6 Proses Perhitungan Jumlah Pengunjung	25
3.7 Perancangan Interface	29
3.7.1 Tampilan Form Awal	29
3.7.2 Tampilan Form Kedua	30
BAB IV Implementasi Dan Pengujian	31
4.1 Implementasi Sistem	31
4.2 Spesifikasi Perangkat Lunak	31
4.3 Pengujian Sistem	32
4.4 Halaman Secara Manual.....	32
4.4.1 Tampilan Halaman Utama	33
4.4.2 Tampilan halaman Menu Browse	33
4.4.3 Tampilan Halaman Setelah Menu Browse	34
4.4.4 Tampilan Halaman Untuk Memilih Durasi	34
4.4.5 Tampilan Halaman Menu Play	35
4.4.6 Tampilan Halaman Menu Simpan	35
4.4.7 Tampilan Halaman Menu Lihat Data	36
4.5 Tampilan Secara Realtime	36
4.5.1 Halaman Awal Saat Program Jalan	36
4.5.2 Halaman Setelah Memilih Nama Perangkat Kamera	37
4.5.3 Halaman Perhitungan Pengunjung Secara Realtime	37
4.5.4 Halaman Untuk Melihat Jumlah Pengunjung Yang Tersimpan.....	38
4.6 Pengujian Non Fungsional	38
4.7 Pengujian Aplikasi Menggunakan Metode Morphology	39

BAB V Kesimpulan Dan Saran	41
5.1 Kesimpulan	41
5.2 Saran	41
Daftar Pustaka	42
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1 Tahap-Tahap Pengolahan Citra Digital	5
Gambar 2.2 Citra Biner	10
Gambar 2.3 Citra Grayscale 2 bit	10
Gambar 2.4 Citra Grayscale 3 bit	11
Gambar 2.5 Vektor Representasi Warna	14
Gambar 2.6 Contoh Proses Dilasi	15
Gambar 2.7 Contoh Proses Erosi	16
Gambar 2.8 Logo Java	18
Gambar 2.9 Logo NetBeans	21
Gambar 3.1 Konsep dasar aplikasi penghitung jumlah pengunjung pada Mall.....	23
Gambar 3.2 Alur proses aplikasi penghitung jumlah pengunjung pada Mall	24
Gambar 3.3 Diagram Alir Program Aplikasi	25
Gambar 3.4 Flowchart Sistem Aplikasi	27
Gambar 3.5 Flowchart Penyimpanan ke Database	29
Gambar 4.1 Tampilan Halaman Menu Utama	34
Gambar 4.2 Tampilan Menu Browse	34
Gambar 4.3 Tampilan Setelah Browse	35
Gambar 4.4 Tampilan Memilih Durasi	35
Gambar 4.5 Tampilan Menu Play	36
Gambar 4.6 Tampilan Menu Simpan	36
Gambar 4.7 Tampilan Menu Lihat Data	37
Gambar 4.8 Tampilan Awal Saat Program Jalan	37
Gambar 4.9 Tampilan Setelah Memilih Nama Perangkat Kamera.....	38
Gambar 4.10 Tampilan Hitung Secara Realtime	38
Gambar 4.11 Tampilan Melihat Jumlah Pengunjung Yang Tersimpan	39

DAFTAR TABEL

Tabel 4.1 Pengujian Non Fungsional Sistem	39
Tabel 4.2 Pengujian Aplikasi Menggunakan Metode Morphology.....	40

BAB I

PENDAHULUAN

1.1 Latar Belakang

Mall sebagai salah satu pusat perbelanjaan yang sering dikunjungi oleh banyak orang untuk berbelanja bahkan untuk sekedar melepaskan penat dalam kesibukan sehari-hari, banyaknya pengunjung yang datang ke Mall tidak dapat di hitung secara manual oleh pihak Manajemen Mall. Oleh karena itu penulis membuat program yang berfungsi untuk mengetahui dan menghitung jumlah pengunjung pada Mall dalam setiap harinya. Jika seseorang ingin membuat suatu kegiatan atau event pada Mall tetapi ingin mengetahui terlebih dahulu jumlah pengunjung yang terbanyak pada saat kapan, agar kegiatan atau event dapat di laksanakan dengan melihat jumlah pengunjung terbanyak yang sudah terhitung oleh program yang di buat. Selain itu, aplikasi menghitung keseluruhan jumlah pengunjung baik yang masuk maupun yang keluar. Jika pihak Mall ingin melakukan peramalan pembelian dan investasi tetapi kesulitan karena tidak dapat mengetahui jumlah pengunjung yang datang pada Mall maka program akan membantu untuk menghitung jumlah pengunjung terbanyak yang ada pada Mall.

Oleh karena itu untuk mengatasi permasalahan tersebut perlu di kembangkan program menggunakan Pengolahan Citra Digital untuk membantu pihak Mall untuk menghitung jumlah pengunjung dengan lebih cepat dan tepat. Pengembangan Pengolahan Citra Digital menggunakan metode Morphology yang langsung berfokus pada kebutuhan pengguna. Melalui aplikasi Pengolahan Citra dengan Metode Morphology ini di harapkan dapat dengan mudah membantu pengguna dalam melakukan perhitungan pengunjung pada Mall.

1.2 Rumusan Masalah

Berdasarkan pada latar belakang yang ada rumusan masalah yang dapat dibentuk adalah bagaimana merancang suatu aplikasi Menggunakan Pengolah Citra Digital dengan Metode Morphology untuk membantu pengguna dalam melakukan perhitungan jumlah pengunjung pada Mall.

1.3 Batasan Masalah

Agar pembahasan dalam skripsi ini tidak meluas, maka ditentukan beberapa batasan masalah diantaranya, yaitu :

1. Aplikasi penghitung jumlah pengunjung, bagaimana menghitung pengunjung Mall.
2. Aplikasi dibuat untuk menghitung jumlah pengunjung masuk dan keluar.
3. Pada penelitian ini menggunakan metode pengolahan citra digital yaitu metode Morphology.
4. Kamera yang di gunakan adalah kamera VGA, kamera Sony Xperia P, kamera Sony Cybershot DSC-W710.
5. Semua objek yang bergerak dikenali sebagai manusia.
6. Menggunakan Database MySQL Versi 5.5.30, untuk melakukan penyimpanan data.

1.4 Tujuan

Tujuan dari penyusunan skripsi ini adalah menghasilkan Aplikasi menggunakan Pengolahan Citra Digital untuk menghitung jumlah pengunjung setiap harinya sehingga dapat membantu pengguna dalam melakukan perhitungan jumlah pengunjung. Menggunakan metode Morphology untuk pengenalan bentuk.

1.5 Metodologi Penelitian

1. Studi Literatur

Pengumpulan data yang dilakukan dengan mencari bahan-bahan kepustakaan dan referensi dari berbagai sumber sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan objek penelitian.

2. Analisa Kebutuhan Sistem

Data dan informasi yang telah diperoleh akan di analisa agar di dapatkan kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem baik hardware maupun software, di mana nantinya akan di gunakan sebagai acuan perancangan sistem.

3. Perancangan Sistem

Berdasarkan data dan informasi yang telah di peroleh serta analisa kebutuhan untuk membangun sistem ini, akan di buat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan di buat dan di implementasikan ke dalam sistem.

4. Eksperimen dan Evaluasi

Pada tahap ini, sistem yang telah selesai di buat akan di uji coba, yaitu pengujian berdasarkan fungsionalitas program dan akan di lakukan koreksi dan penyempurnaan program jika di perlukan.

1.6 Sistematika Penulisan

Dalam menyusun skripsi ini, penulis menggunakan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Bab ini merupakan dasar penyusunan skripsi yang di dalamnya berisi tentang latar belakang masalah, rumusan masalah, tujuan skripsi, batasan masalah, metodologi pengembangan sistem, dan sistematika pembahasan skripsi.

BAB II LANDASAN TEORI

Berisi landasan teori dari beberapa literatur yang berhubungan dengan topik skripsi ini.

BAB III ANALISIS DAN DESAIN

Berisi penjelasan desain mulai dari awal hingga akhir perancangan sistem.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi tentang tahap implementasi yaitu identifikasi untuk menyelesaikan masalah yang di hadapi berdasarkan teori pada Bab II dan Bab III. Bab ini juga berisi hasil pengujian sistem.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan dan saran dari hasil penyusunan skripsi yang telah disusun.

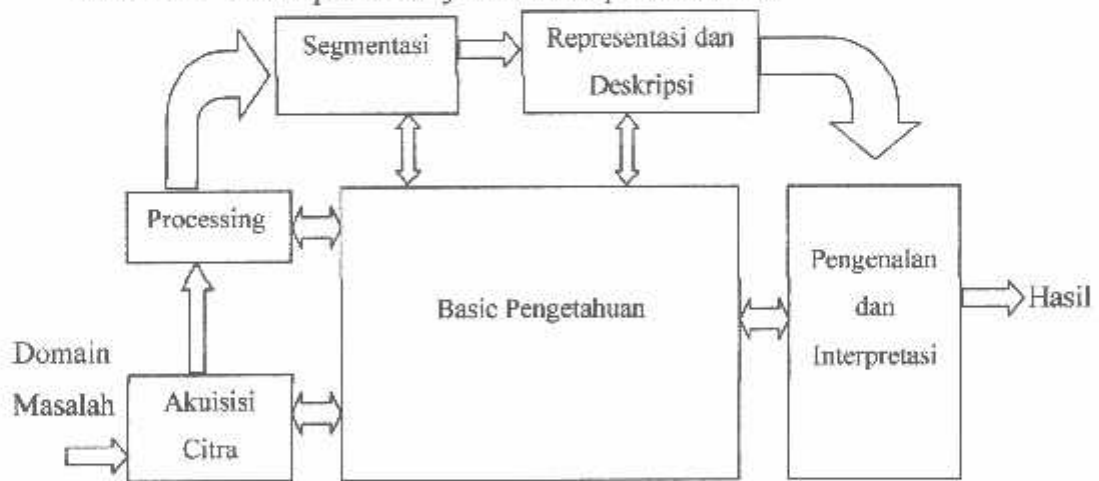
BAB II

LANDASAN TEORI

2.1 Pengolahan Citra Digital

Pengolahan Citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh bit-bit tertentu.

Istilah "citra" yang digunakan dalam bidang pengolahan citra dapat diartikan sebagai suatu fungsi kontinu dari intensitas cahaya dalam bidang dua dimensi. Pemrosesan citra dengan komputer digital membutuhkan citra digital sebagai masukannya. Citra digital adalah kontinu yang diubah dalam bentuk diskrit, baik koordinat ruang maupun intensitas cahayanya. Pengolahan digitalisasi terdiri dari dua proses, yaitu pencuplikan (sampling) posisi, dan kuantisasi intensitas. Citra digital dapat dinyatakan dalam matriks dua dimensi $f(x,y)$ dimana 'x' dan 'y' merupakan koordinat piksel dalam matriks dan 'f' merupakan derajat intensitas piksel tersebut.^[6]



Gambar 2.1 Tahap-tahap pengolahan citra digital^[3]

Keterangan gambar 2.1 sebagai berikut :

1. Akuisisi citra

Akuisisi citra adalah tahap awal untuk mendapatkan citra digital. Tujuan akuisisi citra adalah untuk menentukan data yang diperlukan dan memilih metode perekaman citra digital. Tahap ini dimulai dari objek yang diambil gambarnya, persiapan alat-alat, sampai pada pencitraan. Pencitraan adalah kegiatan transformasi dari citra tampak (foto, gambar, lukisan, patung, pemandangan, dan lain-lain) menjadi citra digital. Beberapa alat yang dapat digunakan untuk pencitraan adalah :^[3]

- a. Video kamera
- b. Kamera digital
- c. Kamera konvensional dan konverter analog to digital
- d. Scanner
- e. Photo sinar x/sinar infra merah

Hasil dari akuisisi citra ini ditentukan oleh kemampuan sensor untuk mendigitalisasi sinyal yang terkumpul pada sensor tersebut. Kemampuan digitalisasi alat ditentukan oleh resolusi alat tersebut.

2. Preprocessing

Tahapan ini diperlukan untuk menjamin kelancaran pada proses berikutnya. Hal-hal penting yang dilakukan pada tingkatan ini diantaranya adalah :^[3]

- a. Peningkatan kualitas citra (kontras, *brightness*, dan lain-lain)
- b. Menghilangkan noise
- c. Perbaikan citra (*image restoration*)
- d. Transformasi (*image transformation*)
- e. Menentukan bagian citra yang akan diobservasi

3. Segmentasi

Tahapan ini bertujuan untuk mempartisi citra menjadi bagian-bagian pokok yang mengandung informasi penting. Misalnya, memisahkan objek dan latar belakang.^[3]

4. Representasi dan deskripsi

Dalam hal ini representasi merupakan suatu proses untuk merepresentasikan suatu wilayah sebagai suatu daftar titik-titik koordinat dalam kurva yang tertutup, dengan deskripsi luasan atau perimeternya. Setelah suatu wilayah dapat direpresentasi, proses selanjutnya adalah melakukan deskripsi citra dengan cara seleksi ciri dan ekstraksi ciri (*Feature Extraction and Selection*). Seleksi ciri bertujuan untuk memilih informasi kuantitatif dari ciri yang ada, yang dapat membedakan kelas kelas objek secara baik, sedangkan ekstraksi ciri bertujuan untuk mengukur besaran kuantitatif ciri setiap piksel. Misalnya rata-rata, standar deviasi, koefisien variasi, *Signal to Noise Ratio* (SNR), dan lain-lain.^[3]

5. Pengenalan dan Interpretasi

Tahap pengenalan bertujuan untuk memberi label pada sebuah objek yang informasinya disediakan oleh *descriptor*, sedangkan tahap interpretasi bertujuan untuk memberi arti atau makna kepada kelompok objek-objek yang dikenali.^[3]

6. Basis pengetahuan

Basis pengetahuan sebagai basis data pengetahuan berguna untuk memandu operasi dari masing-masing modul proses dan mengontrol interaksi antara modul-modul tersebut selain itu, basis pengetahuan juga digunakan sebagai referensi pada proses *template matching* atau sebagai pengenalan pola.^[3]

2.2 Komponen Citra Digital

Citra adalah representasi dua dimensi untuk bentuk fisik nyata tiga dimensi. Citra dalam perwujudannya dapat bermacam-macam, mulai dari gambar hitam-putih pada sebuah foto (yang tidak bergerak) sampai pada gambar berwarna yang bergerak pada pesawat televisi. Proses transformasi dari bentuk tiga dimensi ke bentuk dua dimensi untuk menghasilkan citra akan dipengaruhi oleh bermacam-macam faktor yang mengakibatkan penampilan citra suatu benda tidak sama persis dengan bentuk fisik nyatanya. Faktor-faktor tersebut merupakan efek degradasi atau penurunan kualitas yang dapat berupa rentang kontras benda yang terlalu sempit atau terlalu lebar, distorsi geometrik, kekaburan (*blur*), kekaburan akibat

obyek yang bergerak (*motion blur*), noise atau gangguan yang disebabkan oleh interferensi peralatan pembuat citra, baik berupa transduser, peralatan elektronik ataupun peralatan optik.^[1]

Teknik dan proses untuk mengurangi atau menghilangkan efek degradasi pada citra digital meliputi perbaikan/peningkatan citra (*image enhancement*), restorasi citra (*image restoration*), dan transformasi spasial (*spasial transformation*). Subyek lain dari pengolahan citra digital diantaranya adalah pengkodean citra (*image coding*), segmentasi citra (*image segmentation*), representasi dan deskripsi citra (*image representation and description*).

Pengolahan citra dilakukan dengan komputer digital maka citra yang akan diolah terlebih dahulu ditransformasikan ke dalam bentuk besaran-besaran diskrit dari nilai tingkat keabuan pada titik-titik elemen citra. Bentuk citra ini disebut citra digital. Setiap citra digital memiliki beberapa karakteristik, antara lain ukuran citra, resolusi dan format lainnya. Umumnya citra digital berbentuk persegi panjang yang memiliki lebar dan tinggi tertentu, yang biasanya dinyatakan dalam banyaknya titik atau piksel (*picture elemen/pixel*).^[1]

Ukuran citra dapat juga dinyatakan secara fisik dalam satuan panjang (misalnya mm atau inch). Dalam hal ini tentu saja harus ada hubungan antara ukuran titik penyusun citra dengan satuan panjang. Hal tersebut dinyatakan dengan resolusi yang merupakan ukuran banyaknya titik untuk setiap satuan panjang. Biasanya satuan yang digunakan adalah dpi (*dot per inch*). Makin besar resolusi makin banyak titik yang terkandung dalam citra dengan ukuran fisik yang sama. Hal ini memberikan efek penampakan citra menjadi semakin halus.^[1]

Format citra digital ada bermacam-macam. Karena sebenarnya citra merepresentasikan informasi tertentu, sedangkan informasi tersebut dapat dinyatakan secara bervariasi, maka citra yang mewakilinya dapat muncul dalam berbagai format. Citra yang merepresentasikan informasi yang hanya bersifat biner untuk membedakan dua keadaan tentu tidak sama citra dengan informasi yang lebih kompleks sehingga memerlukan lebih banyak keadaan yang diwakilinya. Pada citra digital semua informasi tadi disimpan dalam bentuk angka, sedangkan penampilan angka tersebut biasanya dikaitkan dengan warna.^[1]

Citra digital (*digital image*) adalah citra kontinu $f(x,y)$ yang sudah di diskritkan baik koordinat spasial maupun tingkat kecerahannya. Setiap titik biasanya memiliki koordinat sesuai dengan posisinya dalam citra. Koordinat ini biasanya dinyatakan indeks x dan y hanya bernilai bilangan bulat positif, yang dapat dimulai dari 0 atau 1. Citra digital yang selanjutnya akan disingkat "citra" sebagai matrik ukuran $M \times N$ yang baris dan kolomnya menunjukkan titik-titik yang diperlihatkan pada persamaan 2.2.^[1]

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix} \dots\dots\dots(2-1)$$

Setiap titik juga memiliki nilai berupa angka digital yang merepresentasikan informasi yang diwakili titik tersebut. Format nilai piksel sama dengan format citra keseluruhan. Pada kebanyakan sistem pencitraan, nilai ini biasanya berupa bilangan bulat positif.^[1]

2.2.1 Representasi Citra Digital

Komputer dapat mengolah isyarat-isyarat elektronik digital yang merupakan kumpulan sinyal biner (bernilai dua: 0 dan 1). Untuk itu, citra digital harus mempunyai format tertentu yang sesuai sehingga dapat merepresentasikan obyek pencitraan dalam bentuk kombinasi data biner.^[5]

Citra yang tidak berwarna atau hitam putih dikenal sebagai citra dengan derajat abu-abu (*citra graylevel/grayscale*). Derajat abu-abu yang dimiliki ini bisa beragam mulai dari 2 derajat abu-abu (yaitu 0 dan 1) yang dikenal juga sebagai citra monochrome, 16 derajat keabuan dan 256 derajat keabuan.

Dalam sebuah citra monochrome, sebuah piksel diwakili oleh 1 bit data yang berisikan data tentang derajat keabuan yang dimiliki piksel tersebut. Data akan berisi 0 bila piksel berwarna hitam dan 1 bila piksel berwarna putih. Citra yang memiliki 16 derajat keabuan (mulai dari 0 yang mewakili warna hitam sampai dengan 15 yang mewakili warna putih) direpresentasikan oleh 4 bit data. Sedangkan citra dengan 256 derajat keabuan (nilai dari 0 yang mewakili warna

hitam sampai dengan 255 yang mewakili warna putih) direpresentasikan oleh 8 bit data.^[5]

Dalam citra berwarna, jumlah warna bisa beragam mulai dari 16, 256, 65536 atau 16 juta warna yang masing-masing direpresentasikan oleh 4,8,16 atau 24 bit data untuk setiap pikselnya. Warna yang ada terdiri dari 3 komponen utama yaitu nilai merah (*red*), nilai hijau (*green*) dan nilai biru (*blue*). Paduan ketiga komponen utama pembentuk warna tersebut dikenal sebagai RGB color yang nantinya akan membentuk citra warna.^[5]

2.2.2 Jenis-jenis Citra Digital

2.2.2.1 Citra Biner (Monokrom)

Banyaknya warna: 2, yaitu hitam dan putih. Dibutuhkan 1 bit dalam memori untuk menyimpan kedua warna ini. Gradasi warna:^[9]



Gambar 2.2 Citra Biner

2.2.2.2 Citra *Grayscale* (Skala Keabuan)

Banyaknya warna: tergantung pada jumlah bit yang disediakan dalam memori untuk menampung kebutuhan warna ini. Citra 2 bit mewakili 4 warna dengan gradasi warna sebagai berikut:^[9]



Gambar 2.3 Citra *Grayscale* 2 bit

Citra 3 bit mewakili 8 warna dengan gradasi warna sebagai berikut:



Gambar 2.4 Citra *Grayscale* 3 bit

2.2.2.3 Citra Warna (*True Color*)

Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RGB = *Red Green Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = byte, yang berarti setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak $28.28.28 = 224 = 16$ juta warna lebih. Itulah sebabnya format ini dinamakan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna alam.^[9]

2.2.3 Elemen-elemen Citra Digital

Berikut ini adalah elemen-elemen yang terdapat pada citra digital.^[9]

1. Kecerahan (*Brightness*)

Kecerahan (*Brightness*) merupakan intensitas cahaya yang dipancarkan piksel dari citra yang dapat ditangkap oleh sistem penglihatan. Kecerahan pada sebuah titik (piksel) di dalam citra merupakan intensitas rata-rata dari suatu area yang melingkupinya.

2. Kontras (*Contrast*)

Kontras (*Contrast*) menyatakan sebaran terang dan gelap dalam sebuah citra. Pada citra yang baik komposisi gelap dan terang tersebar secara merata.

3. Kontur (*Contour*)

Kontur (*Contour*) adalah keadaan yang ditimbulkan oleh perubahan intensitas pada piksel-piksel yang bertetangga. Karena adanya perubahan intensitas inilah mata mampu mendeteksi tepi-tepi objek di dalam citra.

4. Warna

Warna sebagai persepsi yang ditangkap sistem visual terhadap panjang gelombang cahaya yang dipantulkan oleh objek.

5. Bentuk (*Shape*)

Bentuk (*Shape*) adalah properti intrinsik dari objek 3 dimensi, dengan pengertian bahwa bentuk merupakan properti intrinsik utama untuk sistem visual manusia.

6. Tekstur (*Texture*)

Tekstur (*Texture*) dicirikan sebagai distribusi spasial dari derajat keabuan di dalam sekumpulan piksel-piksel yang bertetangga. Tekstur adalah sifat-sifat atau karakteristik yang dimiliki oleh suatu daerah yang cukup besar sehingga secara alami sifat-sifat tadi dapat berulang dalam daerah tersebut. Tekstur adalah keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital. Informasi tekstur dapat digunakan untuk membedakan sifat-sifat permukaan suatu benda dalam citra yang berhubungan dengan kasar dan halus, juga sifat-sifat spesifik dari kekasaran dan kehalusan permukaan tadi, yang sama sekali terlepas dari warna permukaan tersebut.

Berikut kegunaan analisis tekstur:¹⁶¹

- a. Tekstur memainkan peranan penting dalam banyak tugas pada sistem visual, seperti pemeriksaan permukaan, pengelompokan objek pemandangan, orientasi permukaan dan penentuan bentuk objek.
- b. Digunakan untuk segmentasi citra, mengidentifikasi pola-pola yang teratur dan berulang, pola-pola intensitas, permukaan benda yang berhubungan dengan sifat kasar dan halus, koloni mikroba, jalan raya, bahkan sampai pada sifat permukaan bumi atau planet lainnya.
- c. Untuk tujuan pengolahan citra, analisis tekstur adalah menjadikan pola variasi lokal intensitas yang berulang sebagai pembeda, manakala pola variasi tersebut terlalu kecil bila dibandingkan dengan objek yang diamati dalam resolusi yang dipakai.

Syarat terbentuknya tekstur adalah sebagai berikut :

- a. Adanya pola-pola primitif yang terdiri dari satu atau lebih piksel. Bentuk pola primitif ini dapat berupa titik, garis lurus, garis lengkung, luasan dan lain-lain yang merupakan elemen dasar dari sebuah bentuk.
 - b. Pola-pola primitif tadi muncul berulang-ulang dengan interval jarak dan arah tertentu sehingga dapat diprediksi atau ditemukan karakteristik pengulangannya.
-

2.2.4 Format Citra Digital

Citra Digital memiliki beberapa format yang memiliki karakteristik tersendiri. Format pada citra digital ini umumnya berdasarkan tipe dan cara kompresi yang digunakan pada citra digital tersebut. Ada dua format citra digital yang sering dijumpai, antara lain:⁽⁶⁾

1. Bitmap (BMP)

Merupakan format gambar yang paling umum dan merupakan format standard windows. Ukuran filenya sangat besar karena bisa mencapai ukuran megabyte. File ini merupakan format yang belum terkompresi dan menggunakan sistem warna RGB (*Red, Green, Blue*) di mana masing-masing warna pixel-nya terdiri dari 3 komponen R, G, dan B yang dicampur menjadi satu. File BMP dapat dibuka dengan berbagai macam software pembuka gambar seperti ACDSee, Paint, Irvan View dan lain-lain. File BMP tidak bisa (sangat jarang) digunakan di web (internet) karena ukurannya yang besar.

2. Joint Photographic Expert Group (JPEG/JPG)

Format JPEG merupakan format yang paling terkenal sampai sekarang ini. Hal ini karena sifatnya yang berukuran kecil (hanya puluhan/ratusan KB saja), dan bersifat portable. Format file ini sering digunakan pada bidang fotografi untuk menyimpan file foto hasil perekaman analog to digital converter (ADC). Karena ukurannya kecil maka file ini banyak digunakan di web (internet).

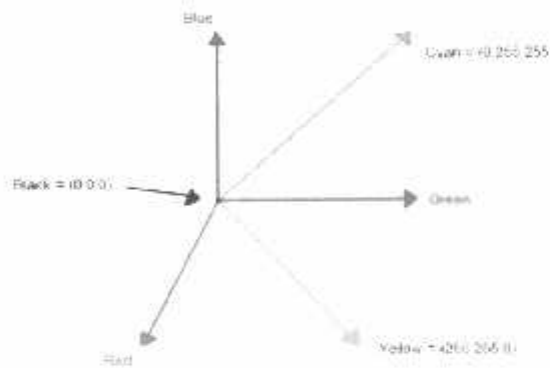
2.2.5 Representasi Warna

Metode yang paling mudah dan sederhana dalam merepresentasikan warna di dalam komputer adalah bitmap. Tipe file berformat seperti ini dapat diibaratkan dengan sebuah matriks yang terdiri dari baris-baris piksel. Dan setiap piksel dari baris tersebut merepresentasikan nilai tampilan warna piksel tersebut. Nilai tampilan warna tersebut disusun berdasarkan tiga warna dasar, yaitu: merah, biru, dan hijau. Metode ini dikenal dengan metode *RGB encoding*.⁽¹⁰⁾

Setiap satu warna dasar (merah, hijau atau biru) memiliki 256 level warna. Sehingga total warna yang dapat dikenali oleh komputer sekarang adalah $256 \times 256 \times 256$ atau 16,8 juta warna yang berbeda. Jumlah tersebut didapatkan dengan mengkombinasikan setiap level warna dasar yang berbeda.

Misalnya saja warna hitam dapat direpresentasikan ke dalam sistem *RGB* sebagai (0, 0, 0) yang berarti level warna merah 0, warna hijau 0, dan warna biru 0. Warna merah dapat direpresentasikan dengan (255, 0, 0) yang berarti level warna merah sebanyak 255, warna hijau sebanyak 0, dan warna biru sebanyak 0. Sedangkan warna putih direpresentasikan ke dalam sistem *RGB* menjadi (255, 255, 255).^[10]

Pada kebanyakan pemrosesan gambar membutuhkan properti-properti yang dimiliki oleh vektor, seperti normalisasi, rotasi dan proyeksi. Sehingga representasi warna dapat juga dilakukan dalam bentuk vektor. Berikut ini akan ditampilkan satu contoh vektor warna dalam ruang 3 dimensi.^[10]



Gambar 2.5 Vektor Representasi Warna

2.3 Metode Morphology

Morfologi adalah satu cabang dari pengolahan citra yang sangat bermanfaat dalam analisis bentuk dalam citra. Aplikasi utama dari morphology adalah pengekstrakan komponen citra yang berguna dalam representasi dan deskripsi bentuk.^[5]

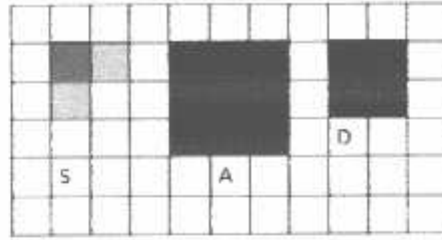
1. Erosi

Erosi adalah mengecilkan atau menipiskan obyek citra biner, berbeda dengan Dilasi yang melakukan penumbuhan atau penebalan. Erosi dapat di anggap sebagai operasi morphological filtering diman detail citra yang lebih kecil dari strel akan difilter (dihilangkan) dari citra.

Berikut persamaan sistematis operasi erosi :

$$E(A, S) = A \ominus S$$

Contoh hasil operasi Erosi :^[3]



Gambar 2.6 Contoh Proses Erosi

Keterangan :

E : Erosi

A : Citra Awal

S : Template

D : Citra Hasil Erosi

Contoh Matrik Erosi :

1 1

1 1

Template

0	0	0	0	0	0	0	0
0	0	2	4	4	2	0	0
0	0	4	8	8	4	0	0
0	0	2	4	4	2	0	0
0	0	0	0	0	0	0	0

Citra Awal

1	1	1	1	1	1	1	1
1	1	1	3	3	1	1	1
1	1	3	7	7	3	1	1
1	1	1	3	3	1	1	1
1	1	1	1	1	1	1	1

Citra Hasil

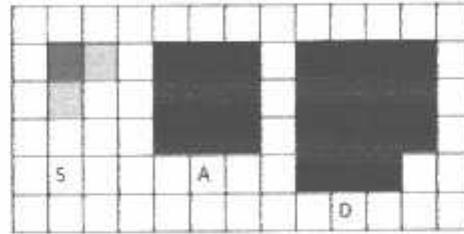
2. Dilasi

Operasi Dilasi adalah penumbuhan atau penebalan dalam citra biner. Dilasi sangat berguna ketika di terapkan dalam obyek-obyek yang terputus dikarenakan hasil pengambilan citra yang terganggu oleh noise, kerusakan obyek fisik yang dijadikan citra digital, atau disebabkan resolusi yang jelek.^[3]

Berikut persamaan sistematis operasi dilasi :^[3]

$$D(A, S) = A \oplus S$$

Contoh hasil operasi Dilasi :



Gambar 2.7 Contoh Proses Dilasi

Keterangan :

D : Dilasi

A : Citra Awal

S : Template

D : Citra Hasil Dilasi

Contoh Matrik Dilasi

$$\begin{array}{cc} 1 & 1 \\ 1 & 1 \end{array}$$

Template

0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	2	4	4	2	0	0	1	1	3	5	5	3	1	1	1
0	0	4	8	8	4	0	0	1	1	5	9	9	5	1	1	1
0	0	2	4	4	2	0	0	1	1	3	5	5	3	1	1	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1

Citra Awal

Citra Hasil

2.4 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal.^[12]



Gambar 2.8 Logo Java

Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (*bytecode*) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, "*Tulis sekali, jalankan di mana pun*". Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.^[12]

1. Kelebihan Java :

1. *Multiplatform*. Kelebihan utama dari Java ialah dapat dijalankan di beberapa *platform* / sistem operasi komputer, sesuai dengan prinsip *tulis sekali, jalankan di mana saja*. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin / *bytecode*) sekali lalu hasilnya

dapat dijalankan di atas beberapa platform tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas operating system Linux tetapi dijalankan dengan baik di atas Microsoft Windows. Platform yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebabnya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan *bytecode* tersebut.^[12]

2. OOP (*Object Oriented Programming* - Pemrogram Berorientasi Objek)
3. Perpustakaan Kelas Yang Lengkap, Java terkenal dengan kelengkapan *library*/perpustakaan (kumpulan program program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.
4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.^[12]
5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).^[12]

2. Kekurangan Java :

1. Tulis sekali, jalankan di mana saja - Masih ada beberapa hal yang tidak kompatibel antara *platform* satu dengan *platform* lain. Untuk J2SE, misalnya *SWT-AWT bridge* yang sampai sekarang tidak berfungsi pada Mac OS X.^[12]

2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan *bytecode* yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/direverse-engineer.^[12]
3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berlutut dengan mesin komputer berumur lebih dari 4 tahun.^[12]

2.5 NetBeans IDE 6.5

NetBeans adalah sebuah pengembangan terpadu (IDE) untuk mengembangkan program menggunakan Java, tetapi juga dengan bahasa lain, khususnya PHP, C / C. dan HTML5. Ini juga merupakan kerangka platform aplikasi untuk aplikasi desktop Java dan lain-lain. NetBeans IDE ditulis dalam Java dan dapat berjalan di Windows, OS X, Linux, Solaris dan platform lain yang mendukung JVM kompatibel. Platform NetBeans memungkinkan aplikasi untuk dikembangkan dari satu set modular komponen software yang disebut modul.^[13]



Gambar 2.9 Logo NetBeans

2.6 MySQL Versi 5.5.30

Suatu perangkat lunak database relasi (*Relational Database Management System* atau RDBMS). MySQL AB menyebut produknya sebagai database open source terpopuler di dunia. Berdasarkan riset dinyatakan bahwa di *platform* web, baik untuk kategori open source maupun umum, MySQL adalah database paling banyak dipakai. MySQL adalah salah satu dari sekian banyak sistem database, merupakan terobosan solusi yang tepat dalam aplikasi database. Didukung oleh ribuan atau bahkan jutaan komunitas pengguna internet yang siap membantu.^[14]

RDBMS (*Relasional Database Management System*) yang didistribusikan secara gratis dibawah lisensi GPL (General Publis Lisensi) dimana setiap orang bebas menggunakan MySQL. MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam database sejak lama, yaitu SQL (*Struktur Query Language*), yang merupakan sebuah konsep pengoperasian database, terutama^[14] untuk pemilihan seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Keuntungan menggunakan Database MySQL adalah sebagai berikut :^[14]

1. Portability

MySQL dapat berjalan stabil pada berbagai sistem operasi diantaranya Windows, Linux, Mac X Server, HP-UX dan masih banyak lagi.

2. Open Source

MySQL didistribusikan secara gratis (*Open Source*) dibawah lisensi GPL.

3. Multi User

MySQL dapat digunakan oleh beberapa User dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini memungkinkan database server MySQL.

4. Coloum Types

5. MySQL memiliki tipe kolom yang sangat kompleks seperti signed / unsigned, integer, float, double, char, varchar, text, date, blob, time, datetime, year, set dan enum.

6. Command dan functions

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah select dan where dalam Query.

7. Security

MySQL memiliki beberapa lapisan sekuritis seperti level sub netmask nama, dan izin akses user dengan sistem perizinan yang mendetail dan password.

8. Scalibility and Limits

MySQL dapat menangani database dalam skala besar, dengan jumlah record lebih dari 50 juta dan 60 ribu tabel serta miliar baris.

9. Struktur Tabel

MySQL mewakili struktur tabel yang lebih fleksibel dalam menangani alter table, dibandingkan dengan database lainnya.

BAB III PERANCANGAN APLIKASI

3.1 Perancangan Sistem

Perancangan sistem yang dibuat adalah mengenai proses menghitung jumlah pengunjung pada Mall, dimana proses menghitung jumlah pengunjung Mall akan dilakukan berdasarkan gambar yang di ambil oleh camera dan bentuk citra yang di inputkan ke dalam sistem.

3.2 Desain Sistem

Demi mencapai hasil yang maksimal maka dibuat suatu desain, dimana desain sistem dibuat untuk mengetahui bagaimana aplikasi ini akan berjalan, adapun konsep-konsep dari desain aplikasi penghitung jumlah pengunjung Mall di tunjukan pada gambar 3.1:

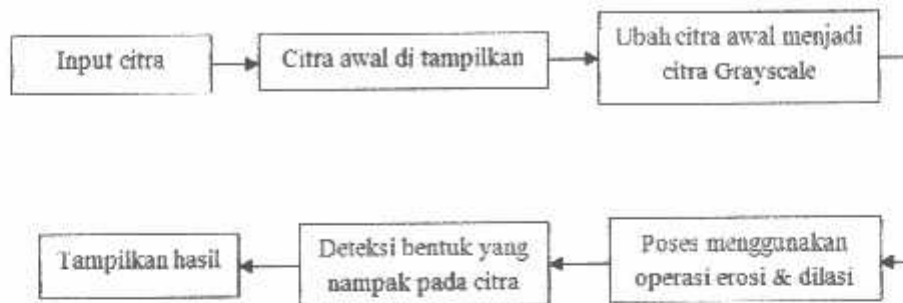


Gambar 3.1 Konsep dasar aplikasi penghitung jumlah pengunjung pada Mall

Dari gambar 3.1 dapat dijelaskan secara umum dan fungsi dari masing-masing gambar sebagai berikut :

1. Kamera mengambil gambar pengunjung yang masuk maupun keluar pada Mall.
2. Gambar yang telah diambil oleh kamera kemudian di proses menggunakan aplikasi yang di buat.
3. Dari hasil proses yang dilakukan oleh aplikasi kemudian aplikasi akan menampilkan jumlah pengunjung yang telah di hitung.

Secara umum aplikasi penghitung jumlah pengunjung pada Mall ini memiliki tahapan seperti gambar 3.2 :



Gambar 3.2 Alur proses aplikasi penghitung jumlah pengunjung pada Mall

Berdasarkan gambar 3.2 diatas, alur proses aplikasi penghitung jumlah pengunjung pada Mall menunjukkan bahwa inputan yang akan diolah berupa sebuah citra kemudian citra tersebut di ubah dari awalnya citra berwarna menjadi citra biner atau citra keabu-abuan. Proses selanjutnya dari citra tersebut di lakukan pendeteksian tepi agar dapat diketahui dengan jelas garis batas antar citra. Dan dilanjutkan dengan proses pendeteksian bentuk pada citra untuk mendapatkan bentuk citra yang sesuai dengan parameter yang telah diberikan. Dari proses yang telah dijelaskan di atas, aplikasi tersebut memberikan hasil akhir berupa klasifikasi citra yang memenuhi parameter.

3.3 Perancangan Aplikasi Sistem

Tujuan dari perancangan secara umum adalah untuk memberikan gambaran secara umum tentang aplikasi penghitung jumlah pengunjung pada Mall. Karena aplikasi penghitung jumlah pengunjung pada Mall ini dimulai dari menentukan fakta awal terlebih dahulu sebelum menemukan sebuah hasil atau kesimpulan. Maka perancangan yang akan dibuat pada aplikasi penghitung jumlah pengunjung Mall ini menggunakan beberapa form yang berkesinambungan agar mendapat tampilan aplikasi yang baik sesuai dengan yang diharapkan.

3.4 Perancangan Diagram Alir

Perancangan diagram alir akan menjelaskan bagaimana proses dan urutan dari aplikasi yang di buat seperti pada gambar :



Gambar 3.3 Diagram Alir Program Aplikasi

Keterangan Gambar 3.3 Diagram Alir Program Aplikasi :

1. Memulai proses.
2. Menginputkan citra yang akan diproses.
3. Menampilkan citra awal setelah diinputkan.
4. Mengubah citra awal RGB menjadi citra grayscale.
5. Deteksi bentuk citra menggunakan metode Morphology.
6. Menampilkan hasil setelah dilakukan proses Morphology.
7. Program selesai.

3.5 Proses Pengolahan Citra

Pengolahan citra merupakan proses pengolahan dan analisa citra. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Istilah pengolahan citra secara umum di definisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang di wakili oleh bit-bit tertentu. Citra digital dapat di definisikan sebagai fungsi dua variabel $f(x,y)$, dimana x dan y adalah keordinat special dan nilai $f(x,y)$ adalah intensitas citra pada koordinat tersebut.

3.5.1 Mengubah Citra RGB Menjadi Citra Grayscale

Mengubah citra awal yang berupa citra berwarna menjadi citra Grayscale adalah pengolahan citra digital membutuhkan memori yang jauh lebih besar dibanding pengolahan teks pada komputer. Untuk menghemat kebutuhan memori, warna citra diubah ke tingkat grayscale terendah yang berarti citra hanya mengandung informasi hitam dan putih pada piksel-piksel penyusunnya, sehingga operasi dapat tetap dijalankan walaupun memori yang tersedia relatif kecil. Selain itu, sifat-sifat pengolahan citra biner umumnya lebih sederhana, cepat dan mudah diimplementasikan.

3.6 Proses Perhitungan Jumlah Pengunjung

Dalam proses ini ada beberapa operasi yang di lakukan untuk menghasilkan citra yang sesuai dengan yang di harapkan dan juga melakukan proses perhitungan dengan operasi yang di pakai sebagai berikut :

1. Erosi

Erosi adalah mengecilkan atau menipiskan obyek citra biner, berbeda dengan Dilasi yang melakukan penumbuhan atau penebalan. Erosi dapat di anggap sebagai operasi morphological filtering diman detail citra yang lebih kecil dari strel akan difilter (dihilangkan) dari citra.

Berikut persamaan sistematis operasi erosi :

$$E(A, S) = A \ominus S$$

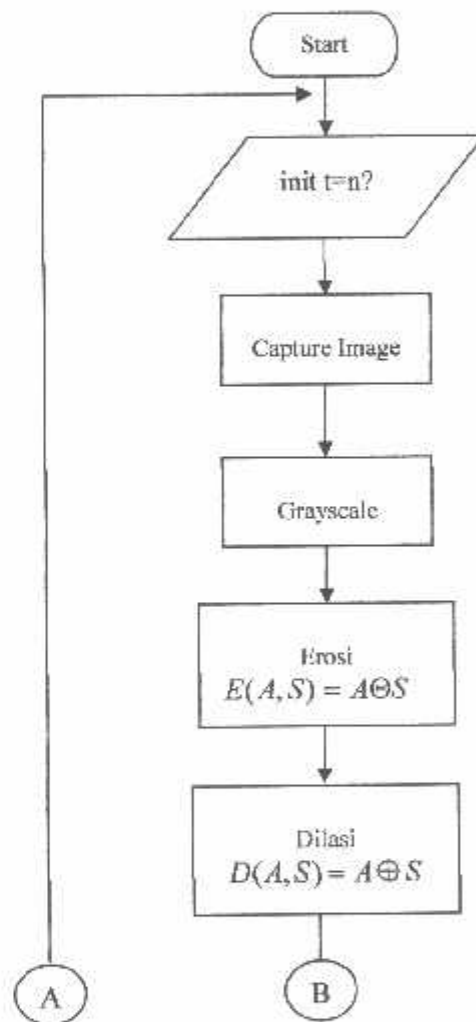
2. Dilasi

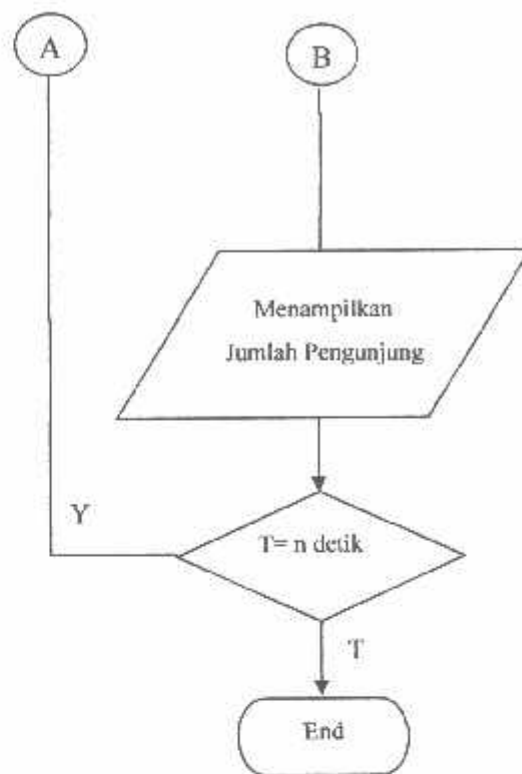
Operasi Dilasi adalah penumbuhan atau penebalan dalam citra biner. Dilasi sangat berguna ketika di terapkan dalam obyek-obyek yang terputus dikarenakan hasil pengambilan citra yang terganggu oleh noise, kerusakan obyek fisik yang dijadikan citra digital, atau disebabkan resolusi yang jelek.

Berikut persamaan sistematis operasi dilasi :

$$D(A,S) = A \oplus S$$

Berikut adalah flowchart sistem aplikasi perhitungan jumlah pengunjung :





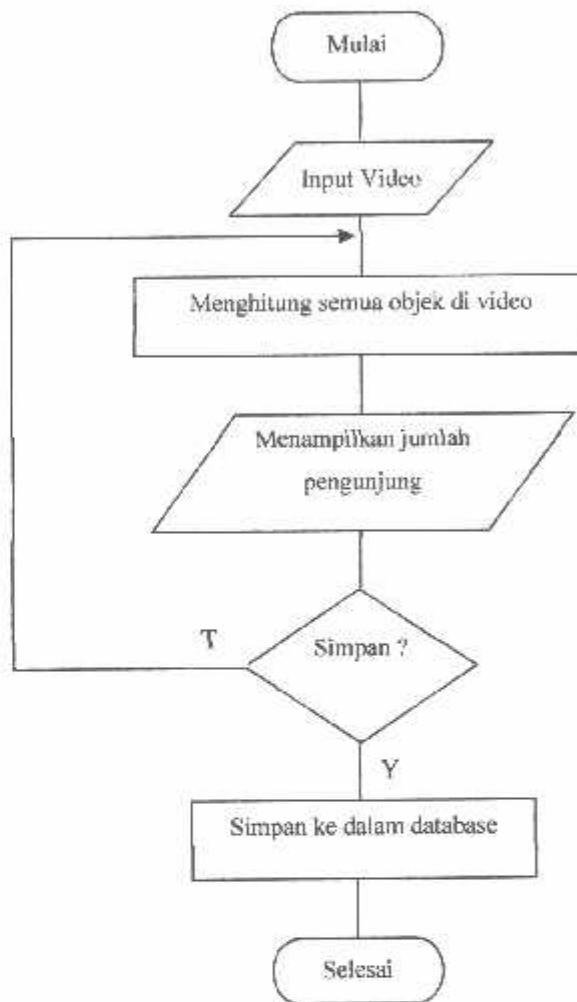
Gambar 3.4 Flowchart Sistem Aplikasi

Keterangan Gambar 3.4 Flowchart Sistem Aplikasi :

1. Memulai Program.
2. Pengguna menginputkan waktu capture.
3. Capture image sesuai dengan waktu yang telah ditentukan pengguna.
4. Citra awal dirubah menjadi grayscale.
5. Proses Dilasi, dilakukan jika image yang dicapture terlalu kecil maka proses dilasi akan memperbesar image tersebut.
6. Proses Erosi, kebalikan dari proses Dilasi jika image yang dicapture terlalu besar maka image tersebut akan diperkecil menggunakan proses Erosi.
7. Setelah dilakukan proses menggunakan Metode Morphology maka jumlah pengunjung yang telah terhitung akan ditampilkan dalam program sesuai dengan jumlah pengunjung yang telah terhitung.

8. Jika akan menghitung lagi maka proses akan mengcapture image sesuai dengan waktu capture yang ditentukan pengguna, jika tidak maka program selesai.
9. Program selesai.

Berikut adalah flowchart proses simpan ke database :



Gambar 3.5 Flowchart Penyimpanan ke Database

Keterangan Gambar 3.5 Flowchart Penyimpanan ke Database :

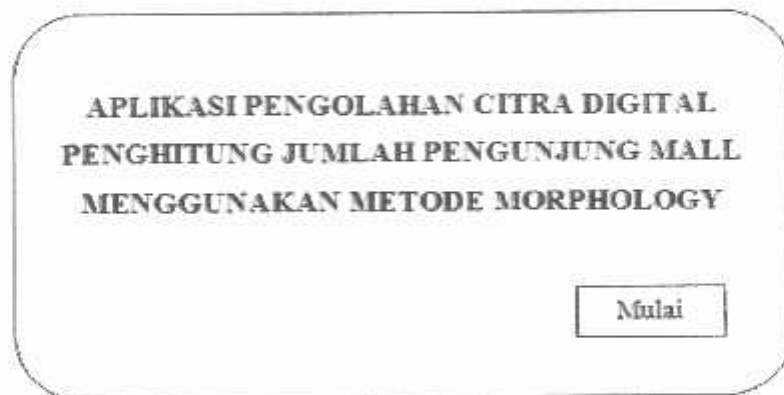
1. Memulai program.
2. Inputkan video yang akan dihitung jumlah pengunjung.
3. Menghitung objek yang ada di dalam video.
4. Menampilkan jumlah pengunjung yang telah dihitung.
5. Jika di simpan maka jumlah pengunjung akan disimpan kedalam database secara manual dan program selesai.
6. Jika tidak disimpan ke dalam database maka program akan mengulang proses perhitungan jumlah pengunjung.
7. Program selesai.

3.7 Perancangan Interface

Perancangan interface merupakan rancang bangun dari interaksi user dengan computer. Interaksi ini dapat berupa proses penginputan data kesistem, pemilihan menu dan menjalankan aplikasi.

3.7.1 Tampilan Form Awal

Pada aplikasi penghitung jumlah pengunjung Mall ini akan menampilkan form awal seperti pada gambar di bawah ini.



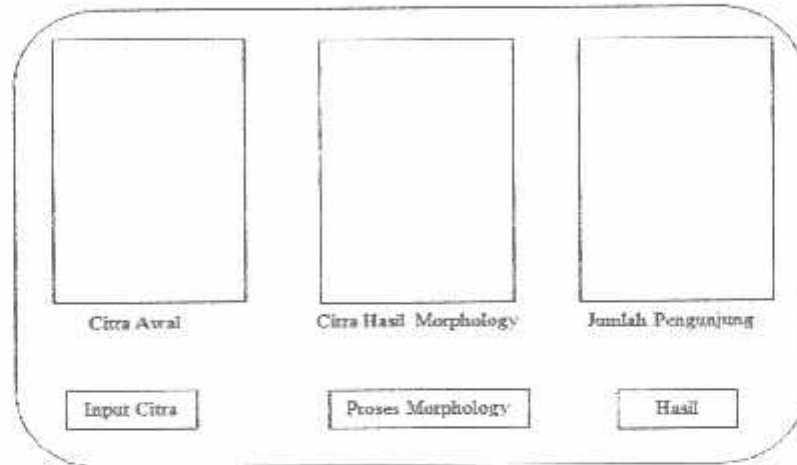
Gambar 3.6 Tampilan Form Awal Aplikasi

Keterangan Gambar 3.6 Tampilan Form Awal Aplikasi :

Rancangan tampilan awal aplikasi pengolahan citra digital penghitung jumlah pengunjung mall menggunakan metode morphology.

3.7.2 Tampilan Form Kedua

Setelah menekan button mulai pada form awal, maka akan muncul form kedua yaitu form proses morphology dan perhitungan jumlah pengunjung. Pada perancangan form kedua ini, user akan menginputkan citra dan mulai melakukan proses pengubahan citra berwarna menjadi citra biner dan kemudian dilakukan proses morphology pada citra. Untuk lebih jelasnya, dapat dilihat pada gambar di bawah ini.



Gambar 3.7 Tampilan Form Kedua

Keterangan Gambar 3.7 Tampilan Form Kedua :

1. Tampilan saat menginputkan citra.
2. Tampilan saat dilakukan proses Morphology.
3. Tampilan hasil setelah melakukan proses perhitungan pengunjung.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Implementasi sistem adalah proses penerapan rancangan sistem yang telah dibuat menjadi suatu aplikasi yang bisa dijalankan pada kenyataannya. Implementasi sistem berfungsi untuk menerapkan sistem sesuai dengan tujuan sistem. Disamping implementasi berfungsi untuk menerapkan sistem, fungsi lainnya adalah untuk mengetahui sejauh mana keberhasilan dari rancangan yang telah dibuat. Implementasikan sistem aplikasi penghitung jumlah pengunjung ini memiliki beberapa kriteria yang harus mendukung terwujudnya implementasi sistem.

4.2 Spesifikasi Perangkat Lunak

Implementasian sistem aplikasi pengolahan citra digital penghitung jumlah pengunjung mall menggunakan metode morphology, software yang digunakan adalah sebagai berikut:

a. Sistem Operasi Windows7

Sistem operasi yang digunakan untuk membuat aplikasi pengolahan citra digital penghitung jumlah pengunjung mall menggunakan metode morphology menggunakan sistem operasi Windows7. Sistem operasi Windows7 berfungsi untuk pembuatan aplikasi yang di buat.

b. NetBeans IDE 6.5

NetBeans adalah sebuah pengembangan terpadu (IDE) untuk mengembangkan program menggunakan Java, tetapi juga dengan bahasa lain, khususnya PHP, C / C, dan HTML5. Ini juga merupakan kerangka platform aplikasi untuk aplikasi desktop Java dan lain-lain. NetBeans IDE ditulis dalam Java dan dapat berjalan di Windows, OS X, Linux, Solaris dan platform lain yang mendukung JVM kompatibel.

c. MySQL Versi 5.5.30

Suatu perangkat lunak database relasi (*Relational Database Management System* atau RDBMS). MySQL AB menyebut produknya sebagai database open source terpopuler di dunia. Berdasarkan riset

dinyatakan bahwa di *platform* web, baik untuk kategori open source maupun umum, MySQL adalah database paling banyak dipakai. MySQL adalah salah satu dari sekian banyak sistem database, merupakan terobosan solusi yang tepat dalam aplikasi database. Didukung oleh ribuan atau bahkan jutaan komunitas pengguna internet yang siap membantu.

d. Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal.

4.3 Pengujian Sistem

Pengujian sistem merupakan tahap uji coba terhadap sistem yang telah dibuat. Pengujian sistem dilakukan bertujuan untuk mengidentifikasi masalah pada sistem bila terjadi kesalahan pada sistem dan bertujuan untuk mengetahui sejauh mana keberhasilan aplikasi ini bisa diterapkan dalam sistem. Beberapa prosedur pengujian sistem yang ditetapkan sebagai berikut:

- a. Buka aplikasi NetBeans 6.5 IDE
- b. Jalankan file aplikasi program MATOS yang ada pada Net Beans 6.5 IDE.
- c. Klik Run Project untuk menjalankan program.

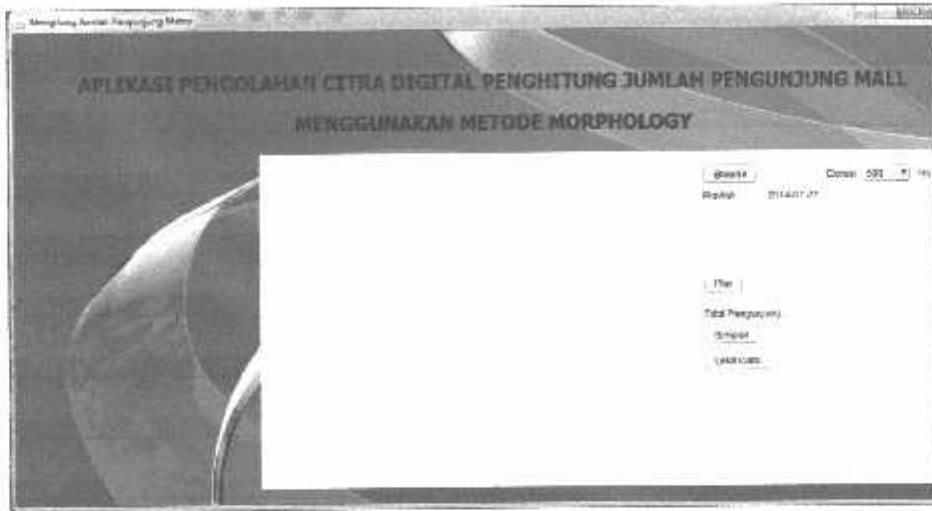
Hasil pengujian sistem telah cukup berhasil dalam menampilkan data dan mengolah data pada aplikasi pengolahan citra digital penghitung jumlah pengunjung mall menggunakan metode morphology.

4.4 Halaman Awal Perhitungan Pengunjung Secara Manual

Halaman awal merupakan halaman yang akan muncul pertama kali pada saat user mengakses aplikasi ini. Pada halaman awal terdapat menu-menu yang telah disediakan untuk user.

4.4.1 Tampilan Halaman Menu Utama

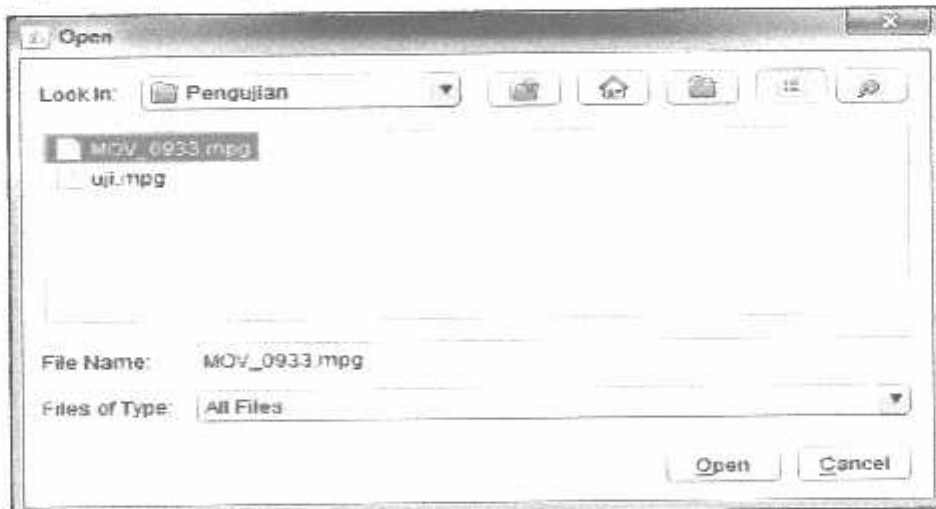
Gambar 4.1 Tampilan menu utama pada aplikasi pengolahan citra digital penghitung jumlah pengunjung mall menggunakan metode morphology adalah tampilan pertama saat menjalankan program.



Gambar 4.1 Halaman Utama Aplikasi Penghitung Jumlah Pengunjung

4.4.2 Tampilan Halaman Menu Browse

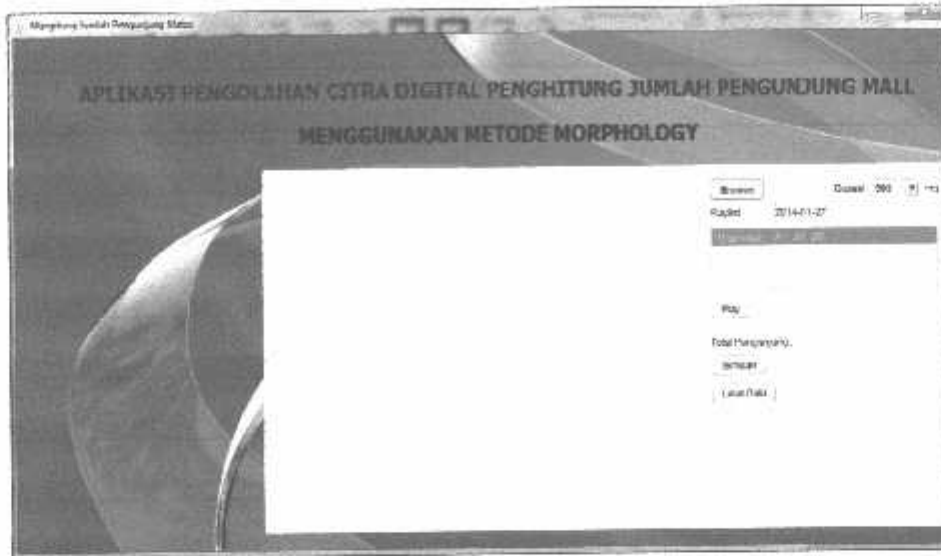
Gambar 4.2 Merupakan tampilan form untuk memilih file yang ingin diproses dilakukan perhitungan jumlah pengunjung.



Gambar 4.2 Halaman Saat Menekan Tombol Browse

4.4.3 Tampilan Halaman Menu Setelah Browse

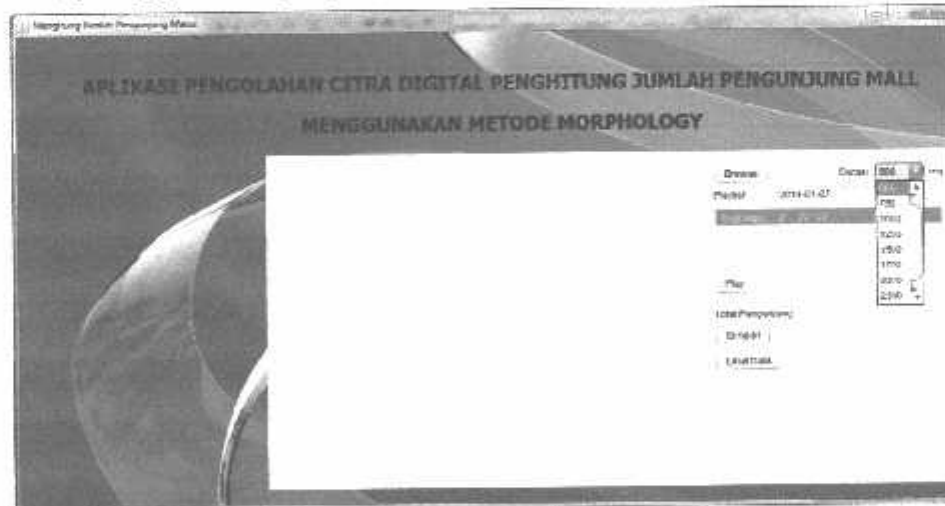
Gambar 4.3 merupakan proses mengambil video yang akan dihitung jumlah pengunjungnya.



Gambar 4.3 Halaman Setelah Memilih File Video

4.4.4 Tampilan Halaman Menu Untuk Memilih Durasi

Gambar 4.4 merupakan tampilan untuk memilih durasi yang sesuai untuk menghitung jumlah objek yang bergerak.



Gambar 4.4 Halaman Untuk Memilih Durasi Untuk Menghitung Jumlah Pengunjung

4.4.5 Tampilan Halaman Menu Play

Gambar 4.5 merupakan proses video perhitungan jumlah pengunjung.

Keterangan :

Lingkaran warna merah merupakan hasil pengunjung yang terhitung.



Gambar 4.5 Halaman Penghitung Jumlah Pengunjung

4.4.6 Tampilan Halaman Menu Simpan

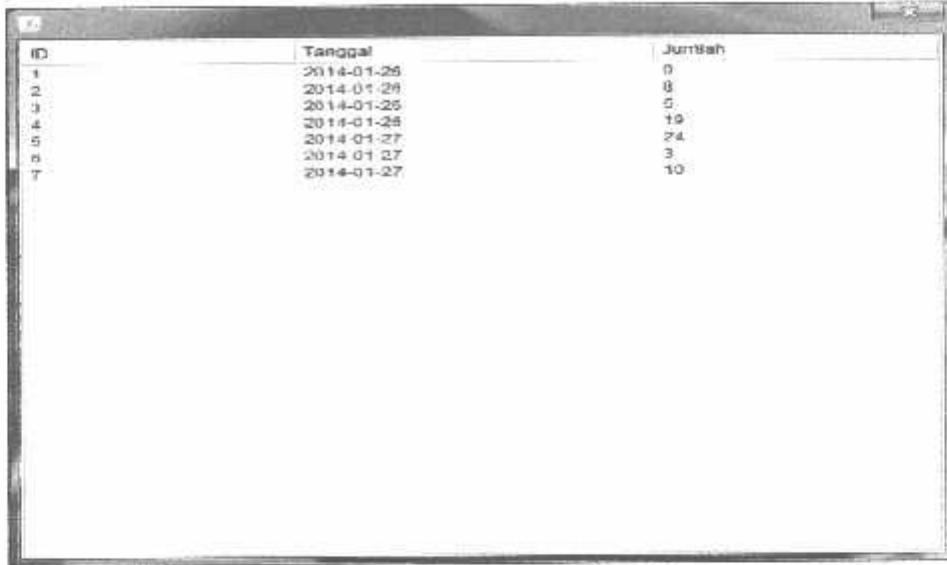
Pada gambar 4.6 merupakan proses penyimpanan jumlah pengunjung



Gambar 4.6 Halaman Menu Simpan

4.4.7 Tampilan Halaman Menu Lihat Data

Pada gambar 4.7 merupakan tampilan jumlah pengunjung yang disimpan kedalam Database MySQL. sesuai dengan tanggal dan jumlah pengunjung yang telah terhitung.



ID	Tanggal	Jumlah
1	2014-01-26	0
2	2014-01-26	8
3	2014-01-26	5
4	2014-01-26	19
5	2014-01-27	24
6	2014-01-27	3
7	2014-01-27	10

Gambar 4.7 Halaman Menu Lihat Data

4.5 Halaman Awal Perhitungan Pengunjung Secara Realtime

Aplikasi yang penulis buat bias melakukan perhitungan secara otomatis atau realtime menggunakan kamera yang dihubungkan ke aplikasi yang telah dibuat.

4.5.1 Halaman Awal Saat Program Jalan

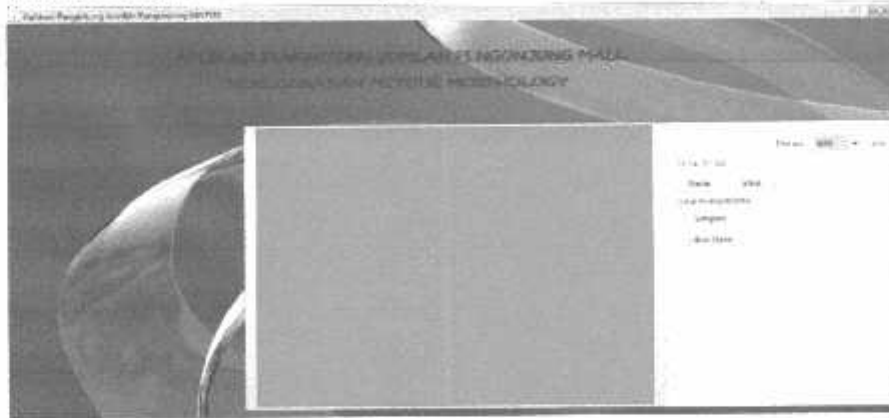
Gambar 4.8 merupakan tampilan saat program dijalankan maka akan muncul tampilan seperti dibawah ini, pengguna harus memilih nama kamera yang digunakan untuk menghitung jumlah pengunjung



Gambar 4.8 Halaman Memilih Perangkat Kamera Yang Digunakan

4.5.2 Halaman Setelah Memilih Nama Perangkat Kamera

Gambar 4.9 merupakan tampilan utama setelah memilih nama perangkat kamera, dihalaman ini ada pilihan untuk memilih waktu capture, tombol mulai untuk memulai perhitungan dan tombol stop untuk berhenti dari proses perhitungan, tombol simpan untuk menyimpan jumlah pengunjung yang telah terhitung, tombol lihat data untuk melihat data pengunjung yang telah tersimpan di database MySQL.

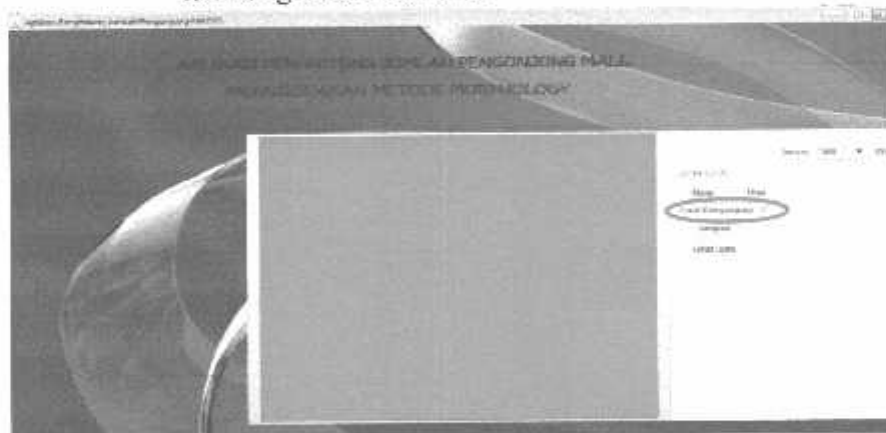


Gambar 4.9 Halaman Setelah Memilih Nama Perangkat Kamera

4.5.3 Halaman Perhitungan Pengunjung Secara Realtime

Gambar 4.10 merupakan tampilan utama saat melakukan perhitungan pengunjung secara realtime.

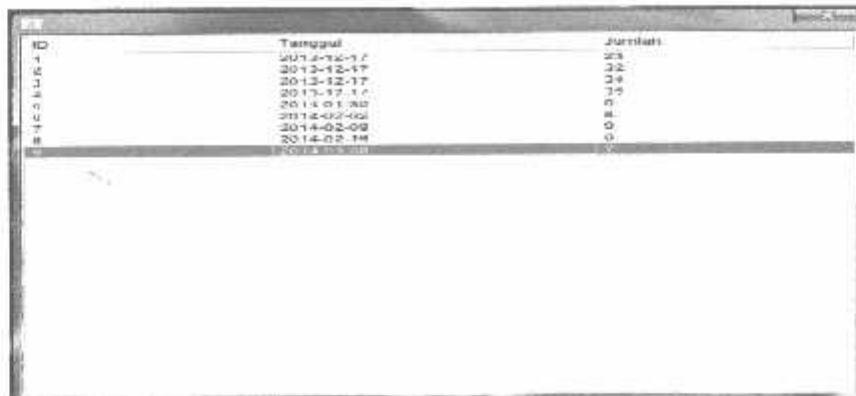
Keterangan : lingkaran berwarna merah adalah jumlah pengunjung yang telah terhitung secara realtime.



4.10 Halaman Perhitungan Pengunjung Secara Realtime

4.5.4 Halaman Untuk Melihat Jumlah Pengunjung Yang Tersimpan

Gambar 4.11 pengguna dapat melihat jumlah pengunjung yang telah tersimpan didatabase MySQL. sesuai tanggal saat melakukan perhitungan pengunjung.



ID	Tanggal	Jumlah
1	2012-12-17	23
2	2013-12-17	22
3	2013-12-17	24
4	2013-12-17	25
5	2014-01-30	0
6	2014-02-06	8
7	2014-02-09	0
8	2014-02-14	0

4.11 Halaman Untuk Melihat Jumlah Pengunjung Yang Tersimpan

4.6 Pengujian Fungsional Sistem

Pengujian yang dilakukan pada Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pengunjung Mall Menggunakan Metode Morphology yang dilakukan dengan menguji aplikasi yang telah dibuat, apakah aplikasi ini berjalan sesuai dengan yang diharapkan atau tidak. Pengujian dilakukan dengan menggunakan metode pengujian *blackbox*. Seperti pada tabel 4.1 :

Tabel 4.1 Pengujian Melalui Metode Blackbox :

NO	FUNGSI	WINDOWS7		LINUX VERSI 10.4	
		YA	TIDAK	YA	TIDAK
1.	Browse	✓	×	✓	×
2.	Durasi	✓	×	✓	×
3.	Hitung Pengunjung	✓	×	✓	×
4.	Play	✓	×	✓	×
5.	Simpan	✓	×	✓	×
6.	Lihat Data	✓	×	✓	×

Keterangan :

x : Gagal

✓ : Berhasil

4.7 Pengujian Aplikasi Menggunakan Metode Morphology

Tabel 4.2 pengujian objek yang tertangkap camera menggunakan metode morphology :

Tabel 4.2 Pengujian Menggunakan Metode Morphology :

No.	Nama File Video	Jenis File Video	Ukuran File Video	Total Pengunjung	Hasil Deteksi		Hasil Percobaan	Terdeteksi	Tidak Terdeteksi
					Terhitung	Tidak Terhitung			
1.	Pengunjung1	Mpeg	13.4 MB	40 Pengunjung	29	11	Semua objek dianggap manusia	✓	-
2.	Pengunjung2	Mpeg	41.3 MB	25 Pengunjung	24	1	Manusia membawa keranjang, tas, dompet, bawahan	✓	-
3.	Pengunjung3	Mpeg	2.54 MB	9 Pengunjung	5	4	Manusia berdiri lebih dari waktu capture	✓	-
4.	Pengunjung4	Mpeg	8.11 MB	25 Pengunjung	24	1	Manusia yang sedang berdekatan	✓	-
5.	Pengunjung5	Mpeg	41.3 MB	14 Pengunjung	21	0	Jarak kamera dan objek 1 meter dan gerak kamera dibagian atas	✓	-

Keterangan :

x : Tidak Terdeteksi

✓ : Terdeteksi.

Keterangan Tabel 4.2 Pengujian Menggunakan Metode Morphology :

1. Total pengunjung adalah 40 pengunjung, yang terhitung menggunakan aplikasi yang dibuat ada 29 pengunjung dan 11 pengunjung tidak terhitung oleh aplikasi.
2. Total pengunjung adalah 25 pengunjung, yang terhitung menggunakan aplikasi yang dibuat ada 24 pengunjung dan 1 pengunjung tidak terhitung oleh aplikasi.
3. Total pengunjung adalah 9 pengunjung, yang terhitung menggunakan aplikasi yang dibuat ada 5 pengunjung dan 4 pengunjung tidak terhitung oleh aplikasi.
4. Total pengunjung adalah 25 pengunjung, yang terhitung menggunakan aplikasi yang dibuat ada 24 pengunjung dan 1 pengunjung tidak terhitung oleh aplikasi.
5. Total pengunjung adalah 14 pengunjung, yang terhitung menggunakan aplikasi yang dibuat ada 21 pengunjung dan 0 pengunjung tidak terhitung oleh aplikasi.

6. Berdasarkan pengujian yang telah dilakukan jarak antara objek dan kamera 1 Meter dan camera diletakkan di atas objek yang akan dihitung.
7. Pada pengujian No 5 jumlah pengunjung yang terhitung lebih dari total pengunjung yang ada di karenakan aplikasi yang dibuat mengenal semua objek sebagai manusia seperti tas dorong, keranjang dan hewan.

Penjelasan Tabel 4.2 Pengujian Menggunakan Metode Morphology :

1. Pada file pengunjung1 jenis file yang digunakan adalah mpeg dan memiliki ukuran file 18,4 MB, total pengunjung pada file pengunjung1 berjumlah 40 pengunjung tetapi hanya terhitung 29 pengunjung dan 11 pengunjung tidak terhitung, dan dari hasil pengujian semua objek dianggap sebagai manusia.
 2. Pada file pengunjung2 jenis file yang digunakan adalah mpeg dan memiliki ukuran file 41,3 MB, total pengunjung pada file pengunjung2 berjumlah 25 pengunjung tetapi hanya terhitung 24 pengunjung dan 1 pengunjung tidak terhitung, dan dari hasil pengujian manusia yang membawa keranjang atau tas dorong dan membawa hewan akan terhitung.
 3. Pada file pengunjung3 jenis file yang digunakan adalah mpeg dan memiliki ukuran file 3,54 MB, total pengunjung pada file pengunjung3 berjumlah 9 pengunjung tetapi hanya terhitung 5 pengunjung dan 4 pengunjung tidak terhitung, dan dari hasil pengujian manusia yang berdiri terlalu lama dari waktu capture akan terhitung.
 4. Pada file pengunjung4 jenis file yang digunakan adalah mpeg dan memiliki ukuran file 8,11 MB, total pengunjung pada file pengunjung4 berjumlah 25 pengunjung tetapi hanya terhitung 24 pengunjung dan 1 pengunjung tidak terhitung, dan dari hasil pengujian manusia yang saling berdekatan akan terhitung.
 5. Pada file pengunjung5 jenis file yang digunakan adalah mpeg dan memiliki ukuran file 41,3 MB, total pengunjung pada file pengunjung5 berjumlah 14 pengunjung tetapi hanya terhitung 21 pengunjung dan 0 pengunjung tidak terhitung, dan dari hasil pengujian jarak kamera dan objek 1 meter dan letak kamera dibagian atas objek.
-

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil pengujian Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pegunjung Mall Menggunakan Metode Morphology maka dapat disimpulkan bahwa:

1. Aplikasi dapat berjalan di Sistem Operasi Windows7 dan Linux Versi 10.4.
2. Aplikasi melakukan perhitungan saat objek bergerak melewati kamera yang telah terpasang.
3. Aplikasi tidak bisa membedakan objek yang tertangkap kamera seperti tas dorong, hewan.
4. Objek yang berdiri lebih dari waktu capture akan dihitung. Dan objek yang berdiri saling berdekatan dihitung satu.

5.2 Saran

1. Perlu dikembangkan dengan jaringan syaraf tiruan single layer yang dapat memberikan output berupa keterangan pada objek yang dicapture
2. Dapat ditambahkan alat pengukur suhu pada aplikasi ini untuk membedakan manusia dan hewan.
3. Perlu adanya perbaikan tampilan pada warna dan desain aplikasi agar lebih dinamis.
4. Perlu ditambahkan face detection yang bisa membedakan objek masuk dan keluar.

DAFTAR PUSTAKA

- [1] Sutoyo. T , Mulyanto Edy, Subartono. Vincent, Dwi Nurhayati Oky, Wijanarko, "*Teori Pengenalan Citra Digital*", Andi Yogyakarta dan UDINUS Semarang, 2009.
- [2] Ca. Wijaya Marvin, Prijono Agus, "*Pengolahan Citra Digital Menggunakan MATLAB*", Informatika Bandung, 2007.
- [3]. Prasetio, Eko, Pemrosesan Morfologi Citra.
- [4] Sugiharto, Aris, "*Pemrograman GUI dengan MATLAB*", Andi, Yogyakarta, 2006.
- [5] Gonzales & Woods, *Teori Pengolahan Citra Digital*, 1992.
- [6] Usman ahmad, 2005, *Pengolahan Citra Digital & Teknik Pemrogramannya*, Graha ilmu.
- [7] Achmad Basuki, Josua F. Palandibdan Fatchurrochman, 2005, *Pengolahan Citra Digital Menggunakan Visual Basic*, Graha Ilmu.
- [8] Aniati murni Arymurthy & Suryana Setiawan, 1992, *Pengantar Pengolahan Citra*, Elex Media Komputindo.
- [9] Jain, Anil K., 1989, *Fundamentals Of Digital Image Processing*, Prentice Hall international.
- [10]Rafael C. Gonzales dan Richart E. Woods, 2008, *Digital Image Processing*, Prentice Hall.
- [11]Rafael C. Gonzales dan Richart E. Woods, 2004, *Digital Image Processing Using MATLAB*, Prentice Hall.
- [12]Kadir, Abdul. 2003, *Dasar Pemrograman Java 2*. Yogyakarta : Andi.
- [13]Hartati, G. Sri.2008. *Pemrograman GUI Swing Java dengan NetBeans*. Yogyakarta : Penerbit Andi.
- [14]Saputro, W, T, 2005, *MySQL Untuk Pemula*, Pena Media, Yogyakarta.

- [15] Rizal Ismanto R, Hidayatno Achmad, dan Adi Khrisna Dika. "*Identifikasi Objek Berdasarkan Bentuk dan Ukuran*". Tersedia : www.undip.ac.id (diakses tanggal 10 Nopember 2013)
- [16] <http://www.isomwebs.net/2012/09/pengertian-netbeans/> (di akses tanggal 30 Nopember 2013)
- [17] Supriatna Somantri Agus, Miskiyah dan Broto Wisnu. "*Identifikasi Mutu Fisik Menggunakan Pengolahan Citra Digital dan Jaringan Syaraf Tiruan*". tersedia : www.scribd.com (diakses tanggal 5 September 2013).
-

LAMPIRAN

#CODING METODE MORFOLOGY

```
package morfologi;

import java.sql.*;
import java.awt.BorderLayout;
import java.io.*;
import java.util.ArrayList;
import javax.media.*;
import javax.swing.*;
import java.awt.Component;
import java.awt.Dimension;
import java.awt.Toolkit;
import java.util.Date;
//import java.util.Timer;
import java.util.TimerTask;

import java.awt.AWTException;
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Graphics2D;
import java.awt.Image;
import java.awt.Rectangle;
import java.awt.Robot;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.image.BufferedImage;

import java.io.File;

import java.net.MalformedURLException;
```



```

import java.net.URL;
import java.text.SimpleDateFormat;
import javax.imageio.ImageIO;
    import javax.swing.JFileChooser;
import javax.swing.Timer;
public class FITama extends javax.swing.JFrame {
    JFileChooser fileChooser = new JFileChooser();
    ArrayList listVideo = new ArrayList();
    ArrayList listVideo1 = new ArrayList();
    // Timer timer = new Timer();
    ActionListener listener =
new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent e) {
            //throw new UnsupportedOperationException("Not supported
yet.");
            if(posisistart==0){
                BufferedImage test=Image();
                int type = test.getType() == 0?
BufferedImage.TYPE_INT_ARGB : test.getType();
                imagelama=resizeImage(test, type);
                posisistart++;
            }else if(posisistart==1){
                BufferedImage test=Image();
                int type = test.getType() == 0?
BufferedImage.TYPE_INT_ARGB : test.getType();

                imagebaru=resizeImage(test, type);
                posisistart++;
            }
        }
    };
}

```

```

        }else{
            int item=Erosi_Dilasi(imagelama, imagebaru);
            if(item>50){
                JumlahPengunjung++;
            }

            LTot.setText("Total Pengunjung :
"+(JumlahPengunjung));

        }

    }

};

private void center(){
    Dimension dimension =
Toolkit.getDefaultToolkit().getScreenSize();
    Dimension d = this.getSize();
    if(d.height>dimension.height){
        d.height=dimension.height;
    }
    if(d.width>dimension.width){
        d.width=dimension.width;
    }

    this.setLocation((dimension.width-
d.width)/2, (dimension.height-d.height)/2);
    // this.setBounds((dimension.width-
d.width)/2, (dimension.height-d.height)/2, 600, 300);

    :
    Koneksi koneksi;
    /** Creates new form FUtama */

```

```

public FUtana() {
    initComponents();
    center();
    koneksi = new Koneksi();
    Date d=new Date();
    SimpleDateFormat fzmt1 = new SimpleDateFormat("yyyy-MM-dd");
    String dateString1=fzmt1.format(d);
    txtTgl.setText(dateString1);
    jDateChooser1.setDate(d);
    s_jam.setMinimum(0);
    s_jam.setMaximum(23);
    s_menit.setMinimum(0);
    s_menit.setMaximum(59);
    s_jam.setValue(d.getHours());
    s_menit.setValue(d.getMinutes());setResizable(false);

}

void save(String tgl, int jumlah){

        try {
            // BufferedImage teImage = ImageIO.read(new
File(alamatFile));
            Connection conn=koneksi.getConnection();

            PreparedStatement pStatement = null;
            String sql ="insert into data " +
                "(tgl, jumlah)" +
                "Values (?,?)";
            pStatement = conn.prepareStatement(sql);
            pStatement.setString(1, tgl);

```

```

        pStatement.setInt(2, jumlah);

        int intTambah= pStatement.executeUpdate();
        if (intTambah>0){
            JOptionPane.showMessageDialog(this,
                "Simpan Sukses", "Informasi",
                JOptionPane.INFORMATION_MESSAGE);
        }
        pStatement.close();
    }

    catch (SQLException e){
        System.out.println("koneksi gagal " + e.toString());
        JOptionPane.showMessageDialog(this,
            "Terjadi Kesalahan :" + e.toString(), "Informasi",
            JOptionPane.INFORMATION_MESSAGE);
    }
}

@SuppressWarnings("unchecked")
private void initComponents() {

    jLabel2 = new javax.swing.JLabel();
    jDateChooser1 = new com.toedter.calendar.JDateChooser();
    jLabel4 = new javax.swing.JLabel();
    s_jan = new com.toedter.components.JSpinField();
    jLabel1 = new javax.swing.JLabel();
    s_merit = new com.toedter.components.JSpinField();

```

```
back1 = new tool.back();
jPanel5 = new javax.swing.JPanel();
jLabel6 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
jPanel8 = new javax.swing.JPanel();
background1 = new tool.background();
jPanel6 = new javax.swing.JPanel();
jPanel11 = new javax.swing.JPanel();
jPanel4 = new javax.swing.JPanel();
jPanel2 = new javax.swing.JPanel();
jPanel3 = new javax.swing.JPanel();
jScrollPane = new javax.swing.JScrollPane();
jList1 = new javax.swing.JList();
jLabel3 = new javax.swing.JLabel();
LTot = new javax.swing.JLabel();
jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();
jLabel5 = new javax.swing.JLabel();
cDurasi = new javax.swing.JComboBox();
jLabel7 = new javax.swing.JLabel();
txtTgl = new javax.swing.JTextField();
jButton3 = new javax.swing.JButton();
jButton4 = new javax.swing.JButton();
jPanel7 = new javax.swing.JPanel();
jPanel9 = new javax.swing.JPanel();

jLabel2.setText("Mulai");

jDateChooser1.setPreferredSize(new
java.awt.Dimension(104,20));
```

```

jLabel4.setText("Jam");
s_jam.setPreferredSize(new java.awt.Dimension(35, 20));
jLabel7.setText(":");
s_menit.setPreferredSize(new java.awt.Dimension(35, 20));

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

setTitle("Mengitung Jumlah Pengunjung Matos");
back1.setPreferredSize(new java.awt.Dimension(1154, 610));
back1.setLayout(new java.awt.BorderLayout());

jPanel5.setOpaque(false);
jPanel5.setPreferredSize(new java.awt.Dimension(784, 160));
jLabel6.setFont(new java.awt.Font("Gill Sans MT", 1, 18));
// NOI18N
jLabel6.setForeground(new java.awt.Color(0, 204, 255));

jLabel6.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel6.setText("APLIKASI PENGOLAHAN CITRA DIGITAL
PENGHITUNG JUMLAH PENGUNJUNG MALL ");
jLabel8.setFont(new java.awt.Font("Gill Sans MT", 1, 18));
// NOI18N
jLabel8.setForeground(new java.awt.Color(0, 204, 255));

jLabel8.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel8.setText(" MENGGUNAKAN METODE MORPHOLOGY");

javax.swing.GroupLayout jPanel5Layout = new
javax.swing.GroupLayout(jPanel5);
jPanel5.setLayout(jPanel5Layout);
jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
.addGroup(jPanel5Layout.createSequentialGroup()

```

```

        .addContainerGap()

        .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)

                .addComponent(jLabel6,
javax.swing.GroupLayout.PREFERRED_SIZE, 1204,
javax.swing.GroupLayout.PREFERRED_SIZE)

                .addComponent(jLabel8,
javax.swing.GroupLayout.PREFERRED_SIZE, 1204,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))

    };

    jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

        .addGroup(jPanel5Layout.createSequentialGroup())

            .addGap(22, 22, 22)

            .addComponent(jLabel6,
javax.swing.GroupLayout.PREFERRED_SIZE, 47,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

            .addComponent(jLabel8,
javax.swing.GroupLayout.PREFERRED_SIZE, 47,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addContainerGap(38, Short.MAX_VALUE))

    };

    back1.add(jPanel5, java.awt.BorderLayout.PAGE_START);

    jPanel8.setOpaque(false);
    jPanel8.setPreferredSize(new java.awt.Dimension(310, 280));

    javax.swing.GroupLayout jPanel8Layout = new
javax.swing.GroupLayout(jPanel8);

```

```

jPanel8.setLayout(jPanel8Layout);
jPanel8Layout.setHorizontalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

    .addGap(0, 310, Short.MAX_VALUE)

);
jPanel8Layout.setVerticalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

    .addGap(0, 345, Short.MAX_VALUE)

);

back1.add(jPanel8, java.awt.BorderLayout.LINE_START);

background1.setPreferredSize(new java.awt.Dimension(300,
400));
background1.setLayout(new java.awt.BorderLayout());

jPanel6.setOpaque(false);
jPanel6.setLayout(new java.awt.BorderLayout());

jPanel11.setPreferredSize(new java.awt.Dimension(200, 370));
jPanel11.setLayout(new java.awt.BorderLayout());
jPanel6.add(jPanel11, java.awt.BorderLayout.CENTER);

jPanel4.setOpaque(false);
jPanel4.setPreferredSize(new java.awt.Dimension(300, 314));
jPanel4.setLayout(new java.awt.BorderLayout());

```



```

jPanel2.setOpaque(false);
jPanel4.add(jPanel2, java.awt.BorderLayout.PAGE_START);
jPanel3.setOpaque(false);
jPanel3.setPreferredSize(new java.awt.Dimension(260, 335));
jScrollPane.setViewportView(jList1);
jLabel3.setText("Playlist");
lTot.setText("Total Pengunjung :");
jButton1.setText("Browse");
jButton1.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
evt) {
        jButton1ActionPerformed(evt);
    }
});

jButton2.setText("Play");
jButton2.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
evt) {
        jButton2ActionPerformed(evt);
    }
});

jLabel5.setText("Durasi");

cDurasi.setModel(new javax.swing.DefaultComboBoxModel(new
String[] { "250", "500", "750", "1000", "1250", "1500", "1750",
"2000", "2500", "3000" }));

```

```

cDurasi.setSelectedIndex(1);

jLabel7.setText("ms");

txtTgl.setEditable(false);

jButton3.setText("Simpan");
jButton3.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
evt) {
        jButton3ActionPerformed(evt);
    }
});

jButton4.setText("Lihat Data");
jButton4.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent
evt) {
        jButton4ActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
    .addGroup(jPanel3Layout.createSequentialGroup()

```

```

        .addContainerGap()

    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 280, Short.MAX_VALUE)
        .addComponent(jButton2)
        .addComponent(LTot)
        .addComponent(jButton3)
        .addGroup(jPanel3Layout.createSequentialGroup())

    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING)
        .addComponent(jButton1)
        .addComponent(jLabel3))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)

    .addGroup(jPanel3Layout.createSequentialGroup())
        .addComponent(jLabel5)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(cDurasi,
javax.swing.GroupLayout.PREFERRED_SIZE, 64,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel7))
        .addComponent(txtTgl,
javax.swing.GroupLayout.DEFAULT_SIZE, 207, Short.MAX_VALUE)))
        .addComponent(jButton4))
        .addContainerGap()

};

```

```

jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)

    .addGroup(jPanel3Layout.createSequentialGroup())

        .addContainerGap()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)

    .addComponent(jButton1)

    .addComponent(jLabel1b)

    .addComponent(cDurasi,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(jLabel7))

    .addGap(3, 3, 3)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)

    .addComponent(jLabel13)

    .addComponent(txtTgl,
javax.swing.GroupLayout.PREFERRED_SIZE, 22,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 79,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(jButton2)

    .addGap(18, 18, 18)

    .addComponent(LIot)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(jButton3)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton4)
    .addContainerGap(72, Short.MAX_VALUE)
);

jPanel14.add(jPanel13, java.awt.BorderLayout.CENTER);

jPanel16.add(jPanel14, java.awt.BorderLayout.CENTER);

background1.add(jPanel16, java.awt.BorderLayout.LINE_END);

jPanel7.setOpaque(false);
jPanel7.setLayout(new java.awt.BorderLayout());
background1.add(jPanel7, java.awt.BorderLayout.CENTER);

back1.add(background1, java.awt.BorderLayout.CENTER);

jPanel9.setOpaque(false);
jPanel9.setPreferredSize(new java.awt.Dimension(868, 20));

javax.swing.GroupLayout jPanel9Layout = new
javax.swing.GroupLayout(jPanel9);
jPanel9.setLayout(jPanel9Layout);
jPanel9Layout.setHorizontalGroup(

jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
    .addGroup(
        .addGap(0, 1224, Short.MAX_VALUE)
    )
);
jPanel9Layout.setVerticalGroup(

```

```
jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.  
LEADING;
```

```
    .addGroup(0, 20, Short.MAX_VALUE)
```

```
    );
```

```
back1.add(jPanel9, java.awt.BorderLayout.PAGE_END);
```

```
getContentPane().add(back1, java.awt.BorderLayout.CENTER);
```

```
pack();
```

```
///  
// </editor-fold>
```

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)  
{
```

```
// TODO add your handling code here:
```

```
    addFile(this);
```

```
    clearList();
```

```
    updateList();
```

```
    jList1.setSelectedIndex(jList1.getLastVisibleIndex());
```

```
}
```

```
Timer timert ;
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)  
{
```

```
// TODO add your handling code here:
```

```
    try {
```

```
        jPanel7.remove(vid);
```

```
    } catch (Exception e) {
```

```
    }
```

```
    try {
```

```
        timert.stop();
```

```

    } catch (Exception e) {
    }

    play(jList1.getSelectedIndex());

    JumlahPengunjung=0;

    timert = new
    Timer(Integer.parseInt(oDurasi.getSelectedItem().toString()), listener);

    timert.start();

    setResizable(false);
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{
// TODO add your handling code here:
    save(txtTgl.getText(), JumlahPengunjung);
}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
// TODO add your handling code here:
    new Data(this,
    rootPaneCheckingEnabled).setVisible(rootPaneCheckingEnabled);
}

    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
        javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    }
}

```

```

    }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(FUtama.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(FUtama.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);

    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(FUtama.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(FUtama.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);

    }
    //</editor-fold>
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new FUtama().setVisible(true);
        }
    });
}
BufferedImage imagelama=null;
BufferedImage imagebaru=null;
private BufferedImage image() {
    BufferedImage buf=null;
    try {
        Robot robot = new Robot();

```



```

    // BufferedImage img = robot.createScreenCapture(new
Rectangle(size));

    buf = robot.createScreenCapture(new
Rectangle(this.getSize()));

    ImageIO.write(buf, "JPG", new File("screenShot.jpg"));
} catch (Exception e) {
}

return buf;
}

int posisistart=0;
int JumlahPengunjung;

private BufferedImage getGrayScale(BufferedImage img){
    //BufferedImage imgHasil=new
BufferedImage(img.getWidth(),img.getHeight(),BufferedImage.TYPE_INT_
RGB);

    BufferedImage imgHasil=img;

    System.out.println("y= "+imgHasil.getHeight());
    System.out.println("x= "+imgHasil.getWidth());
    for(int y=0;y<imgHasil.getHeight();y++){
        for(int x=0;x<imgHasil.getWidth();x++){
            int gray=convertToGrayscale(img.getRGB(x, y));

            Color color=new Color(gray,gray,gray);
            imgHasil.setRGB(x, y, color.getRGB());
        }
    }

    return imgHasil;
}

private int convertToGrayscale(int rgb){
    int gray;

```

```

        gray=(int)(({(rgb>>16) & 0xFF)+((rgb>>8) & 0xFF)+((rgb>>0) &
0xFF))/3);

        return gray;
    }

    private static final int IMG_WIDTH = 100;
    private static final int IMG_HEIGHT = 100;

    private static BufferedImage resizeImage(BufferedImage
originalImage, int type){
        BufferedImage resizedImage = new BufferedImage(IMG_WIDTH,
IMG_HEIGHT, type);

        Graphics2D g = resizedImage.createGraphics();

        g.drawImage(originalImage, 0, 0, IMG_WIDTH, IMG_HEIGHT, null);

        g.dispose();

        return resizedImage;
    }

    private int Erosi_Dilasi( BufferedImage Fyzy,
BufferedImage Logic){

        int X1=0;
        int X2=0;
        int batas=70;

        for(int y=0;y<Fyzy.getHeight();y++){

            // System.out.println("progres 1 "+progres);

            for(int x=0;x<Fyzy.getWidth();x++){

                int gray=convertToGrayscale(Fyzy.getRGB(x, y));

                if(gray>batas){

                    //gray=1;

                }else{

                    //gray=0;

                    X1++;

                }

            }

        }
    }

```

```

    }

    }

    for(int y=0;y<Logic.getHeight();y++){
        //System.out.println("progres 2 "+progres);
        for(int x=0;x<Logic.getWidth();x++){
            int gray=convertToGrayscale(Logic.getRGB(x, y));
            if(gray>batas){
                // gray=1;
            }else{
                //gray=0;
                x2++;
            }
        }
    }

    int jum=0;
    if(x1>x2){
        }else{
            jum=x2-x1;
        }

    System.out.println(jum);

    image lama=Logic;

    BufferedImage test=Image();

        int type = test.getType() == 0?
BufferedImage.TYPE_INT_ARGB : test.getType();

        imagebaru=resizeImage(test, type);

return jum;

```

```
)  
  
// Variables declaration - do not modify  
private javax.swing.JLabel LTot;  
private tool.back back1;  
private tool.background background1;  
private javax.swing.JComboBox cDurasi;  
private javax.swing.JButton jButton1;  
private javax.swing.JButton jButton2;  
private javax.swing.JButton jButton3;  
private javax.swing.JButton jButton4;  
private com.toedter.calendar.JDateChooser jDateChooser1;  
private javax.swing.JLabel jLabel1;  
private javax.swing.JLabel jLabel2;  
private javax.swing.JLabel jLabel3;  
private javax.swing.JLabel jLabel4;  
private javax.swing.JLabel jLabel5;  
private javax.swing.JLabel jLabel6;  
private javax.swing.JLabel jLabel7;  
private javax.swing.JLabel jLabel8;  
private javax.swing.JList jList1;  
private javax.swing.JPanel jPanel1;  
private javax.swing.JPanel jPanel2;  
private javax.swing.JPanel jPanel3;  
private javax.swing.JPanel jPanel4;  
private javax.swing.JPanel jPanel5;  
private javax.swing.JPanel jPanel6;  
private javax.swing.JPanel jPanel7;  
private javax.swing.JPanel jPanel8;
```

```

private javax.swing.JPanel jPanel9;
private javax.swing.JScrollPane jScrollPane1;
private com.toedter.components.JSpinField s_jam;
private com.toedter.components.JSpinField s_menit;
private javax.swing.JTextField txtTgl;

// End of variables declaration

File file;
double m;

public void addFile(JFrame frame) {
    int fileValid = fileChooser.showOpenDialog(frame);
    if (fileValid == javax.swing.JFileChooser.APPROVE_OPTION) {
        file = fileChooser.getSelectedFile();
        System.out.print(file.getAbsolutePath());

        try {
            media=Manager.createRealizedPlayer(file.toURL());
        } catch (Exception e) {
        }

        m=media.getDuration().getSeconds();
        int durasi=(int)Math.round(m)/60;
        int jam=s_jam.getValue();
        int menit=s_menit.getValue();
        int waktu=menit+durasi;
        System.out.println("dfa"+m);
        String s= String.valueOf(durasi+"      "+jam+" : "+waktu);
        duration=durasi;
        listVideo.add(s);
        listVideo1.add(file);
    }
}

```

```

public ArrayList getListSong() {
    return listVideo;
}

public ArrayList getListSong1() {
    return listVideo1;
}

public void updateList() {
    ArrayList arl = getListSong();
    ArrayList arl1 = getListSong1();
    DefaultListModel model = new DefaultListModel();
    for (int i = 0; i < arl.size(); i++) {
        int j = i + 1;
        model.add(i, j + " " + ((File)arl1.get(i)).getName() + "
" + ((String) arl.get(i)));
        System.out.println("addo" + i);
        System.out.println("addso" + model);
    }
    jList1.setModel(model);
}

public void clearList() {
    jList1.removeAll();
}

int index;

public void play(int idx) {
    index = idx;
    int lastIndex = jList1.getLastVisibleIndex();
    if(index > lastIndex) {
        index = 0;
    } else if(index < 0) {

```

```

        index = lastIndex;
    }
    // timer.cancel();
    jList1.setSelectedIndex(index);
    playMusic(index);
    second = 1;
}
File song;
Player media;
public void setNewSong(int index) {
    song = (File) listVideo1.get(index);
    javax.media.Manager.setHint(javax.media.Manager.LIGHTWEIGHT_RENDERER, this);
    try {
        media=Manager.createRealizedPlayer(song.toURL());
    }
    catch (Exception ex) {
        System.out.println("harum"+ex);
    }
    System.out.println("stjh =" +statusPlay);
}
int statusPlay=0;
int duration = 1;
Component vid;
public void playMusic(int index) {
    if (statusPlay == 0) {
        setNewSong(index);
    } else if (statusPlay == 1) {
        media.stop();
        setNewSong(index);
    }
}

```

```

        System.out.println("status play? "+statusPlay);
    }

    vid=media.getVisualComponent();
    Component cntr=media.getControlPanelComponent();
    if (vid!=null) {
        jPanel17.add(vid, BorderLayout.CENTER);
    }
    if (cntr!=null) {
        //jPanel17.add(cntr, BorderLayout.SOUTH);
    }
    this.setVisible(true);
    this.setResizable(true);

    media.start();
    media.realize();
}

public int getPlayerDuration() {
    return duration;
}

int second;

private BufferedImage Dilasi(BufferedImage img){

    //BufferedImage imgHasil=new
    BufferedImage(img.getWidth(),img.getHeight(),BufferedImage.TYPE_INT_
    RGB);

    BufferedImage imgHasil=img;

    int getG[][]=new
    int [imgHasil.getHeight()][imgHasil.getWidth()];

    int getCl[][]=new
    int [imgHasil.getHeight()][imgHasil.getWidth()];

```



```

int gray=0;
int skala=10;
for(int y=0;y<imgHasil.getHeight();y++){
    for(int x=0;x<imgHasil.getWidth();x++){
        //imgHasil.setRGB(x, y, 0);
        gray=convertToGrayscale(img.getRGB(x, y));
        if(gray==0){

        }else{
            gray=255;
        }
        getG[x][y]=gray;
        getGl[x][y]=255;

        //System.out.println(((color.getRGB()>>16)&0xFF)+"
"+((color.getRGB()>>8)&0xFF)+" "+((color.getRGB()>>0)& 0xFF));
    }
}
for(int y=0;y<imgHasil.getHeight();y++){
    for(int x=0;x<imgHasil.getWidth();x++){

        if ( (x<=skala-1) ||
            (x>= imgHasil.getWidth()-skala) ||
            (y<=skala-1) ||
            (y >= imgHasil.getHeight()-skala)){

        }else{
            if(getG[x][y]==0){
                for(int j=y-skala;j<=y+skala;j++){
                    for(int i=x-skala;i<=x+skala;i++){

```

```

        getG1[i][j]=0;
    }
}
}
}
}

for(int y=0;y<imgHasil.getHeight();y++){
    for(int x=0;x<imgHasil.getWidth();x++){
        if(getG[x][y]==getG1[x][y])
        {
            gray=getG[x][y];
        }else{
            gray=getG1[x][y];
        }

        Color color=new Color(gray,gray,gray);
        imgHasil.setRGB(x, y, color.getRGB());
        //System.out.println(((color.getRGB()>>16)&0xFF)-"
        "+((color.getRGB()>>8)&0xFF)+" "+((color.getRGB()>>0)& 0xFF))
    }
}
return imgHasil;
}

private BufferedImage Erosi(BufferedImage img){
    //BufferedImage imgHasil=new
    BufferedImage(img.getWidth(),img.getHeight(),BufferedImage.TYPE_INT_
    RGB);
}

```

```

BufferedImage imgHasil=img;

int getG[][]=new
int[imgHasil.getHeight()][imgHasil.getWidth()];

int getG1[][]=new
int[imgHasil.getHeight()][imgHasil.getWidth()];

int gray=0;

int skala=10;

int ketelitian=100;

for(int y=0;y<imgHasil.getHeight();y++){
    for(int x=0;x<imgHasil.getWidth();x++){
        //imgHasil.setRGB(x, y, 0);

        gray=convertToGrayscale(img.getRGB(x, y));

        if(gray<ketelitian){
            gray=0;
        }else{
            gray=255;
        }

        getG[y][x]=gray;
    }
}

for(int y=0;y<imgHasil.getHeight();y++){
    for(int x=0;x<imgHasil.getWidth();x++){
        if ( (x<=skala-1) ||
            (x>= imgHasil.getWidth()-skala) ||
            (y<=skala-1) ||
            (y >= imgHasil.getHeight()-skala)){
        }else{
            if(getG[y][x]==255){
                for(int j=y-skala;j<=y+skala;j++){
                    for(int i=x-skala;i<=x+skala;i++){

```

```
        }  
        }  
    }  
    }  
    for(int y=0;y<imgHasil.getHeight();y++){  
        for(int x=0;x<imgHasil.getWidth();x++){  
            gray=getG1[y][x];  
            Color color=new Color(gray,gray,gray);  
        }  
    }  
    return imgHasil;  
}  
}
```



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo, KM 2 Malang

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

NAMA : Citra Muliana Baragain
NIM : 10.18.178
JURUSAN : Teknik Informatika S-1
JUDUL : **APLIKASI PENGOLAHAN CITRA DIGITAL PENGHITUNG
JUMLAH PENGUNJUNG MALL MENGGUNAKAN
METODE MORPHOLOGY**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :
Hari : Selasa
Tanggal : 18 Februari 2014
Nilai : 81.75 (A)

Panitia Ujian Skripsi :
Ketua Majelis Penguji

Joseph Dedy Irawan, ST., MT.
NIP. 19740416 2005011002

Anggota Penguji :

Dosen Penguji I

Karina Auliasari, ST., M.Eng
NIP.P. 1031000426

Dosen Penguji II

Sandy Nataly Mantja, S.Kom
NIP.P. 1030800418



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo, KM 2 Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata 1 Program Studi Teknik Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : Citra Muliana Baragain
NIM : 10.18.178
JURUSAN : Teknik Informatika S-1
JUDUL : APLIKASI PENGOLAHAN CITRA DIGITAL PENGHITUNG
JUMLAH PENGUNJUNG MALL MENGGUNAKAN
METODE MORPHOLOGY

No	Penguji	Tanggal	Uraian	Paraf
1.	Penguji I	18 Februari 2014	1. Perbaiki coretan-coretan dan lipatan-lipatan pada laporan yang ada. 2. Perbaiki pengujian sistem.	
2.	Penguji II	18 Februari 2014	1. Lakukan pengujian ditempat ramai, pusat perbelanjaan tidak harus Mall tapi bisa juga Minimarket. 2. Pengujian pada Bab IV lebih dilengkapi misal ada objek lain yang bergerak, dll. 3. Hitung juga lama waktu pengujian terdeteksi berapa objek? 4. Tulis dan uji coba jarak letak camera ke objek, jenis camera yang dipakai.	

Dosen Penguji I

Karina Auliasari, ST., M.Eng
NIP.P. 1031000426

Dosen Pembimbing I

Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002

Dosen Penguji II

Sandy Nataly Mantia, S.Kom
NIP.P. 1030800418

Dosen Pembimbing II

Nurlaily Vendvansyah, ST



INSTITUT TEKNOLOGI NASIONAL MALANG
 Kampus I : Jln. Bend. Sigara-gura No. 2 Malang
 Kampus II : Jln. Raya Karanglo Km. 2 Malang

PERMOHONAN PERSETUJUAN SKRIPSI

Yang bertanda tangan dibawah ini :

Nama : Citra Muliana Baragaim
 Alamat Rumah : Wadukon Mang salak no 10A. KendalPuyuk - Malang
 No Telp : Rumah : 081544963690 HP : 085154825603
 NIM : 1010178
 Semester : VII (tujuh)
 Fakultas : Teknologi Industri
 Jurusan : Teknik Informatika SI

Dengan ini kami mengajukan permohonan persetujuan membuat SKRIPSI Tingkat Sarjana :
 Untuk melengkapi permohonan tersebut, bersama ini kami lampirkan persyaratan - persyaratan
 yang harus dipenuhi.

Adapun persyaratan-persyaratan pengajuan SKRIPSI adalah sebagai berikut :

- | | |
|---|---|
| 1. Telah melaksanakan semua Praktikum | (<input checked="" type="checkbox"/>) |
| 2. Telah Lulus dan menyerahkan Laporan Praktek Kerja | (<input checked="" type="checkbox"/>) |
| 3. Telah menempuh mata kuliah \geq 134 sks dengan IPK \geq 2 dan tidak ada nilai E. | (<input checked="" type="checkbox"/>) |
| 4. Telah mengikuti secara aktif kegiatan seminar skripsi yang diadakan jurusan | (<input checked="" type="checkbox"/>) |
| 5. Memenuhi persyaratan administrasi. | (<input checked="" type="checkbox"/>) |

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya
 kami ucapkan terima kasih.

Telah diteliti kebenaran data tersebut diatas
 Recording T. Informatika SI

Malang, 13 September 2013
 Pemohon,

Titik Rembati
 NIP.Y. 1039200229

Citra Muliana Baragaim

Disetujui
 Ketua Jurusan T. Informatika SI

Mengetahui,
 Dosen Wali,

Joseph Dedy Irawan, ST, MT
 NIP : 197404162005011002

Sonny Prasetio, ST, MT
 NIP-P. 1031000433

Catatan :

Bagi mahasiswa yang telah memenuhi persyaratan mengajukan SKRIPSI agar membuat proposal
 dan mendapat persetujuan dari Ketua Jurusan/ Sekretaris Jurusan Teknik Informatika SI

Malang, 21 Oktober 2013

Lampiran : 1(Satu) berkas
Perihal : Kesediaan sebagai Pembimbing Skripsi

Kepada : Yth. Bpk/Ibu **Joseph Dedy Irawan, ST, MT**
Dosen Pembina Prodi Teknik Informatika S-1
Institut Teknologi Nasional
MALANG

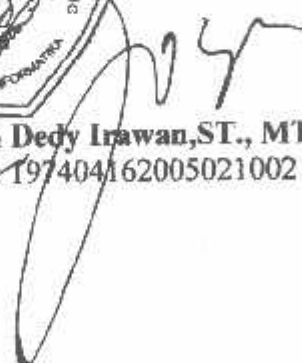
Yang bertanda tangan dibawah ini:

Nama : CITRA MULIANA BARAGAIN
Nim : 1018178
Prodi : Teknik Informatika S-1

Dengan ini mengajukan permohonan, kiranya bapak bersedia menjadi Dosen Pembimbing Utama / ~~Pendamping~~ *), untuk penyusunan Skripsi dengan judul (Proposal Terlampir) :

**Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pengunjung Mall
Menggunakan Metode Morphology**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik. Demikian permohonan kami dan atas kesediaan bapak kami sampaikan terima kasih.

Prodi T. Informatika S-1
Ketua,

Joseph Dedy Irawan, ST., MT.
NIR: 197404162005021002

Hormat Kami,

CITRA MULIANA BARAGAIN

Form S-3a

Malang, 21 Oktober 2013

Lampiran : 1(Satu) berkas
Perihal : Ketersediaan sebagai Pembimbing Skripsi

Kepada : Yth. Bpk/Ibu **Nurlaily Vandyansyah, ST**
Dosen Pembina Prodi Teknik Informatika S-1
Institut Teknologi Nasional
MALANG

Yang bertanda tangan dibawah ini:

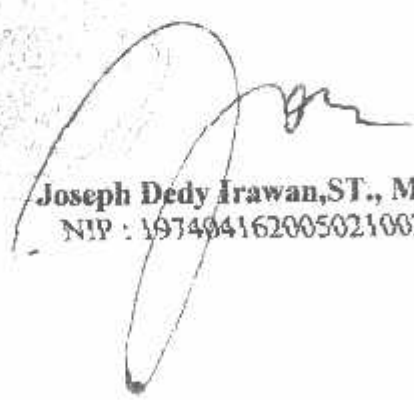
Nama : CITRA MULIANA BARAGAIN
Nim : 1018178
Prodi : Teknik Informatika S-1

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing ~~Utama~~ / Pendamping *), untuk penyusunan Skripsi dengan judul (Proposal Terlampir) :

**Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pengunjung Mall
Menggunakan Metode Morphology**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik. Demikian permohonan kami dan atas kesediaan bapak kami sampaikan terima kasih.

Prodi T. Informatika S-1
Ketua,


Joseph Dedy Irawan, ST., MT.
NIP : 197404162005021002

Hormat Kami,


CITRA MULIANA BARAGAIN

Form S-3a

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : CITRA MULIANA BARAGAIN

Nim : 1018178

Program Studi : Teknik Informatika

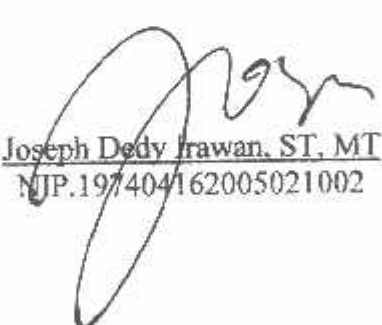
Dengan ini menyatakan bersedia / ~~tidak bersedia~~ *) membimbing skripsi dari mahasiswa tersebut dengan judul :

**Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pengunjung Mall
Menggunakan Metode Morphology**

Demikian Surat Pernyataan ini kami buat agar dipergunakan seperlunya.

Malang, 21 OKTOBER 2013

Hormat Kami,


Joseph Dedy Irawan, ST, MT
NIP.197404162005021002

Catatan :
Setelah disetujui agar formulir ini diserahkan mahasiswa/i
yg bersangkutan kepada Jurusan untuk diproses lebih lanjut
*) coret yang tidak perlu

Form S-3b

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : CITRA MULIANA BARAGAIN

Nim : 1018178

Program Studi : Teknik Informatika


Dengan ini menyatakan bersedia / tidak bersedia *) membimbing skripsi dari mahasiswa tersebut dengan judul :

**Aplikasi Pengolahan Citra Digital Penghitung Jumlah Pengunjung Mall
Menggunakan Metode Morphology**

Demikian Surat Pernyataan ini kami buat agar dipergunakan seperlunya.

Malang, 21 OKTOBER 2013

Hormat Kami,



Nurlaily Vendyansyah, ST

Catatan :

Setelah disetujui agar formulir ini diserahkan mahasiswa/i yg bersangkutan kepada Jurusan untuk diproses lebih lanjut

*) coret yang tidak perlu

Form S-3b



FORMULIR BIMBINGAN SKRIPSI

Nama : CITRA MULIANA BARAGAIN
NIM : 10.15.178
Masa Bimbingan : 21 OKTOBER 2013 s/d 21 MARET 2014
Judul Skripsi : APLIKASI PENEOLAHAN CITRA DIGITAL PENGHITUNG
JUMLAH PENGUNTING MALL MENGGUNAKAN METODE MORPHOL

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	30 Januari 2014	Demo Program + Revisi	
2	04 februari 2014	Revisi Laporan	
3	05 februari 2014	Revisi Makalah Seminar	
4	07 februari 2014	Acc. Makalah Seminar	
5	15 februari 2014	Review Seminar	
6	17 - 02 - 2014	All complete	
7			
8			
9			
10			

Malang, 15 Februari 2014

Dosen Pembimbing

Joseph Oedy Irawan, ST., MT.

NIP : 197404162005021002



FORMULIR BIMBINGAN SKRIPSI

Nama : CITRA MULIANA BARAGAIN
NIM : 10.18.178
Masa Bimbingan : 21 OKTOBER 2013 S/D 21 MARET 2014
Judul Skripsi : APLIKASI PENGOLAHAN CITRA DIGITAL PENGHITUNG TUNJUK PENGUNTING MALL MENGGUNAKAN METODE MORPHOLOGY

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	06 November 2013	Revisi Laporan Bab 1, 2	
2	11 Desember 2013	Revisi Laporan Bab 3	
3	30 Januari 2014	Revisi Laporan + Demo program	
4	03 februari 2014	Revisi Laporan Bab IV	
5	04 februari 2014	Revisi Laporan Bab IV dan V	
6	5 Februari 2014	Revisi Makalah Seminar	
7	07 Februari 2014	Acc. Makalah Seminar	
8	12 Februari 2014	Review Seminar	
9	13 Februari 2014	Revisi Bab III dan IV	
10	14 Februari 2014	Acc. Ujian Skripsi	

Malang, 15, Februari 2014
Gesen Pembimbing
14/2 2014
Nurbaily Vandyansyah, ST
NIP :