**LAPORAN PENELITIAN DOSEN**

****

**PENGEMBANGAN SISTEM INFORMASI DETEKSI PRESENSI OTOMATIS MENGGUNAKAN BUBBLE DETECTOR**

**Peneliti:**

**Yusron Rijal, S.Si, MT (NIDN. 0715067901)**

**LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**BANGIL**

**MARET 2013**

**HALAMAN PENGESAHAN**

**PENELITIAN DOSEN PEMULA**

**Judul** : PENGEMBANGAN SISTEM INFORMASI DETEKSI PRESENSI OTOMATIS MENGGUNAKAN BUBBLE DETECTOR

**Kode/Nama Rumpun** : 058/Teknik Informatika

**Ketua Tim Pengusul**

1. Nama Lengkap : Yusron Rijal, S.Si, MT
2. NIDN : 0715067901
3. Jabatan Fungsional : Asisten Ahli
4. Program Studi : Teknik Informatika
5. Nomor HP : 0856-5500-7801
6. Alamat Surel (E-mail) : [yusronrijal@stmik-yadika.ac.id](mailto:yusronrijal@stmik-yadika.ac.id)

**Biaya Penelitian** : - Diusulkan Ke DIKTI Rp -

* Dana Internal PT Rp 2.900.000,-
* Dana Institusi Lain Rp -
* Inkind Rp -

|  |  |
| --- | --- |
| Bangil, 16 Maret 2013 | |
| Mengetahui,  Ketua STMIK Yadika,  Tanda tangan  **Dr. Djoko Sugiono, MT** | Ketua Tim Pengusul,  Tanda tangan  **Yusron Rijal, S.Si, MT**  NIDN. 0716027302 |
| Menyetujui,  Ketua LPPM  Tanda tangan  **M. Imron, ST**  NIK. 09110680007 | |

**DAFTAR ISI**

Halaman Pengesahan ii

Daftar Isi iii

Ringkasan iv

Bab I Pendahuluan 1

Rumusan Masalah 2

Batasan Masalah 2

Tujuan Penelitian 3

Luaran Penelitian 3

Kontribusi Penelitian 3

Bab II Tinjauan Pustaka 4

Bab III Metode Penelitian 11

Bab IV Biaya Dan Jadwal Penelitian 13

Daftar Pustaka 14

**RINGKASAN**

STMIK Yadika bangil, institusi pendidikan yang menerapkan sistem informasi yang  menumbuhkan sikap istiqomah dalam menuntut ilmu. Oleh karena itu pencatatan kehadiran mahasiswa digunakan sebagai syarat untuk mengikuti UAS (Ujian Akhir Semester) yaitu minimal 75% mahasiswa harus menghadiri perkuliahan. Pencatatan kehadiran mahasiswa juga digunakan sebagai catatan bahwa mahasiswa mengikuti perkuliahan serta dapat digunakan sebagai bahan pembanding prestasi mahasiswa. Namun sistem perkuliahan bisa dikatakan dapat berjalan dengan baik, jika pencatatan kehadiran mahasiswa efektif dan efisien. Sistem presensi STMIK Yadika saat ini sudah terkomputerisasi namun penggunaannya masih manual yakni admin menginputkan satu persatu data kehadiran mahasiswa. Sistem presensi kehadiran mahasiswa yang digunakan saat ini dirasa kurang efektif dan efisien serta memiliki banyak kelemahan, selain memerlukan banyak waktu dan intervensi pegawai administrasi SDM, peluang kesalahan penginputan data kehadiran sangat besar.

Sistem pencatatan dan pelaporan yang secara simultan, dengan hasil data yang dapat dipertanggungjawabkan keautentikannya sangat diperlukan, selain lebih efektif pantauan dan antisipasi lebih dini terhadap perkembangan mahasiswa. Hal inilah yang melatarbelakangi diperlukannya suatu sistem informasi yang dapat diandalkan untuk menyelesaikan masalah tersebut.

Oleh karena itu banyak ditemukan beberapa solusi yang mendukung akan persoalan tersebut, seperti membagun sistem presensi berbasis deteksi wajah (Hanif, 2006) dan sidik jari, namun pencatatan kehadiran

**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

Menuntut  ilmu diwajibkan sejak seseorang dilahirkan sampai meninggal dunia. Dengan demikian,   dalam  menuntut ilmu dibutuhkan sikap sungguh-sungguh dan konsisten *(istiqomah)*  dalam mempelajari dan mengkaji berbagai ilmu demi tercapainya sebuah   cita-cita  *(‘azam)*.(ta’limul muta’allim, hal 15, 88)

Begitupulah dengan STMIK Yadika bangil, institusi pendidikan yang menerapkan sistem informasi yang  menumbuhkan sikap istiqomah dalam menuntut ilmu. Oleh karena itu pencatatan kehadiran mahasiswa digunakan sebagai syarat untuk mengikuti UAS (Ujian Akhir Semester) yaitu minimal 75% mahasiswa harus menghadiri perkuliahan. Pencatatan kehadiran mahasiswa juga digunakan sebagai catatan bahwa mahasiswa mengikuti perkuliahan serta dapat digunakan sebagai bahan pembanding prestasi mahasiswa. Namun sistem perkuliahan bisa dikatakan dapat berjalan dengan baik, jika pencatatan kehadiran mahasiswa efektif dan efisien. Sistem presensi STMIK Yadika saat ini sudah terkomputerisasi namun penggunaannya masih manual yakni admin menginputkan satu persatu data kehadiran mahasiswa. Sistem presensi kehadiran mahasiswa yang digunakan saat ini dirasa kurang efektif dan efisien serta memiliki banyak kelemahan, selain memerlukan banyak waktu dan intervensi pegawai administrasi SDM, peluang kesalahan penginputan data kehadiran sangat besar.

Sistem pencatatan dan pelaporan yang secara simultan, dengan hasil data yang dapat dipertanggungjawabkan keautentikannya sangat diperlukan, selain lebih efektif pantauan dan antisipasi lebih dini terhadap perkembangan mahasiswa. Hal inilah yang melatarbelakangi diperlukannya suatu sistem informasi yang dapat diandalkan untuk menyelesaikan masalah tersebut.

Oleh karena itu banyak ditemukan beberapa solusi yang mendukung akan persoalan tersebut, seperti membagun sistem presensi berbasis deteksi wajah (Hanif, 2006) dan sidik jari, namun pencatatan kehadiran menggunakan sidik jari membutuhkan biaya yang lebih besar karena harga sensor sidik jari yang relatif mahal (Sumarmo, 2012), sedangkan kebutuhan akan piranti Finger Print pada sebuah instansi sekolah tidak cukup hanya satu atau dua alat saja mengingat begitu banyaknya jumlah mahasiswa yang ada. terlebih presepsi publik dari sejumlah responden hanya 58% yang dapat menerima impelementasi terhadap perangkat biometric, 41 % merasa tidak nyaman jika menggunakan iris scan dan 47 % untuk penggunaan retina scan. (Moody, 2004)

Dengan adanya beberapa permasalan tersebut penulis tertarik mengambil judul **“PENGEMBANGAN SISTEM INFORMASI DETEKSI PRESENSI OTOMATIS MENGGUNAKAN BUBBLE DETECTOR”** yang memanfaatkan scanner dengan mengimplementasikan image *thresholding,* operasi morfologi *– opening(*Masfran, dkk. 2012)dan *chain code, (*Yusron, 2010) untuk mengembangkan sistem informasi presensi yang mampu mengenerate bulatan-bulatan secara otomatis sesuai dengan data pada lembar presensi mahasiswa yang nantinya akan di-detect dan di-extract kembali sebagai input rekap data presensi

1. **Perumusan Masalah**

Berdasarkan latar belakang di atas, maka ditemukan beberapa rumusan masalah sebagai berikut :

1. Bagaimana merancang sistem informasi yang mampu mengolah citra digital untuk di-Geenerate secara Otomatis menjadi Lembar Presensi Mahasiswa ?
2. Bagaimana merancang sistem informasi yang mampu mengolah citra digital untuk mengekstrak data Lembar Presensi Mahasiswa?
3. **Batasan Masalah**

Melihat luasnya permasalahan yang ada, maka dalam penelitian ini akan dibatasi pada masalah-masalah sebagai berikut:

1. Sistem informasi manajemen ini dirancang untuk Mahasiswa STMIK Yadika
2. Data yang diolah pada Lembar Presensi Mahasiswa sesuai dengan rancangan desain yang ada.
3. Kertas yang digunakan berukuran A4 dengan berat minimal 70gms
4. Citra LPM berformat Bitmap (.BMP)
5. Database Presensi Mahasiswa mengambil dari SIAKAD STMIK Yadika
6. Scanner yang digunakan berjenis Automatic Document Feeder
7. Informasi kelayakan mahasiswa mengikuti UAS ditampilkan oleh SIAKAD.
8. Kecepatan olah data menyesuaikan jenis Scanner dan Komputer yang digunakan.
9. Tidak menjamin kemungkinan peluang mahasiswa untuk melakukan manipulasi data kehadiran.
10. **Tujuan dan Manfaat penelitian**
11. Tujuan yang akan dicapai dalam penelitian ini adalah :
12. Rancang Bangun sistem informasi dengan mengolah citra Template LPM untuk di-Geenerate secara Otomatis menjadi Lembar Presensi Mahasiswa
13. Rancang Bangun sistem informasi dengan mengolah citra digital untuk mengekstrak data Lembar Presensi Mahasiswa
14. Manfaat yang diperoleh dari skripsi ini adalah :
15. Bagi penyusun
16. Menerapkan ilmu yang telah diperoleh selama proses kuliah di STMIK Yadika Bangil
17. Menambah wawasan dalam teknologi pengolahan citra digital.
18. Pembelajaran untuk membangun sebuah aplikasi yang bertujuan menyelesaikan masalah yang muncul.
19. Sebagai bahan pertimbangan teori dan praktek sehingga bermanfaat untuk masa yang akan dating
20. Bagi STMIK Yadika Bangil

Sebagai bahan perpustakaan dan studi banding bagi rekan-rekan mahasiswa yang melakukan penelitian tentang masalah yang ada serta sebagai bahan pertimbangan dalam Pengembangan Sistem Informasi Deteksi Presensi Otomatis Menggunakan *Bubble Detector*

1. Bagi Pengguna

Solusi lain yang dapat digunakan sebagai pencatatan kehadiran yang efektif dan efisien waktu, tenaga, biaya serta mengurangi kesalahan entri data, sehingga dapat dijangkau serta diaplikasikan pada instansi-instansi khususnya instansi pendidikan.

1. **Metode Penelitian**
   1. Studi Literatur

Bentuk pencarian informasi dengan cara membaca/mengambil informasi dari makalah, jurnal ilmiah, buku dan juga pemanfaatan internet sebagai sumber informasi, dengan jalan melihat informasi yang disediakan oleh situs-situs web, forum diskusi dan sebagainya.

* 1. Observasi dan Wawancara

Pada tahap ini peneliti melakukan pengamatan dan wawancara kepada pihak terkait meliputi lembaga pendidikan, praktisi pendidikan, dan instansi-intansi.

* 1. Perancangan Sistem

Pada tahap ini yang dilakukan adalah membuat perancangan sistem yaitu membuat block diagram, desain form, menentukan dan menyusun algoritma untuk Pengembangan Sistem Informasi Deteksi Presensi Otomatis Menggunakan *Bubble Detector*

* 1. Pembuatan Program

Dalam hal ini yang dilakukan oleh peneliti adalah membuat program Pengembangan Sistem Informasi Deteksi Presensi Otomatis Menggunakan *Bubble Detector* berdasarkan algoritma yang telah ditentukan dan disusun pada tahap sebelumnya.

* 1. Uji Coba Sistem

Pada tahap ini dilakukan uji coba program untuk mengetahui sejauh mana hasil dari program yang telah dibuat, serta melakukan perbaikan apabila terjadi kesalahan pada program.

* 1. Analisa Hasil Program

Pada tahap terakhir ini dilakukan analisa terhadap hasil program Pengembangan Sistem Informasi Deteksi Presensi Otomatis Menggunakan *Bubble Detector* yang telah diterapkan, kemudian mengambil suatu kesimpulan dari hasil uji coba tersebut.

1. **Sistematika Penulisan**

Sistematika penulisan proposal skripsi ini adalah:

* BAB I PENDAHULUAN

Bab ini merupakan dasar penyusunan laporan yang di dalamnya berisi tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

* BAB II LANDASAN TEORI

Dalam bab ini membahas mengenai teori dasar yang mendukung dalam perencanaan sistem serta penjelasan tentang komponen/piranti yang menunjang perealisasian sistem.

* BAB III METODOLOGI PENELITIAN

Bab ini membahas mengenai cara pengambilan dan pengolahan data dengan menggunakan alat-alat analisis yang ada.

* BAB IV PERANCANGAN SISTEM

Bab ini berisi tentang tahap analisis yaitu identifikasi dan analisis masalah, serta analisis kebutuhan sistem untuk menyelesaikan masalah yang dihadapi. Bab ini juga berisi hasil perancangan yaitu proses kelanjutan dari tahap analisis meliputi proses akusisi pengetahuan.

* BAB V HASIL DAN PEMBAHASAN

Pada bab ini membahas tentang implementasi dari aplikasi. Serta melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah aplikasi telah dapat menyelesaikan permasalahan yang dihadapi sesuai dengan yang diharapkan.

* BAB VI PENUTUP

Bab ini berisi kesimpulan dan saran-saran pengembangan lebih lanjut dari alat tersebut.

* DAFTAR PUSTAKA

Berisi daftar pustaka atau referensi-referensi baik berupa media cetak maupun media elektronik yang dapat dijadikan acuan dalam penelitian tugas akhir.

* LAMPIRAN

**BAB II**

**LANDASAN TEORI**

1. **Definisi Pengembangan Sistem ( System Development )**

[Jogiyanto,1998] mendefiniskan pengembangan sistem (Systems Development) dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada.

1. **Sistem Informasi**

Untuk menghasilkan informasi yang berkualitas maka dibuatlah sistem informasi. Menurut Robert A. Laitch dan K. Roscoe Bavis "Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan."

Dalam suatu sistem informasi terdapat komponen-komponen sebagai berikut :

* + Perangkat Keras (Hardware). Mencakup berbagai peranti fisik seperti komputer dan printer.
  + Perangkat Lunak (Software) atau program, yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data.
  + Prosedur, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki
  + Orang, yaitu semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.
  + Basis data(database) yaitu sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data
  + Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sumber (resources)dipakai secara bersama atau diakses oleh sejumlah pemakai. (Kusrini,dkk 2007)
  1. **Definisi Sistem**

Sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan.

Dari definisi diatas maka dapat diketahui manfaat sistem yaitu untuk menyatukan atau mengintegrasikan semua unsur yang ada dalam suatu ruang lingkup, dimana komponen-komponen tersebut tidak dapat berdiri sendiri. (Kusrini, dkk. 2007)

* 1. **Informasi**

Informasi dapat didefinisikan sebagai suatu data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, dan informasi ini menggambarkan kejadian-kejadian nyata untuk menambah pemahamannya terhadap fakta-fakta yang ada, sehingga dapat digunakan untuk pengambilan suatu keputusan. Sumber informasi adalah data. Data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan yang nyata

1. **Deteksi Presensi Otomatis**

Deteksi Presensi Otomatis merupakan presensi pengganti tenaga manusia dengan tenaga mesin yang terkontrol dan terkomputerisasi secara otomatis sehingga dapat mengatur pekerjaan yang tidak lagi memerlukan pengawasan khusus.

1. **Deteksi**

Pemantuan dengan menggunakan alat pencatatan atau perekaman,(Windi, 2010)

1. **Presensi**

Presensi adalah suatu pendataan kehadiran, bagian dari pelaporan aktifitas suatu institusi, atau komponen institusi itu sendiri yang berisi data-data kehadiran yang disusun dan diatur sedemikian rupa sehingga mudah untuk dicari dan dipergunakan apabila sewaktu-waktu diperlukan oleh pihak yang berkepentingan.[http:// edipurwanto1988.blogspot.com/2009/06/keamanan-sistem-informasi-presensi.html (09-September-2012)]

1. **Otomatis**

Otomatis diartikan yang berkemampuan untuk bekerja dengan sendirinya. (Windi, 2010), otomatis dapat juga didefinisikan sebagai suatu tekhnologi yang berkaitan dengan aplikasi mekanik, elektronik dan sistem yang berbasis komputer. Semuanya bergabung menjadi satu untuk memberikan fungsi terhadap manipulator (mekanik) sehingga akan memiliki fungsi tertentu. (Cahyana, 2009)

1. **Bubble Detector**

Bubble detektor merupakan kombinasi dari beberapa algoritma yang difungsikan untuk mengenali pola dan posisi lingkaran yang kemudian diekstraksi dan dibedakan antara bulatan satu dengan yang lain yang salah satunya melalui Integral Projection (Riwinoto. 2012), (Amaliah. 2011)

1. **Penerapan Bubble Detector**

Algoritma yang digunakan pada Bubble Detector juga sering diterapkan dalam deteksi bentuk. Tak sedikit penelitian menggunakan tekni ini dengan berbagai metode yang berbeda. Berikut ini adalah beberapa penelitian yang berkaitan dengan bubble detector.

1. *A Chain Code Approach for Recognizing Basic Shapes*. (Azzam Ibrahim, 2006)

Azam memanfaatkan algoritma freeman chain code untuk mengenali jenis dan ukuran bentuk pada beberapa bangun ruang 2D

1. *Segmentasi Tepi Citra CT Scan Paru-paru Menggunakan Metode Chain Code dan Operasi Morfologi* (Masfran,dkk, 2012)

Di dalam jurnalnya beliau menerapkan metode chain code untuk mendeteksi tepi pada paru-paru manusia, sebagai dasar diagnosa penyakit paru-paru

1. **Pengertian Citra**

Citra adalah suatu representasi, kemiripan, atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal – sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan (Pillai, 2002).

Citra dapat dikelompokkan menjadi dua bagian yaitu citra diam yaitu citra tunggal yang tidak bergerak dan citra bergerak yaitu rangkaian citra diam yang ditampilkan secara beruntun, sehingga memberi kesan pada mata sebagai gambar yang bergerak. Setiap citra didalam rangkaian itu disebut frame. Gambar-gambar yang tampak pada film layar lebar atau televisi yaitu terdiri dari ribuan sampai ratusan-ribu frame (Sutoyo, T,dkk. 2009)

* + 1. **Citra Analog**

Citra analog adalah citra yang bersifat kontinu, seperti gambar pada monitor televisi, foto sinar-X yang tercetak di kertas foto, lukisan, pemandangan alam, hasil *CT-scan*, dan lain sebagainya. Citra analog tidak dapat direpresentasikan dalam komputer sehingga tidak bisa diproses di computer secara langsung. Agar citra analog dapat diproses di komputer, proses konversi analog ke digital harus dilakukan terlebih dahulu. (Sutoyo, T,dkk. 2009)

* + 1. **Citra Digital**

Citra digital dapat didefinisikan sebagai fungsi dua variabel, *f(x,y)*, dimana *x* dan *y* adalah koordinat spasial dan nilai *f(x,y)* adalah intensitas citra pada koordinat tersebut, hal tersebut diilustrasikan pada gambar 2.1. Teknologi dasar untuk menciptakan dan menampilkan warna pada citra digital berdasarkan pada penelitian bahwa sebuah warna merupakan kombinasi dari tiga warna dasar, yaitu merah, hijau, dan biru (*Red, Green, Blue* - RGB).

Sensor optik yang terdapat di dalam sistem pencitraan disusun sedemikian rupa sehingga membentuk bidang dua dimensi (x, y). besar intensitas yang diterima sensor di setiap titik (x, y) disimbolkan oleh f(x, y) dan besarnya tergantung pada intensitas yang dipantulkan oleh objek. Ini berarti f(x, y) sebanding dengan energi yang dipancarkan oleh sumber cahaya. Konsekuensinya, besar intensitas f(x, y) tidak bolah nol dan harus berhingga, yaitu :

0 < f (x, y) < ∞

(2.1)

Fungsi f(x, y) dapat dipisahkan menjadi dua komponen, yaitu:

* + - * 1. Jumlah cahaya yang berasal dari sumbernya disimbolkan oleh i(x, y) (*illumination*), nilainya antara 0 dan ∞,
        2. Derajat kemampuan objek memantulkan cahaya r(x, y) (*reflection*), nilainya antara 0 dan 1. (Pillai, 2002)

Besar f(x, y) merupakan kombinasi perkalian dari keduanya,

0 < i(x, y) < ∞

f(x, y) = i(x, y).r(x, y)

(2.2)

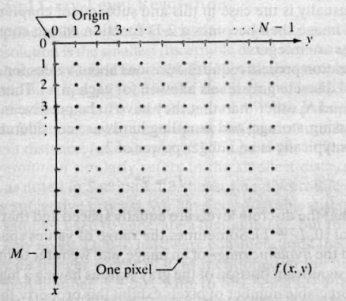
di mana

(2.3)

dan

0 < r(x, y) < 1

(2.4)

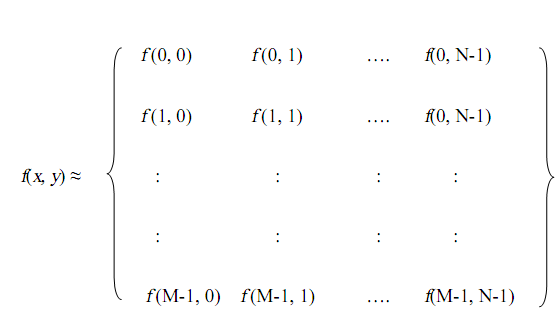
Nilai i(x, y) ditentukan oleh sumber cahaya, sedangkan r(x,y) ditentukan oleh karakteristik objek di dalam gambar. Nilai r(x, y) = 0 mengindikasikan penyerapan local, sedangkan r(x, y) = 1 menyatakan pemantulan total. Jika permukaan mempunyai derajat pemantulan nol maka fungsi intensitas cahaya f(x, y) juga nol. Sebaliknya, jika permukaan mempunyai derajat pemantulan 1 maka fungsi intensitas cahaya sama dengan iluminasi yang diterima oleh permukaan tersebut. Intensitas f(x, y) di titik (x, y) disebut derajat keabuan (gray level), yang dalam hal ini derajat keabuan bergerak dari hitam ke putih, sedangkan citranya disebut citra skala keabuan (grayscale image). Derajat keabuan memiliki rentang nilai dari Imin < f(x, y) < Imax atau bisa ditulis dalam bentuk (Imin, Imax). Rentang nilai ini sering digeser untuk alasan-alasan praktis menjadi selang [0, L] yang dalam hal ini intensitas 0 menyatakan hitam, nilai intensitas L menyatakan putih. (Sutoyo, T,dkk. 2009).

**Gambar 2.1** Koordinat Citra Digital

Sumber: Putra, Darma (2010,p20)

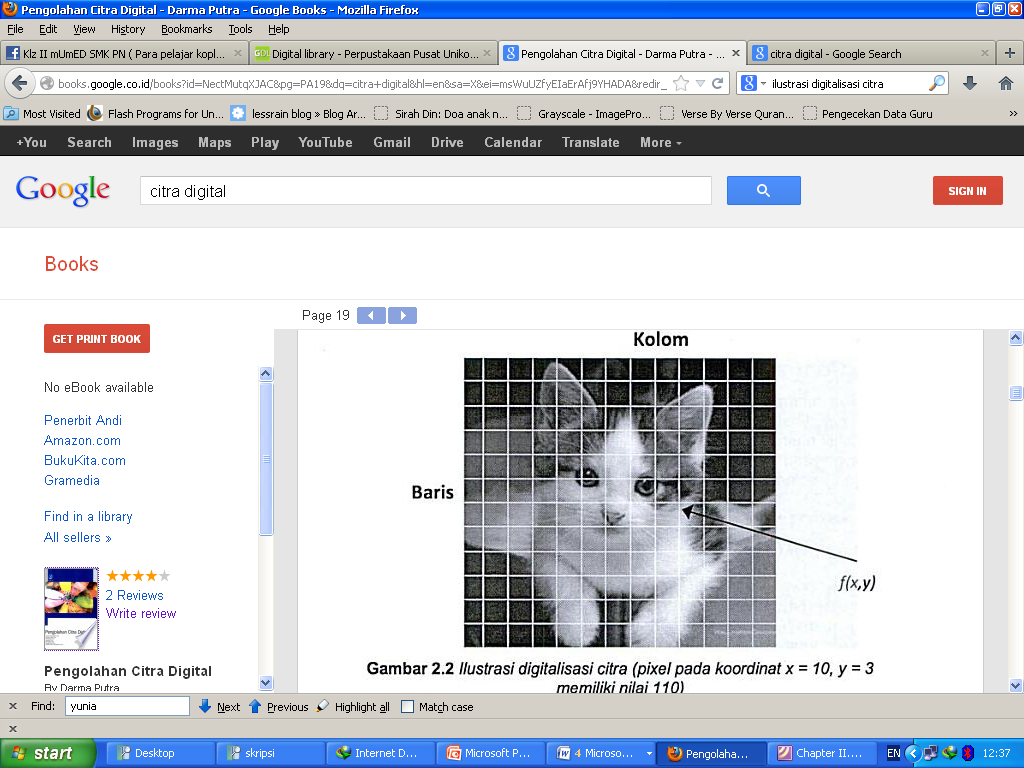
Pada Gambar 2.1 mengilustrasikan citra digital sebagai fungsi dua variable, f(x,y), dimana x dan y adalah koordinat spasial dan nilai f(x,y) adalah intensitas citra pada koordinat tersebut.

Citra digital yang berukuran M x N biasanya dinyatakan dalam bentuk matriks yang berukuran M baris dan N kolom, sebagai berikut:



(2.5)

Sumber: Putra, Darma (2010,p20)

Setiap elemen pada citra digital (elemen matriks) disebut image element, picture element atau pixel, makin tinggi nomor piksel maka makin gelap (hitam) piksel tersebut. Begitu juga sebaliknya makin rendah nilai piksel tersebut maka semakin terang. Sistem yang umum memiliki 256 tingkat kecerahan untuk setiap piksel, yang paling terang adalah 255 dan yang paling gelap adalah 0.

**Gambar 2.2** Ilustrasi digitalisasi citra

(pixel pada koordinat x=10, y=3 memiliki nilai 110)

Berdasarkan nilai pikselnya, gambar dapat dibedakan menjadi:

* 1. Citra Biner

Citra yang hanya memiliki dua kemungkinan nilai piksel yaitu hitam dan putih.

* 1. Citra Grayscale

Citra grayscale merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pikselnya. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih.

* 1. Citra Warna (8 bit)

Citra yang setiap piksel dari citra warna hanya diwakili oleh 8 bit dengan jumlah warna maksimum yang dapat digunakan 256 warna.

* 1. Citra Warna (16 bit)

Citra warna 16 bit biasanya disebut sebagai citra highcolor , setiap piksel nya diwakili dengan 2 byte memory (16 bit). Warna 16 bit memiliki 65.536 warna.

* 1. Citra Warna (24 bit)

Setiap piksel dari citra warna 24 bit diwakili dengan 24 bit sehingga memiliki total 16.777.216 warna. Setiap poin informasi piksel (Red,Green,Blue) disimpan ke dalam 1 byte data. 8 bit menyimpan nilai biru, 8 bit menyimpan nilai hijau, dan 8 bit menyimpan nilai merah.

Gambar digital juga memiliki sistem ruang warna. Sistem warna yang umumnya digunakan ialah:

1. RGB (Red Green Blue) yang terdiri dari tiga buah warna utama yaitu merah, hijau, dan biru.
2. YCbCr yang merupakan merupakan domain yang berbeda dengan domain warna RGB yang terdiri dari luminance (Y), chrominance biru (Cb), chrominance merah (Cr). Luminance membawa informasi tentang tingkat kecerahan. Sedangkan chrominance membawa informasi tentang kekuatan warna.
3. **Pengolahan Citra ( Image Processing )**

Pengolahan citra (*image processing*) adalah pengolahan suatu citra (gambar) dengan menggunakan komputer secara khusus, untuk menghasilkan suatu citra yang lain (Fadilsyah, 2007)

Sesuai dengan perkembangan komputer itu sendiri, pengolahan citra mempunyai dua tujuan utama, yaitu sebagai berikut:

1. Memperbaiki kualitas citra, dimana citra yang dihasilkan dapat menampilkan informasi secara jelas. Hal ini berarti manusia sebagai pengolah informasi *(human perception).*
2. Mengekstraksi informasi ciri yang menonjol pada suatu citra, dimana hasilnya adalah informasi citra dimana manusia mendapatkan informasi ciri dari citra secara numerik atau dengan kata lain *computer* (mesin) melakukan interprestasi terhadap informasi yang ada pada citra melalui besaran-besaran data yang dapat dibedakan secara jelas (berupa besaran numerik).

Dalam perkembangan lebih lanjut*, image processing* dan *computer vision* digunakan sebagai pengganti mata manusia dengan perangkat *input image capture* seperti kamera dan scanner dijadikan sebagai mata dan mesin komputer dijadikan sebagai otak yang mengolah informasi. Sehingga muncul beberapa pecahan bidang yang menjadi penting dalam *computer vision* antara lain: *pattern recognition* (pengenalan pola), *biometric* (pengenalan identifikasi manusia berdasarkan ciri-ciri biologis yang tampak pada badan manusia), *content based image and video retrieval* (mendapatkan kembali citra atau video dengan informasi tertentu), *video editing* dan lain-lain.

* + 1. **Pemodelan RGB**

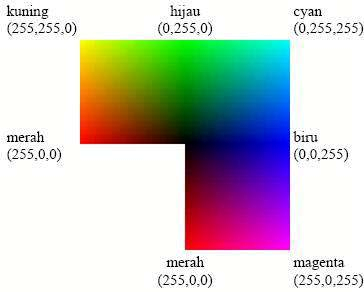
RGB adalah suatu model warna yang terdiri dari merah, hijau, dan biru, digabungkan dalam membentuk suatu susunan warna yang luas. Setiap warna dasar, misalnya merah, dapat diberi rentang-nilai. Untuk monitor komputer, nilai rentangnya paling kecil = 0 dan paling besar = 255. Pilihan skala 256 ini didasarkan pada cara mengungkap 8 digit bilangan biner yang digunakan oleh mesin komputer. Dengan cara ini, akan diperoleh warna campuran sebanyak 256 x 256 x 256 = 1677726 jenis warna. Sebuah jenis warna, dapat dibayangkan sebagai sebuah vektor di ruang 3 dimensi yang biasanya dipakai dalam matematika, koordinatnya dinyatakan dalam bentuk tiga bilangan, yaitu komponen-x, komponen-y, dan komponen-z. Misalkan sebuah vektor dituliskan sebagai r = (x,y,z). Untuk warna, komponen-komponen tersebut digantikan oleh komponen R(ed), G(reen), B(lue). Representasi dalam citra digital dinyatakan dalam persamaan :

fR(x,y) Σ [0…255]

fG(x,y) Σ [0…255]

fB(x,y) Σ [0…255]

(2.6)



**Gambar 2.3** Representasi Citra Warna

Sumber : Pillai, 2002

Gambar 2.3 menunjukkan piksel dari warna-warna yang dapat merupakan kombinasi dari tiga warna utama RGB, yaitu *red, green,* and *blue*.Jadi, sebuah jenis warna dapat dituliskan sebagai : warna = RGB(30, 75, 255), putih = RGB (255,255,255), sedangkan untuk hitam= RGB(0,0,0).

* + 1. **Konversi RGB ke Grayscale**

Citra *Grayscale* adalah citra yang hanya memiliki satu nilai kanal pada setiap pixelnya, dengan kata lain nilai bagian *Red = Green = Blue.* Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan dan putih. Tingkat keabuan di sini merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih. (Putra, Darma, 2010)

Pengubahan dari citra berwarna ke bentuk *grayscale* mengikuti aturan sebagai berikut :

(2.7)

* + 1. **Threshold**

Tresholding digunakan untuk mengatur jumlah derajat keabuan yang ada pada citra. Proses tresholding ini pada dasarnya adalah proses pengubahan kuantisasi pada citra untuk mendapatkan citra biner



**Gambar 2.4** Citra Biner

Gambar 2.4 memperlihatkan piksel yang memiliki derajat keabuan lebih kecil dari nilai batas yang ditentukan akan diberikan nilai 0, sementara piksel yang memiliki derajat keabuan yang lebih besar dari batas akan diubah menjadi bernilai 1

F (x, y)‘ = a1, f (x, y) < T

a2, f (x, y) ≥ T (2.8)

Pixel-pixel yang nilai intensitasnya di bawah 128 diubah menjadi hitam (nilai intensitas = 0), sedangkan pixel-pixel yang nilai intensitasnya di atas 128 diubah menjadi putih (nilai intensitas =1) (Masroeri, 2011)

* + 1. **Integral Proyeksi**

Integral proyeksi adalah suatu metode yang digunakan untuk mencari daerah lokasi dari obyek. Metode ini dapat kita gunakan untuk mendeteksi batas dari daerah gambar yang berbeda sehingga kita bisa mencari titik acuan pada Citra LPM. Integral ini juga di sebut dengan integral baris dan kolom dari pixel, karena integral ini menjumlahkan pixel per baris dan pixel per kolom. Dari metode ini akan dengan mudah untuk menemukan daerah lokasi objek yang diperlukan. Integral Proyeksi didefinisikan dengan

2.9

Di bawah ini merupakan contoh Perhitungan integral proyeksi :

**Gambar 2.5** Perhitungan nilai Pixel dengan Integral Proyeksi

Sumber : Basuki, Achmad 2006. *Mengenali Angka menggunakan*

*Fitur Bentuk Integral Proyeksi.* Surabaya : PENS – ITS

* + 1. **Reduksi Noise**

Pengkaburan citra sering digunakan dalam proses pelembutan citra yang bertujuan untuk menekan gangguan (noise) pada citra. Gangguan pada citra umumnya berupa variasi intensitas suatu pixel yang tidak berkolerasi dengan pixel-pixel tetangganya. Secara kasat mata, gangguan mudah dilihat oleh mata, karena tampak berbeda dengan pixel tetangganya.

Pengkaburan citra didapatkan dengan mengkonvolusikan citra dengan sebuah penapis (filter).

g (x,y) = f(x,y) \* h(x,y)

h (x,y) : fungsi penapis 2.10

* + 1. **Morfologi**

Pengolahan citra secar a morfo lo gi adalah alat untuk men gekstrak atau memod ifikasi informasi p ada bentuk dan struktur dari objek di d alam citra (Dou gherty , 2009). Op erator morfologi y an g umum digunak an adalah erosi dan dilasi, sedan gk an op erator lainny a merup akan p engemb an gan dar i keduany a.

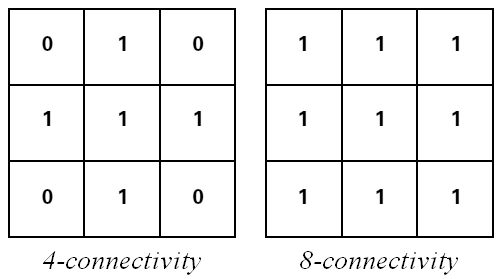
* + - 1. **Erosi**

Erosi adalah salah satu op erasi dasar dalam p emrosesan citra secara morfolo gi. Erosi adalah sebuah op erasi y ang men in gkatkan ukuran dar i latar belakan g (d an men gikis objek latar dep an) p ada citra biner (Dou gherty, 2009).

Dengan memisalk an *A* sebagai ob jek p ada citra masukan, *B* sebagai elemen terstruktur, dan *C* sebagai objek p ada citra keluaran hasil erosi, mak a p roses erosi dap at dinotasikan den gan:

*C = ( A Ө B )* 2.11

Erosi dilakukan d en gan b antuan elemen terstruktur. Elemen terstruktur membantu menentukan p iksel tetangga y ang akan ditelusuri den gan p roses erosi. Elemen terstruktur y ang umu m digun akan ad alah 4-kon ektivitas dan 8-konektivitas.



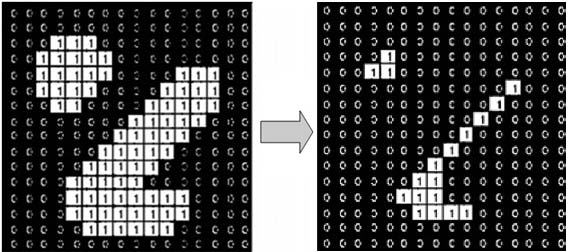
**Gambar 2.6** Elemen terstruktur 4- dan 8-konektivitas

Erosi memilik i karakteristik:

* Erosi p ada umumny a memp erkecil ukur an dari objek d an men ghilan gkan elemen atau anomali kecil den gan men guran gi objek dengan sebuah radius y ang lebih kecil dari ukuran elemen terstruktur.
* Dengan citra biner, erosi men gh ilan gk an objek y ang leb ih kecil dar i elemen terstruktur dan mengeliminasi p iksel p arameter dari objek citra y ang lebih besar.

Algoritma y an g dilakukan p ada dilasi adalah sebagai ber ikut:

1. Posisikan elemen terstruktur di bagian atas (menutup i) tiap -tiap p iksel dari citra masukan hingga titik p usat dari elemen terstruktur bertep atan dengan p osisi p iksel masukan.
2. Jika p aling sedikit satu p iksel p ada elemen terstruktur bertemu den gan p iksel latar belakan g di b awahny a (y ang ditutup iny a), maka tetap kan p iksel keluaran p ada citra baru ke nilai latar belak an g. Begitu juga jika bertemu dengan p iksel latar belakan g, maka tetap kan p iksel keluaran p ada citra baru ke nilai latar dep an.



**Gambar 2.7** Erosi den gan men ggunak an elemen terstruktur 8-konektivitas

Dengan begitu, p iksel latar belakang p ada citra masukan akan menjad i latar belakan g p ada citra keluar an dan ju ga p iksel latar dep an p ada citra masukan akan men jadi latar belak an g p ada citra keluaran.

* + - 1. **Dilasi**

Dilasi adalah salah satu op erasi dasar dalam morfolo gi matematika. Pad a citra bin er, dilasi adalah sebu ah op erasi y ang men geksp ansi atau memp erbesar ukuran dari objek latar dep an (Dougherty , 2009). Biasany a objek p ada citra dilamb an gkan d en gan p iksel p utih, walaup un untuk beberap a imp lementasi objek p ada citra dilamban gk an den gan p iksel hitam. Konektivitas antar p iksel p usat dengan tetanggany a dibuat berdasarkan elemen terstruktur y ang terdefinisi.

Dengan memisalk an *A* sebagai ob jek p ada citra masukan, *B* sebagai elemen terstrukstur, dan *C* sebagai objek p ada citra keluaran hasil dilasi, maka p roses dilasi dap at dinotasikan den gan:

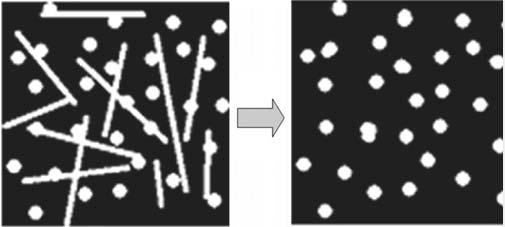
*C = ( A*⊕*B )* 2. 12

Dilasi memiliki kar akteristik:

* Dilasi p ada umu mny a memp erbesar ukur an dar i objek, men gisi luban g dan area y ang rusak, dan men ghubun gk an area y ang dip isahkan oleh jarak y ang lebih kecil dar i ukuran elemen terstruktur.
* Dengan citra biner, d ilasi men ghubun gkan area y ang dip isahkan oleh jar ak y ang lebih kecil d ari elemen terstruktur dan menambahkan p iksel ke p arameter dari setiap objek citra.
  + - 1. **Opening**

*Opening* didefinisik an sebagai p roses erosi y ang diikuti oleh p roses dilasi d en gan men ggunak an elemen terstruktur y ang sama untuk kedua op erasi (Dough erty , 2009).

Proses erosi y ang *merup* akan bagian dar i p roses ini men ghilan gkan p iksel latar dep an dari tep i daerah p iksel latar dep an tersebut, kemudian p roses dilasi mengembalikan ukur an dari p iksel y ang tersisa ke ukuran semula. Pada umumny a, *opening* digun akan untuk men geliminasi no ise y ang terlihat sebagai latar dep an.

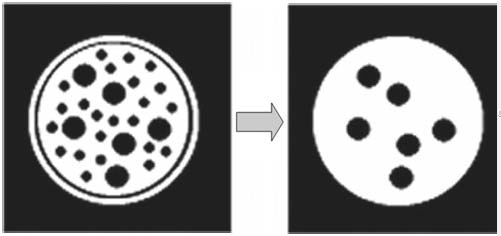


**Gambar 2.8** Proses *opening*

* + - 1. **Closing**

*Closing* didefin isikan sebagai p roses dilasi y ang d iikuti oleh p roses erosi dengan men ggunak an elemen terstruktur y ang sama untuk kedua op erasi (Dough erty , 2009).

Proses dilasi memp erbesar dan menghubun gk an p iksel-p iksel latar dep an, kemudian p roses erosi men gemb alik an ukuranny a k e ukuran semula. Proses *closing* berfungsi untuk men ghaluskan kontur dari ob jek latar dep an. Op erator ini menghubun gkan celah semp it dan mengelimin asi luban g-luban g k ecil y ang ad a p ada objek.

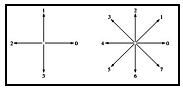


**Gambar 2.9** Proses *closing*

1. **Freeman Chain Code**

Chain code (kode rantai) pertama kali diperkenalkan untuk merepresentasikan kurva digital oleh Herbert Freeman, karena itu chain code disebut juga dengan Freeman code, sesuai dengan nama pencetus idenya (Shabab, dkk. 2009)

Chain code digunakan untuk menggambarkan batas obyek atau jumlah piksel yang berada dalam satu obyek. Batas obyek direpresentasikan dengan piksel-piksel yang saling terhubung dan memiliki nilai yang sama. Chain code mendeskripsikan sebuah obyek dengan segmen garis yang berurutan berdasarkan arah prioritas penelusuran yang telah ditetapkan. Arah dari tiap segmen direpresentasikan dengan angka tertentu. Elemen pertama pada sebuah urutan harus memberikan informasi mengenai posisinya sehingga rekonstruksi area atau perhitungan luas dapat dilakukan. Chain code berjalan dengan menelusuri piksel-piksel pada citra berdasarkan prioritas arah yang telah ditentukan. Sebuah chain code bisa terdiri dari 4 arah mata angin atau 8 arah mata. (Gonzalez,dkk. 2002)



**Gambar 2.10.** Chain code arah 4 mata angin dan 8 mata angin [8]

Chain code umumnya telah diterima dalam pengolahan citra digital karena menawarkan sejumlah keuntungan. Diantaranya adalah kelebihan untuk dapat mencari nilai dari keliling dan luas dari sebuah objek dari chain code. Selain itu, chain code juga dapat digunakan untuk memperhalus kontur dalam pengurangan derau (Shabab,dkk. 2009)

1. **Digitalisasi Citra**

Citra analog tidak bisa diproses langsung oleh komputer. Citra analog harus diubah menjadi citra digital (pencitraan) agar komputer bisa memprosesnya. Proses mengubah citra analog menjadi citra digital disebut digitalisasi citra. Ada dua hal yang harus dilakukan pada digitalisasi citra, yaitu digitalisasi spasial yang disebut juga sebagai sampling dan digitalisasi intensitas yang sering disebut sebagai kuantisasi.

Citra yang dihasilkan dari peralatan digital (citra digital) langsung bisa diproses oleh komputer karena di dalam peralatan digital sudah terdapat sistem sampling dan kuantisasi. Sedangkan peralatan analog tidak dilengkapi kedua sistem tersebut. Kedua sistem inilah yang bertugas memotong-motong citra menjadi x kolom dan y baris (proses sampling), sekaligus menentukan besar intensitas yang terdapat pada titik tersebut (proses kuantisasi) sehingga menghasilkan resolusi citra yang diinginkan (Sutoyo, T,dkk. 2009)

1. **Trigonometri**

Trigonometri (dari [bahasa Yunani](http://id.wikipedia.org/wiki/Bahasa_Yunani) *trigonon* = tiga sudut dan *metro* = mengukur) adalah sebuah cabang [matematika](http://id.wikipedia.org/wiki/Matematika) yang berhadapan dengan [sudut](http://id.wikipedia.org/wiki/Sudut) [segi tiga](http://id.wikipedia.org/wiki/Segi_tiga) dan [fungsi trigonometrik](http://id.wikipedia.org/wiki/Fungsi_trigonometrik) seperti [sinus](http://id.wikipedia.org/wiki/Sinus), [cosinus](http://id.wikipedia.org/wiki/Cosinus), dan [tangen](http://id.wikipedia.org/wiki/Tangen). Trigonometri memiliki hubungan dengan [geometri](http://id.wikipedia.org/wiki/Geometri), meskipun ada ketidaksetujuan tentang apa hubungannya; bagi beberapa orang, trigonometri adalah bagian dari geometri.

Dalam trigonometri kita mengenal beberapa fungsi trigonometri, antara lain sinus ( sin ), cosinus ( cos ), tangen ( tan ), secan ( sec ), cosecan ( cosec ), dan cotangen (cot ).

1. **Perangkat Lunak Pendukung**
   * 1. **Microsoft Visual Basic 6.0**

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang bersifat *event driven* dan menawarkan *Integrated Development Environment* (IDE) visual untuk membuat program aplikasi berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman Common Object Model (COM). Visual Basic merupakan turunan bahasa BASIC dan menawarkan pengembangan aplikasi komputer berbasis grafik dengan cepat, akses ke basis data menggunakan Data Access Objects (DAO), Remote Data Objects (RDO), atau ActiveX Data Object (ADO), serta menawarkan pembuatan kontrol ActiveX dan objek ActiveX. Beberapa bahasa skrip seperti Visual Basic for Applications (VBA) dan Visual Basic Scripting Edition (VBScript), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda.

Para programmer dapat membangun aplikasi dengan menggunakan komponen-komponen yang disediakan oleh Microsoft Visual Basic Program-program yang ditulis dengan Visual Basic juga dapat menggunakan Windows API, tapi membutuhkan deklarasi fungsi eksternal tambahan.

Dalam pemrograman untuk bisnis, Visual Basic memiliki pangsa pasar yang sangat luas. Dalam sebuah survey yang dilakukan pada tahun 2005, 62% pengembang perangkat lunak dilaporkan menggunakan berbagai bentuk Visual Basic, yang diikuti oleh C++, JavaScript, C#, dan Java.

Bill Gates, pendiri Microsoft, memulai bisnis softwarenya dengan mengembangkan interpreter bahasa Basic untuk Altair 8800, untuk kemudian ia ubah agar dapat berjalan di atas IBM PC dengan sistem operasi DOS. Perkembangan berikutnya ialah diluncurkannya BASICA (basic-advanced) untuk DOS. Setelah BASICA, Microsoft meluncurkan Microsoft QuickBasic dan Microsoft Basic (dikenal juga sebagai Basic Compiler).

Sejarah BASIC di tangan Microsoft sebagai bahasa yang diinterpretasi (BASICA) dan juga bahasa yang dikompilasi (BASCOM) membuat Visual Basic diimplementasikan sebagai gabungan keduanya.

Programmer yang menggunakan Visual Basic bisa memilih kode terkompilasi atau kode yang harus diinterpretasi sebagai hasil executable dari kode VB. Sayangnya, meskipun sudah terkompilasi jadi bahasa mesin, DLL bernama MSVBVMxx.DLL tetap dibutuhkan. Namun karakteristik bahasa terkompilasi tetap muncul (ia lebih cepat dari kalau kita pakai mode terinterpretasi).

Visual Basic merupakan bahasa yang mendukung OOP, namun tidak sepenuhnya. Beberapa karakteristik obyek tidak dapat dilakukan pada Visual Basic, seperti Inheritance tidak dapat dilakukan pada class module. Polymorphism secara terbatas bisa dilakukan dengan mendeklarasikan class module yang memiliki Interface tertentu. Visual Basic (VB) tidak bersifat case sensitif.

Visual Basic menjadi populer karena kemudahan desain form secara visual dan adanya kemampuan untuk menggunakan komponen-komponen ActiveX yang dibuat oleh pihak lain. Namun komponen ActiveX memiliki masalahnya tersendiri yang dikenal sebagai DLL hell. Pada Visual Basic .NET, Microsoft mencoba mengatasi masalah DLL hell dengan mengubah cara penggunaan komponen (menjadi independen terhadap *registry*).

* + 1. **Visual Basic Dot Net**

Visual Studio DOT NET menyediakan tools bagi para developer untuk membangun aplikasi yang berjalan di. Net Framework. VS. Net membawa perubahan besar dalam gaya pemrograman, karena setiap programmer dituntut untuk memahami .NET object model dan Object Oriented Programming dengan baik, jika tidak ingin menghasilkan aplikasi dengan performa rendah.

Dunia scripting yang akrab bagi programmer yang menggunakan SQL Server 2005 akan sulit ditemukan dalam. NET, karena pemrograman web sudah bersifat full object oriented, dengan fasilitas event driven programming sebagaimana layaknya windows programming. Pemrograman ini menjadi lebih mudah dan menyenangkan bagi para programmer windows, sedangkan anda para veteran scripting language sudah saatnya untuk beralih ke ASP.NET, yang dapat diprogram menggunakan VB, C#, C++ maupun Phyton dan COBOL sekalipun. Microsoft masih menyediakan Jscript.NET bagi anda para pecandu JavaScript dan JScript, sedangkan versi .Net dari VB.Script belum diketahui apakah akan disediakan atau tidak. Para veteran VB.Script disarankan untuk mempelajari VB, sehingga dapat menggunakan VB.Net untuk membangun aplikasi web. Bahasa pemrograman yang terdapat di VS.NET adalah VB.NET, C#, C++ .NET, J#, dan Jscript .NET. Dalam masa mendatang akan terus ditambah berbagai bahasa pemrograman lain, program yang di keluarkan beragam–ragam definisi. NET Framework, NET Framework merupakan “Man Behind The Gun” dari VB.NET sehingga kita perlu megetahui jeroan dari teknologi yang ada di dalamnya.

Microsoft .NET adalah software yang menghubungkan informasi, orang, sistem dan device Perangkat yang menjangkau client, server dan tool pengembang. Dot NET Framework, digunakan untuk membangun dan menjalankan semua macam softwere termasuk aplikasi berbasis Web, aplikasi smart client, dan layanan web XML (Extensible Markup Language). Menyediakan komponen untuk berbagai data melalui Network menggunakan komponen platform protokol independent seperti XML, SOAP, dan HTTP.

Tool–tool pengembang seperti Microsoft Visual Studio. NET yang penyediaan IDE (Integrated Development Enveronment lingkungan pengembangan terintegrasi) untuk memaksimalkan produktivitas pengembangan menggunakan NET Framework. Serangkaian server termasuk Microsoft Windows Server 2003,

* + 1. **MySQL**

MySQL merupakan salah satu software database ( basis data) *open source* yang dikembangkan sebuah komunitas bernama *MySQL AB* dengan tujuan membantu *user* untuk meyimpan data dalam tabel – tabel. Tabel terdiri atas *field* (kolom) yang mengelompokkan data – data berdasarkan kategori tertentu, misalnya nama, alamat, nomor telepon, dan sebagainya. Bagian lain dari tabel adalah *record* (baris) yang mencantumkan data yang sebenarnya.

MySQL sebagaimana *software database* lainnya, dapat menampung banyak schemata, dimana masing – masing schemata ini dapat digunakan oleh aplikasi – aplikasi yang berbeda, baik dari sisi tujuan maupun dari sisi bahasa pemrograman yang digunakan oleh masing – masing aplikasi yang bersangkutan.

Terdapat empat instruksi dasar yang digunakan dalam sql *( structured query language ),* yaitu:

a. *select* ( menampilkan data )

b. *insert* ( menginput atau menambah data )

c. *update* ( mengubah data )

d. *delete* ( menghapus data ) dalam *database.*

* + 1. **MySQL Connector**

MySQL Connector adalah driver ODBC untuk database MySQL. Tanpa ada driver tersebut, aplikasi VB tidak akan dapat mengakases database yang ada di MySQL. ODBC merupakan interface (antar muka) yang memungkinkan suatu aplikasi (seperti VB), yang digunakan untuk mengakses database pada sebuah RDBMS (seperti MySQL).

* + 1. **XAMPP**

XAMPP adalah aplikasi *web server* instan yang dibutuhkan untuk membangun aplikasi berbasis *web*. Fungsi XAMPP adalah sebagai *server* yang berdiri sendiri ( localhost ), yang terdiri atas program Apache, http *server*, MySQL, *database,* dan penterjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

Nama XAMPP merupakan singkatan dari X (X=Cross Platform), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam lisensi GNU *( General Public License )* dan gratis. Dengan menginstal XAMPP, kita tidak perlu menginstal aplikasi *server* satu persatu karena di dalam XAMPP sudah terdapat :

a) Apache 2.2.14 ( Ipv6 Enabled) + open SSL 0.9.8l

b) MySQL 5.1.41 + PBXT engine

c) PHP 5.3.1

d) PHPMyAdmin 3.2.4

e) Perl 5.10.1

f) Filezilla FTP Server 0.9.33.

g) Mercury Mail Transport System 4.72.

* + 1. **CorelDraw X3**

Program grafis ini memang begitu besar pengaruhnya terhadap perkembangan desain komputer pada umumnya. Banyak pengguna yang menyatakan bahwa perkembangan CorelDraw telah begitu pesat mempengaruhi perkembangan seni grafis.

Pemanfaatan CorelDraw telah merambah kesemua bidang tidak hanya desain grafis semata, sehingga pemakaian CorelDraw semakin banyak diminati karena keluwesannya serta kemudahannya dalam hal pemakaian tool yang disediakan untuk memenuhi segala kebutuhan publikasi.

Melalui CorelDraw Anda dapat membentuk ilustrasi sesuai keinginan. Ilustrasi tersebut dapat berupa satu objek gambar atau lebih dan membentuk suatu lukisan. Adapun objek gambar yang dapat dibentuk sangatlah beragam,mulai dari gambar sederhana seperti persegi panjang, kotak, teks, dan lain sebagainya hingga objek gambar yang komplek yang merupakan gabungan dari beberapa objek gambar sederhana melalui. proses tertentu. Tujuannya

**BAB III**

**METODE PENELITIAN**

* 1. **Jenis Penelitian**

Jenis penelitian yang dilakukan pada penelitian ini adalah penelitian *deskriptif*, yaitu suatu bentuk penelitian yang ditujukan untuk mendeskripsikan fenomena-fenomena yang ada, baik fenomena alamiah maupun fenomena buatan manusia. Fenomena itu bisa berupa bentuk, aktivitas, karakteristik, perubahan, hubungan, kesamaan, dan perbedaan antara fenomena yang satu dengan fenomena lainnya (Sukmadinata, 2006:72).

Dalam skripsi ini secara garis besar metode penelitian terbagi menjadi 2 yaitu metode pengumpulan data dan metode pengembangan perangkat lunak.

* 1. **Rancangan Umum**

Dibawah ini adalah bagan yang menggambarkan tahap dari kegiatan penelitian yang dilakukan. Kegiatan penelitian yang dilakukan dalam penyusunan tugas akhir adalah sebagai berikut.



**Gambar 3.1**. Flow Chart Garis Besar Perencanaan Program

1. Identifikasi Masalah

Identifikasi masalah merupakan tahap awal dari penelitian ini, yaitu mengidentifikasi cara mengenerate dan mengekstrak data pada citra presensi mahasiswa menggunakan algoritma pengolahan citra.

1. Studi Literatur

Setelah melakukan identifikasi masalah kemudian dilanjutkan dengan studi pustaka terhadap catatan atau arsip yang berkaitan dengan metode, alat dan bahan yang digunakan pada penelitian makalah, jurnal ilmiah, buku dan juga pemanfaatan internet sebagai sumber informasi, dengan jalan melihat informasi yang disediakan oleh situs-situs web, forum diskusi dan sebagainya.

1. Pengumpulan Data

Pengumpulan data berguna untuk mengumpulkan data-data yang digunakan sebagai bahan untuk penelitian. Data-data yang digunakkan dalam penelitian ini yaitu mencari responden sebanyak 3 orang bagian tata usaha untuk mencoba SIM Generate dan Deteksi Bubble. Total keseluruhan data citra yang didapat sebanyak 25. Data citra dibagi menjadi 2 bagian, 80% untuk data pengujian dan 20% untuk data penelitian.

1. Pra-proses Generate Bubble

Pada tahap pra-proses Generate Bubble, yang pertama dilakukan adalah akuisisi citra yang berfungsi untuk pengambilan data sebagai bahan baku penelitian. Dalam hal ini dilakukan pengambilan citra atau gambar yang dijadikan sebagai bahan pemrosesan citra. Citra yang akan digunakan sebagai bahan baku penelitian adalah citra template lembar presensi mahasiswa yang sudah dalam bentuk citra digital dengan format citra *bitmap* (\*.bmp), yang kemudian dijadikan template untuk mengenerate data presensi mahasiswa yang diambil dari tabel perkuliahan untuk dicetak sebagai lembar presensi mahasiswa.

1. Pra-proses Deteksi Bubble

Pada tahap pra-proses Deteksi Bubble, yang pertama dilakukan adalah akuisisi citra yang berfungsi untuk pengambilan data sebagai bahan baku penelitian. Dalam hal ini dilakukan pengambilan citra atau gambar yang dijadikan sebagai bahan pemrosesan citra. Citra yang akan digunakan sebagai bahan baku penelitian adalah citra template lembar presensi mahasiswa yang sudah dalam bentuk citra digital dengan format citra *bitmap* (\*.bmp), bila citra tersebut bukan berformat *bitmap* (\*.bmp) maka akan diubah menjadi citra berformat *bitmap* (\*.bmp). Setelah citra berformat *bitmap* (\*.bmp), dilakukan pengubahan ukuran citra menjadi 850 x 1202 pixel. Pengubahan ukuran dilakukan agar data citra yang akan diproses tidak terlalu besar dan seragam. Langkah selanjutnya adalah transformasi warna yaitu pengubahan data citra menjadi citra *Grayscale* yang kemudian dilanjutkan dengan proses binerisasi citra menggunakan metode *Thresholding*. Proses transformasi warna bertujuan untuk mempercepat dan mempermudah pengolahan citra. Langkah terakhir adalah menyimpan citra dalam format BMP.

1. Reduksi Noise

Setelah melalui tahap pra-proses, data citra dilakukan tahap reduksi *noise* yang berguna untuk memperbaiki kualitas citra sehingga mudah untuk dilakukan analisa. Metode reduksi *noise* yang digunakan adalah morfologi – *opening* alasan penggunaan kedua metode tersebut adalah berdasarkan karakteristik *noise* yang ada*.*

1. Normalisasi Data

Setelah data citra direduksi *noisenya*, langkah selanjutnya adalah menormalisasikan data citra tersebut ke dalam bentuk normal ketika citra tersebut dideteksi miring. Dalam skripsi ini menggunakan metode pencarian gradien garis lurus berdasarkan 2 buah nilai (x,y) dari titik acuan (*mark*).

1. Deteksi Data Presensi

Setelah data baru dihasilkan dari normalisasi, data citra tersebut dilakukan analisa bentuk (*shape*) berdasarkan luas, mendapatkan posisi (x,y) dari bentuk (*shape*) serta pemberian label pada bentuk (*shape*) tersebut. Hasil dari proses ini yaitu bentuk (*shape*) yang telah diberi label kemudian dilakukan pencocokan dengan data *template* sehingga diketahui informasi yang terkandung pada bentuk (*shape*) tersebut. Informasi disimpan dalam database MySQL.

1. Pengujian

Setelah tahap diatas telah dilakukan, semua data citra hasil dilakukan pengujian. Data yang diuji berupa 25 data citra hasil *scan* lembar presensi mahasiswa yang telah terisi.

1. Dokumentasi

Setelah melalui tahap pengujian, lalu dilakukan dokumentasi yang merupakan tahapan akhir dari penelitian. Dokumentasi diperlukan untuk menyimpan dan mengumpulkan data-data hasil penelitian yang dijadikan sebagai acuan.

* 1. **Perangkat Penelitian**

Alat pendukung dalam pembuat sistem ini terdiri dari perangkat keras dan perangkat lunak, diantaranya :

Spesifikasi perangkat keras yang digunakan:

1. Satu unit laptop dengan spesifikasi sebagai berikut:

* Prosesor : Intel® Atom ™ CPU 330 1,60GHz
* RAM : 2GB DDR2
* Resolusi Monitor : 12”
* Harddisk : 160GB HDD

2. Template Citra Lembar Presensi Mahasiswaukuran A4 (210 x 297 mm).

3. *Scanner .*

4. Kabel USB penghubung PC dengan *scanner.*

Spesifikasi perangkat lunak yang digunakan:

1. *Visual Basic.Net 2008 Express Edition*
2. *Microsoft Visual Basic 6.0*
3. *CorelDraw X3*
4. *Xampp for Windows*
5. *MySQL Connector Net 6.5.4*
6. *MySQL ODBC Connector*
7. *Driver scanner*
8. Sistem operasi *Windows XP Professional* SP 3
   1. **Waktu dan Lokasi Penelitian**

Penelitian ini dilakukan selama 4 bulan, yaitu dari bulan Januari 2013 sampai bulan Juli 2012. Sedangkan tempat penelitian dan pengujian dilakukan di 3 tempat berbeda yaitu di tempat tinggal penulis di daerah Krembung, tempat tinggal Bapak Yusron Rizal selaku pembimbing skripsi dan lab Multimedia di lantai 2 di gedung STMIK Yadika Bangil.

* 1. **Fokus Penelitian**

Fokus penelitian ini adalah:

1. Generate Bubble data Presensi Mahasiswa pada Citra LPM
2. Deteksi Bubble pada Citra LPM.
3. Pengaruh nilai ambang *threshold* terhadap proses reduksi *noise*.
4. Pengaruh *noise* terhadap keakuratan hasil analisa bentuk (*shape)*.
5. Pengaruh kemiringan citra terhadap keakuratan hasil ekstraksi data.
   1. **Rencana Pengujian Sistem**

Glen Myers dalam buku klasiknya mengenai pengujian perangkat lunak menyatakan sejumlah aturan yang berfungsi sebagai sasaran pengujian pada perangkat lunak adalah:

1. Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
2. *Testcase* yang baik adalah *testcase* yang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.
3. Pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya.

Pengujian B*lack box* berfokus pada persyaratan fungsional perangkat lunak untuk melihat apakah fungsi-fungsi dasar aplikasi berjalan sebagaimana mestinya.(Benartha dkk, 2011).

Rencana Pengujian Sistem Informasi Lembar Presensi Mahasiswa Otomatis Menggunakan Bubble Detector dapat dilihat pada tabel berikut.

|  |  |  |
| --- | --- | --- |
| **Kelas Uji** | **Butir Uji** | **Jenis Pengujian** |
| Praproses Generate Bubble | Akuisi Citra template, baca database, Generate Bubble, Cetak LPM | *Black Box* |
| Praproses Deteksi bubble | Akuisisi, merubah format, merubah ukuran dan segmentasi citra lembar Presensi Mahasiswa | *Black Box* |
| Reduksi Noise | Operasi morfologi – *opening* | *Black Box* |
| Deteksi Kemiringan | Sudut kemiringan dan rotasi citra Lembar Presensi Mahasiswa | *Black Box* |
| Ekstraksi Informasi | *Boundary detection,* analisa objek dengan *Chain Code,* deteksi *markVertical* dan *markHorizontal,* dan ekstraksi data lembar presensi mahasiswa | *Black Box* |

**Tabel 3.1** Rencana Pengujian Sistem Informasi

* 1. **Review Riset Relevan**

Junior Silalahi mengembangkan perangkat lunak Rekapitulasi Absensi Berdasarkan Scanning Dan Pengolahan Citra Digital[[1]](#footnote-2). Sistem ini bekerja dengan cara memeriksa tanda tangan mahasiswa pada Lembar absensi setelah itu di-scan lalu hasil scan-nya akan diolah oleh aplikasi untuk menghasilkan informasi kehadiran mahasiswa yang kemudian ditampilkan dalam file excel berupa jumlah jam mahasiswa yang tidak hadir.

**BAB IV**

**PERANCANGAN SISTEM**

1. **Analisa Sistem Lama**

Berdasarkan hasil penelitian terdahulu yang dilakukan oleh Renita Silviya (2011) pada perancangan sistem informasi akademik mengindikasikan kepada pihak manajemen secara khusus untuk memperhatikan langkah-langkah dalam perencanaan perancangan sistem, sehingga menghasilkan sistem yang mampu mengakomodir segala kebutuhan administrasi akademik, sehingga hasil perancangan sistem akademik dapat menyelesaikan permasalahan-permasalahan yang di hadapi, terutama memberikan solusi untuk mengatasi masalah keterlambatan dalam pendistribusian informasi dan kemudahan mengakses informasi bagi semua kalangan pada lembaga akademik. Sistem yang di bangun mencakup bagian administrasi akademik, kendala yang dihadapi terdapat pada proses entry data presensi mahasiswa yang mana sudah terkomputerisasi namun pengentrian data presensi masih manual, dalam artian bagian BAK harus menginputkan data kehadiran tiap mahasiswa, dalam Perancangannya Sistem lama menggunakan database MySQL dan aplikasi dengan pemrograman PHP, berikut DFD Level I Proses Perkuliahan pada Sistem Informasi Akademik



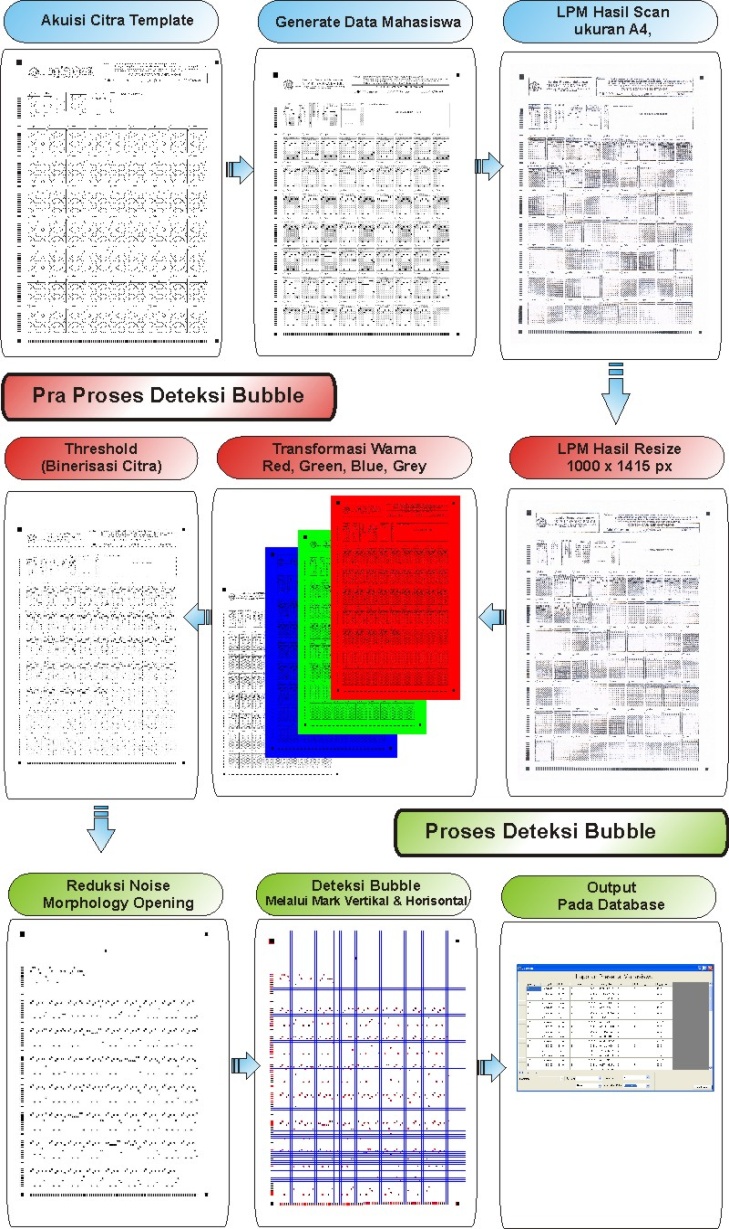
**Gambar 4.1** DFD level 1 Proses Perkuliahan Sistem Lama

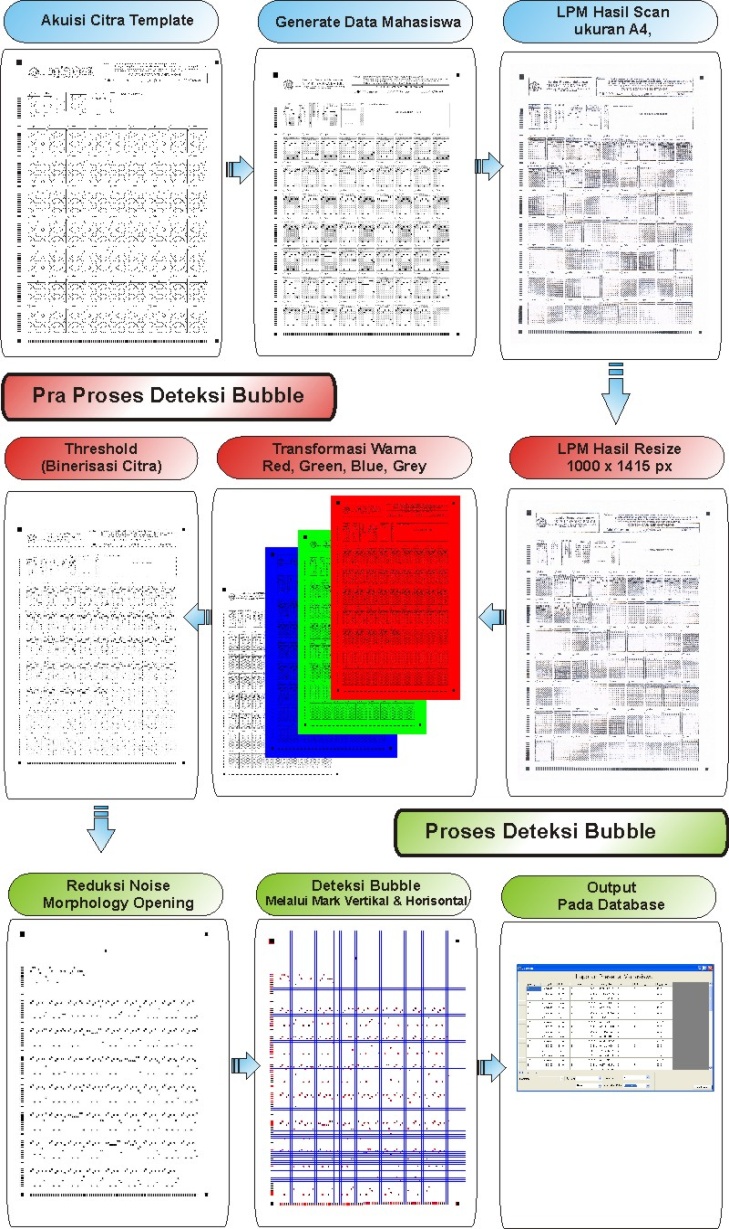
DFD yang terdapat pada kotak merah merupakan bagian proses yang akan dikembangkan oleh peneliti, yang meliputi proses pembuatan presensi, pendataan presensi kuliah dan cetak form presensi mahasiswa.

1. **Perancangan Pengembangan Sistem**
2. **Analisa Sistem Baru**

Pada penelitian ini akan dibangun sebuah Sistem Informasi Lembar Presensi Mahasiswa Otomatis Menggunakan Bubble Detektor otomatis yang diharapkan akan membuat sistem informasi presensi mahasiswa menjadi lebih efektif dan efisien karena Admin/BAK tidak perlu menginputkan satu persatu data kehadiran mahasiswa, hanya men-*scan-*kan lembar presensi mahasiswa pada *scanner* dan sistem yang secara otomatis akan memeriksa data pada lembar presensi mahasiswa tersebut.

Sistem ini dibangun untuk menganalisis citra lembar presensi mahasiswa dan menghasilkan output sebuah nilai, kemudian nilai tersebut akan disimpan dalam database stmik\_yadika.Sistem ini dapat digambarkan sebagai berikut:



**

**Gambar 4.2** Konsep Sistem Informasi LPM

* Pada tahap pra-proses Generate Bubble, yang pertama dilakukan adalah akuisisi citra template lembar presensi mahasiswa yang sudah dalam bentuk citra digital dengan format citra *bitmap* (\*.bmp), yang kemudian dijadikan template untuk mengenerate data presensi mahasiswa yang diambil dari t\_krs dan t\_jadwal\_kuliah untuk dicetak sebagai lembar presensi mahasiswa.
* Untuk pinginputkan data presensi dilakukan proses akuisi citra melalui *scanner*, yang sebelumnya Lembar Presensi Mahasiswa sudah diisi (dihitamkan) oleh Dosen ataupun Mahasiswa.
* Data berupa citra digital lembar presensi mahasiswa yang telah melalui pra proses yaitu pengubahan format citra menjadi *bitmap* dan pengubahan ukuran citra menjadi 850 x 1202 piksel. Selanjutnya dilakukan transformasi warna pada citra tersersebut untuk mempercepat dan mempermudah algoritma pengolahan citra. Proses selanjutnya berupa beberapa algoritma pengolahan citra dengan dengan menggunakan algoritma yang berkaitan yaitu *thresholding,* operasi morfologi *– opening* dan *chain code,* dan untuk uji cobanya adalah program mampu *mengenerate* data mahasiswa pada LPM sesuai dengan database serta mampu *mengekstrak* data kehadiran mahasiswa sehingga bisa digunakan untuk input data presensi mahasiswa pada table perkuliahan.
  + - Outputnya adalah data laporan kehadiran yang kemudian data tersebut akan disimpan dalam database

Sistem ini dibangun dengan menggunakan bahasa pemrograman *Visual Basic* sedangkan untuk mendesain citra lembar presensi mahasiswa menggunakan software *CorelDraw X3*.

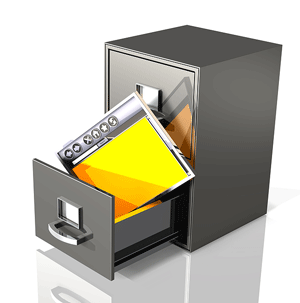
1. **Perancangan Proses**

Menurut Jogiyanto H.M (2001:196), Perancangan Sistem yaitu: “Perancangan Sistem dapat didefenisikan sebagai penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi”.

Perancangan sistem yang digunakan dalam penelitian ini antara lain analisa sistem yang telah ada dan digambarkan dalam bentuk Desain Arsitektur Sistem dan document flow. Selanjutnya document flow tersebut digambarkan dengan model terstruktur disain analis (DFD)

1. **Desain Arsitektur Sistem**

Secara garis besar Pengembangan Pengembangan Sistem Informasi Deteksi Presensi Otomatis Menggunakan *Bubble Detector* ini adalah terdiri 7 proses utama yaitu pembacaan Data Perkuliahan, Generare Value pada LPM, Cetak LPM, Scanning dan Ekstraksi citra LPM yang kemudian hasil presensi disimpan pada Tabel Presensi. Desain sistem ini dapat dilihat pada gambar.4.3



Convert

P\_LPM to D\_LPM

Isi Presensi

Cetak LPM

Generate LPM

Tabel

Perkuliahan

Simpan data

di Tabel Presensi

Extraksi Bubble Citra LPM

**Gambar 4.3**

Desain Sistem Informasi Deteksi Presensi Otomatis

Menggunakan *Bubble Detector*

Secara singkat, kerja perangkat lunak ini adalah :

* 1. Table Perkuliahan akan dieksekusi oleh komputer yang kemudian dijadikan *input*-an untuk di-*generate* di template lembar presensi dan dicetak sebagai LPM,
  2. Setelah pengisian data presensi oleh Dosen/Mahasiswa dilakukan, dilanjutkan proses Scanner yang dihubungkan ke PC menggunakan kabel USB akan memindai / menscan LPM kemudian mentransmisikan ke PC sebagai citra *input*-an yang disimpan dalam satu folder Citra LPM
  3. PC akan memproses citra LPM tersebut dengan menggunakan algoritma pengolahan citra berupa metode *chain-code* untuk menghasilkan keluaran berupa data, kemudian data tersebut akan disimpan dalam database.

1. **Diagram Block**

Isi Data LPM

Scan

Print

Generate LPM

T\_Perkuliahan

Chain code

Image Thresholding

Akuisi Citra

Normalisasi Acuan

Hasil Deteksi

Simpan T\_Presensi

Identifikasi Posisi Bubble

**Gambar 4.4.**

Block diagram Sistem Informasi Deteksi Presensi Otomatis

Menggunakan *Bubble Detector*

1. **Document Flow**

Document Flow ini akan menjelaskan tentang bagaimana alur dari Mencetak Presensi hingga pelaporan data presensi

Pada Gambar 4.1 menunjukkan document flow dari proses presensi



**Gambar 4.5** Document Flow (Proses Presensi Mahasiswa)

1. **Data Flow Diagram (DFD)**

*Data Flow Diagram* (DFD) menunjukkan aliran data dalam Pengembangan Sistem Informasi Lembar Presensi Mahasiswa Otomatis menggunakan Bubble Detector. Tahapan pertama pada perancangan dengan menggunakan DFD adalah membuat susunan gambar utama yang merupakan gambaran *Context Diagram.* Gambaran *Context Diagram* dari Pengembangan Sistem Informasi Lembar Presensi Mahasiswa Otomatis menggunakan Bubble Detector dapat dilihat pada Gambar 4.15. Dalam diagram konteks tersebut terdapat 2 (dua) entitas antara lain : BAAK dan Manajemen

0.wmf

**Gambar 4.6** Diagram Konteks Sistem Informasi LPM Otomatis

Dari diagram konteks diatas, dapat diuraikan menjadi diagram level 1 seperti Gambar 4.5. Dalam diagram level 1 tersebut, Pengembangan Sistem Informasi LPM terdiri atas tiga proses utama, yaitu proses generate citra LPM , proses input presensi mahasiswa dan proses sistem pelaporan.

1.wmf

**Gambar 4.7** Diagram Lv.1 Sistem Informasi LPM Otomatis

Dari DFD level 1 diatas terdapat beberapa proses yang dapat diuraikan hingga diagram level 2. Seperti yang dapat dilihat pada Gambar 4.8, merupakan hasil penguraian dari Proses Generate Citra LPM pada DFD level 1. Pada level ini, proses Generate Citra LPM digital terbagi menjadi tiga subproses. Pertama adalah subproses Load Citra LPM yaitu user menentukan Template Citra LPM yang akan digenerate dan subproses kedua Proses Algoritma Generate Bubble, dimana pada proses ini data presensi akan digenerate secara otomatis ke dalam citra Template LPM yang sudah diload dan dilanjutkan pada subproses ketiga yaitu Cetak Form Lembar Presensi Mahasiswa untuk mencetak hasil Generate.dfd lv2 Generate Bubble.wmf

**Gambar 4.8** DFD level 2 Proses Generate Citra

Dari DFD di atas data presensi mahasiswa yang diambil dari tabel t\_krs dan t\_jadwal\_kuliah akan digenerate pada Citra Template sehingga bisa diteruskan kedalam proses cetak form pengisin presensi

dfd lvl 2 input presensi.wmf

**Gambar 4.9** DFD level 2 Proses Input Hasil Presensi

Pada Gambar 4.9, merupakan hasil penguraian dari subproses input Presensi mahasiswa, di dalam sub proses tersebut tardapat sub 2 proses diantaranya proses memasukkan LPM digital dan sub proses algoritma Bubble Detector

Pada Gambar 4.10, merupakan hasil penguraian dari algoritma *bubble* Detektor. Pada level ini, algoritma pengekstrak *bubble* LPM terbagi menjadi empat subproses. Pertama adalah subproses Preproses yaitu proses menampilkan data citra LPM yang akan diekstrak kemudian mengubah format dan ukuran citra tersebut, subproses Segmentasi yaitu mengelompokkan warna citra menjadi dua kelompok berdasarkan kriteria yang telah ditentukan sehingga menghasilkan citra biner, subproses Reduksi noise yaitu proses Morfologi Opening data citra lembar presensi sebelum proses ekstraksi informasi dan subproses Ekstraksi informasi yaitu proses mengekstrak identitas pada data citra lembar presensi mahasiswa.

dfd lvl3 bubble detektor.wmf

**Gambar 4.10.** DFD level 3 Algoritma pengekstrak *bubble* LPM

Dari DFD level 3 diatas terdapat beberapa proses yang dapat diuraikan hingga diagram level 4. Seperti yang dapat dilihat pada Gambar 4.11, merupakan hasil penguraian dari proses Preproses pada DFD level 3 diatas. Pada level ini, proses Preproses terbagi menjadi dua subproses. Pertama adalah subproses Akuisisi citra yaitu proses menampilkan data citra LPM yang akan diekstrak dan subproses Ubah ukuran citra yaitu proses mengubah format citra menjadi *bitmap* dan mengubah ukuran citra menjadi 850 X 1202 pixel.

dfd lvl4 preproses.wmf

**Gambar 4.11.** DFD level 4 Preproses

Pada Gambar 4.12, merupakan hasil penguraian dari proses Segmentasi pada DFD level 3 gambar 4.10 diatas. Pada level ini proses Segmentasi terbagi menjadi dua subproses. Pertama adalah subproses GetRGB yaitu proses mendapatkan nilai RGB dari data citra LPM yang akan diekstrak dan subproses *Thresholding* yaitu proses mengelompokkan warna citra menjadi dua kelompok setelah mendapatkan nilai *grey* dari subpreses RGBto Grey berdasarkan nilai *Threshold* yang telah ditentukan sehingga menghasilkan citra biner.

dfd lvl4 segmentasi.wmf

**Gambar 4.12.** DFD level 4 proses Segmentasi

Pada Gambar 4.13, merupakan hasil penguraian dari proses Ekstraksi informasi pada DFD level 2 gambar 4.9 diatas. Pada level ini, proses Ekstraksi informasi terbagi menjadi lima subproses. Pertama adalah subproses Analisa bentuk dengan *chain code* yaitu proses menggambarkan batas objek atau jumlah piksel yang berada dalam satu objek pada data citra LPM yang akan diekstrak yang berguna untuk mendapatkan luas objek, posisi X,Y titik tengah objek dan nilai minimal dan maksimal X dan Y objek, dan subproses Deteksi acuan bebas yaitu proses pencarian titik acuan dengan cara mengurutkan semua obyek yang telah terdeteksi berdasarkan luas area terbesar atau area dengan jumlah piksel terbanyak kemudian mencari titik acuan yang kedua yaitu dengan cara mencari obyek yang mempunyai posisi y terdekat dengan obyek dengan luas area terbesar, subproses Normalisasi data citra LPM digital yaitu proses pencarian derajat kemiringan (gradien) data citra LPM berdasarkan posisi (x,y) kedua obyek tersebut kemudian merotasi data citra LPM tersebut jika derajat kemiringan lebih besar dari 0 (nol), subproses Deteksi dan rekam *mark* yaitu proses mendapatkan posisi *mark* vertikal yang berjumlah 97 buah dan *mark* horizontal yang berjumlah 91 buah dan menyimpannya ke dalam array, dan subproses Ekstraksi identitas presensi mahasiswa yaitu proses mendapatkan informasi dari *bubble* dengan membentuk koordinat posisi berdasarkan posisi *mark* vertikal dan *mark* horisontal kemudian mengekstrak data presensi untuk perhitungan perekapan presensi guna ditampilkan dan disimpan pada database.

dfd lvl 4 extraksi informasi.wmf

**Gambar 4.13.** DFD level 4 proses Ekstraksi informasi

Dari diagram level 1 gambar 4.7 dari algoritma tersebut, terdapat satu sub proses yang masih dapat diuraikan hingga level 2, yaitu proses Sistem Pelaporan Presensi Mhs yang dapat dilihat pada Gambar 4.14. Proses tersebut dapat dibagi menjadi dua subproses, yaitu subproses pencarian data, subproses seleksi db dan subproses cetak laporan presensi.

dfd lvl2 pelaporan.wmf

**Gambar 4.14.** DFD level 2 proses Pelaporan

1. **Perancangan Data**

Perancangan data adalah aktivias pertama dan beberapa sering mengatakan yang terpenting dari empat aktivitas perancangan yang dilakukan selama pengembangan sistem. Pengaruh struktur data pada struktur program dan kompleksitas procedural menyebabkan perancangan data berpengaruh penting terhadap kualitas perangkat lunak.

Konsep informasi dan abstraksi data memberikan dasar bagi peneliti terhadap perancangan data. Tanpa melihat teknik perancangan yang digunakan, data yang didesain dengan baik dapat membawa kepada struktur program dan modularitas yang baik serta mengurangi kompleksitas prosedural.

Berikut tahapan untuk perancangan data, antara lain:

1. ***Conceptual Data Model* (CDM)**

Pada *Conceptual Data Model* (CDM) ini terdapat 3 tabel. Untuk lebih jelasnya, CDM dapat ditunjukkan pada gambar 4.15.

SEMESTER

menentukan

mengambil

T\_KRS

ID\_KRS

NIM

NAMA\_MHS

ANGKATAN

PRODI

JURUSAN

SEMESTER

KELAS

KODE\_MK

TH\_AJARAN

DOSEN\_WALI

T\_Perkuliahan

ID

NIDN

PRODI

JURUSAN

KODE\_MK

SEMESTER

KELAS

DOSEN\_PENGAMPUH

PERTEMUAN

NIM

NAMA\_MHS

MATA\_KULIAH

SKS

KEHADIRAN\_MHS

KEHADIRAN\_DOSEN

KD\_RUANG

KETERANGAN

UTS

UAS

TH\_AJARAN

KUIS1

KUIS2

TUGAS1

TUGAS2

TUGAS3

TUGAS4

TUGAS5

TUGAS6

TUGAS7

TUGAS8

TUGAS9

TUGA10

TOTAL\_NILAI

T\_Jadwal\_Kuliah

ID

PRODI

KELAS

KODE\_MK

MATA\_KULIAH

SKS

HARI

JAM

KD\_RUANG

DOSEN\_PENGAMPUH

TH\_AJARAN

**Gambar 4.15**

*Conceptual Data Model* (CDM) Sistem Informasi Deteksi Presensi

1. **Struktur Tabel**

Berikut ini akan diuraikan struktur tabel yang nantinya digunakan dalam pembuatan sistem.

1. Nama Tabel : t\_jadwal\_kuliah  
   Fungsi Tabel : Untuk menyimpan data jadwal Primary Key : ID

Foreign Key : -

Tabel 4.1 t\_ jadwal\_kuliah

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Type** | **Constraint** |
| ID | Integer | Primary key |
| JENJANG | Varchar(3) |  |
| SEMESTER | Varchar(3) |  |
| KODE\_MK | Varchar(10) |  |
| MATA\_KULIAH | Varchar(50) |  |
| SKS | Integer |  |
| KD\_RUANG | Varchar(7) |  |
| KELAS | Varchar(2) |  |
| HARI | Varchar(10) |  |
| JAM | Varchar(15) |  |
| NIDN | Varchar(30) |  |
| DOSEN\_PENGAMPUH | Varchar(30) |  |
| TH\_AJARAN | Varchar(6) |  |

1. Nama Tabel : t\_krs

Fungsi Tabel : Untuk mengambil data mahasiswa yang telah mengisi kartu rencana studi

Primary Key : ID\_KRS

Foreign Key : -

Tabel 4.2 t\_ krs

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Type** | **Constraint** |
| ID\_KRS | Integer | Primary key |
| JENJANG | Varchar(3) |  |
| PRODI | Varchar(50) |  |
| SEMESTER | Varchar(3) |  |
| KELAS | Varchar(2) |  |
| ANGKATAN | Integer |  |
| NIM | Varchar(12) |  |
| NAMA\_MHS | Varchar(40) |  |
| KODE\_MK | Varchar(10) |  |
| TH\_AJARAN | Varchar(6) |  |
| DOSEN\_WALI | Varchar(30) |  |

1. Nama Tabel : t\_perkuliahan  
   Fungsi Tabel : Untuk menyimpan data perkuliahan setiap mahasiswa

Primary Key : ID

Foreign Key : -

Tabel 4.3 t\_ perkuliahan

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Type** | **Constraint** |
| ID | Integer | Primary key |
| NIDN | Varchar(30) |  |
| DOSEN\_PENGAMPUH | Varchar(30) |  |
| PERTEMUAN | Varchar(2) |  |
| JENJANG | Varchar(3) |  |
| PRODI | Varchar(50) |  |
| SEMESTER | Varchar(3) |  |
| KELAS | Varchar(2) |  |
| KODE\_MK | Varchar(10) |  |
| MATA\_KULIAH | Varchar(50) |  |
| SKS | Integer |  |
| KD\_RUANG | Varchar(7) |  |
| NIM | Varchar(12) |  |
| NAMA\_MHS | Varchar(40) |  |
| KEHADIRAN\_MHS | Integer |  |
| KEHADIRAN\_DOSEN | Integer |  |
| KUIS1 | Float(5,2) |  |
| KUIS2 | Float(5,2) |  |
| TUGAS1 | Float(5,2) |  |
| TUGAS2 | Float(5,2) |  |
| TUGAS3 | Float(5,2) |  |
| TUGAS4 | Float(5,2) |  |
| TUGAS5 | Float(5,2) |  |
| TUGAS6 | Float(5,2) |  |
| TUGAS7 | Float(5,2) |  |
| TUGAS8 | Float(5,2) |  |
| TUGAS9 | Float(5,2) |  |
| TUGAS10 | Float(5,2) |  |
| UTS | Float(5,2) |  |
| UAS | Float(5,2) |  |
| TOTAL\_NILAI | Float(5,2) |  |
| KETERANGAN | Varchar(50) |  |

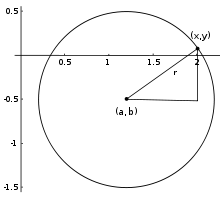
1. **Proses Generate LPM**

Proses Generate LPM merupakan proses dasar dalam mencetak lembar presensi mahasiswa dimana data yang digunakan adalah mengambil dari tabel t\_krs dan t\_jadwal\_kuliah pada database Sistem Informasi Akademik, kemudian data presensi seperti jadwal perkuliahan dan data mahasiswa digenerate sesuai dengan koordinat pada citra Template Lembar Presensi mahasiswa yang sudah diketahui dari Software CorelDraw X3.

Tujuan men-generate data presensi ialah untuk mengelompokkan data karakter dan simbolik secara otomatis oleh mesin (dalam hal ini komputer) untuk mengenali suatu objek di dalam citra. Berbeda dengan Manusia bisa mengenali objek yang dilihatnya karena otak menusia telah belajar mengklasifikasi objek-objek di alam sehingga mampu membedakan suatu objek dengan objek lainnya. Kemampuan menerima masukan berupa citra objek yang akan diidentifikasi, memproses citra tersebut, dan memberikan keluaran berupa deskripsi objek di dalam citra. Kasus yang akan dibangun adalah pengenalan citra untuk menerjemahkan objek citra menjadi huruf dan angka dalam hal ini citra Lembar Presensi Mahasiswa berupa format *grayscale* dengan extension BMP. Metode yang digunakan untuk membaca objek citra ini adalah segmentasi citra dan Freeman Chain code.

1. **Proses Membuat Bubble**

Proses membuat lingkaran merupakan salah satu proses inti yang terdapat pada sistem informasi ini, dimana proses ini digunakan untuk mengenerate data sesuai dengan koordinat yang telah ditentukan di tiap-tiap Objek kolom identitas Lembar Presensi.



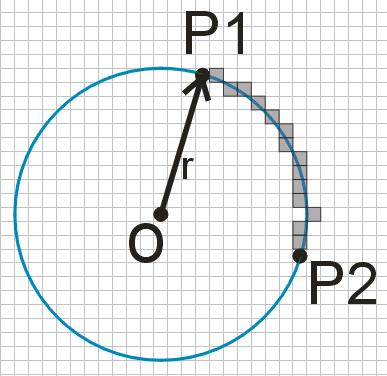
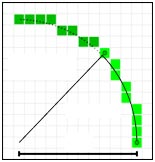
**Gambar 4.16** Membuat lingkaran fungsi trigonometri

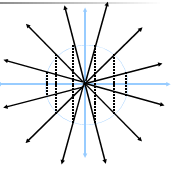
Pada gambar 4.16 terdapat titik pusat dimana titik (a) merupakan titik tengah dari sumbu x dan titik (b) merupakan titik tengah dari sumbu y. Dari titik pusat (a,b) dengan titik yang dipilih (x,y) dan ditarik garis lurus ke x-axis. Terciptalah segitiga dengan panjang sisi miring sebesar 1, Koordinat dari titik yang dipilih merupakan sinus dan kosinus dari sudut θ. Panjang dari kaki-kainya adalah koordinat x dan y dari titik yang dipilih. Jika diterapkan dalam definisi trigonometri maka

Sehingga didapat persamaan

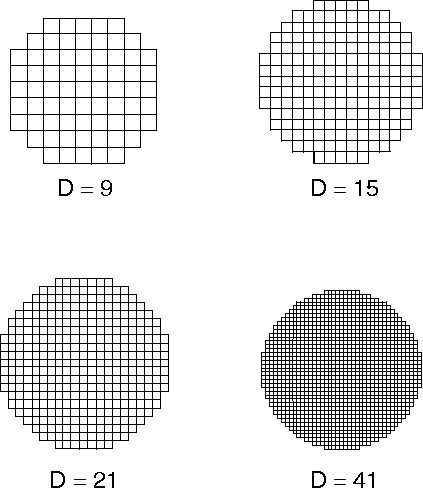
….4.1

Jika kita pindahkan titik yang dipilih keliling lingkaran ke kuadran II, III, dan IV, maka akan dapat disimulasikan seperti di bawah ini :



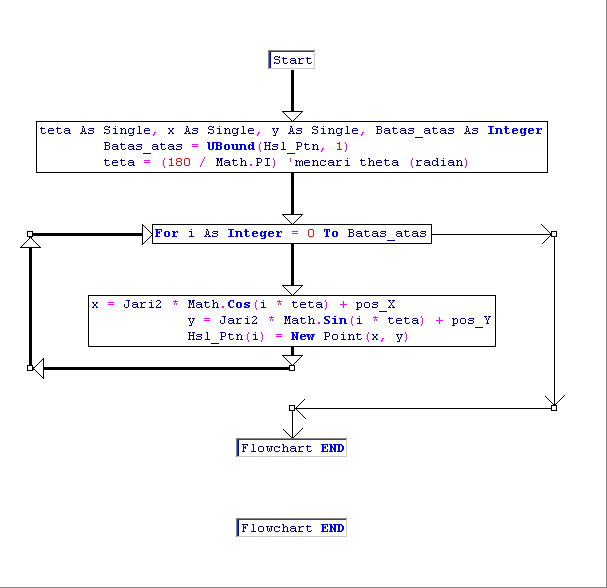


**Gambar 4.17** Simulasi pembuatan lingkaran



**Gambar 4.18** Hasil pembuatan lingkaran

Dari gambar 4.18 dapat dilihat lingkaran yang memiliki nilai diameter 9 tidak sepenuhnya melingkar sedangkan diameter yang bernilai 41 tingkat kerapatan pixelnya cenderung lebih banyak, sehingga dapat ditarik kesimpulan semakin besar nilai diameter lingkaran, semakin sempurna pola lingkaran yang terbentuk.

Berikut Flow Chart proses pembuatan lingkaran

**Gambar 4.19** Flowchart Pembuatan Bubble

1. **Objek Citra LPM**

Contoh objek citra yang akan diuji adalah citra yang berbentuk *form* yang sudah dimodelkan terlebih dahulu dengan menggunakan *tools* CorelDRAW X3. Dari *form* ini hanya ditampilkan data pada area “tanggal” seperti terlihat pada gambar 4.20.

|  |  |
| --- | --- |
| **form tgl.jpg**  **Gambar 4.20**  Citra Template bagian Tanggal | hasil generate tgl.jpg  **Gambar 4.21**  Citra Hasil Generate Tanggal |

Dari *form* tersebut terlihat jarak antar bubble baik secara vertikal maupun horizontal sama-sama memiliki nilai 23 pixel dan letak koordinat awal memiliki nilai x= 238 dan y= 509 yang akan digunakan untuk mengenali area kolom tanggal bulan dan tahun sehingga didapatkan seperti pada gambar 4.21

Berikut ini adalah *flowchart* dari proses Generate data pada Form tanggal



**Gambar 4.22** Flowchart Generate Tanggal

1. **Proses Transformasi Warna**

Proses awal yang banyak dilakukan dalam pengolahan citra adalah transformasi citra berwarna menjadi citra *grayscale*. Tujuan dari transformasi warna ini adalah untuk menyerdehanakan model citra. Citra berwarna terdiri dari 3 layer matrik yaitu Rlayer, Glayer dan Blayer. Sehingga untuk melakukan proses - proses selanjutnya tetap diperhatikan tiga layer di atas.. Bila setiap proses perhitungan dilakukan menggunakan tiga layer, berarti dilakukan tiga perhitungan yang sama. Sehingga konsep itu diubah dengan mengubah 3 layer di atas menjadi 1 layer yaitu matrik grayscale dan hasilnya adalah citra grayscale.

Satu piksel diwakili oleh satu bilangan bulat untuk merepresentasikan informasi pada piksel. Proses pemisahan warna piksel ke dalam warna primer dilakukan dengan cara melakukan operasi AND pada nilai tersebut:

1. Warna merah

Untuk mendapatkan warna merah dari nilai piksel tersebut di-AND-kan dengan nilai 255. Sebagai contoh piksel tersebut bernilai 12563250, maka perhitungannya:

12563250 = 101111111011001100110010

255 = 1111111 AND

50 = 00110010

sehingga didapat nilai warna merah 00110010b atau sama dengan 50.

1. Warna Hijau

Untuk mendapatkan warna hijau dari nilai piksel tersebut sebelum di-AND-kan dengan nilai 255 bit-bit pada piksel tersebut digeser ke kanan sebanyak 8 kali.

12563250 >> 8 = 000000001011111110110011

255 = 1111111 AND

179 = 10110011

sehingga didapat nilai warna merah 10110011b atau sama dengan 179.

1. Warna Biru

Untuk mendapatkan warna hijau dari nilai piksel tersebut sebelum di-AND-kan dengan nilai 255 bit-bit pada piksel tersebut digeser ke kanan sebanyak 8 kali.

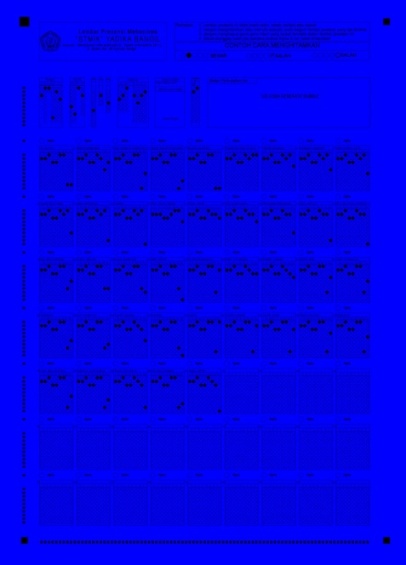
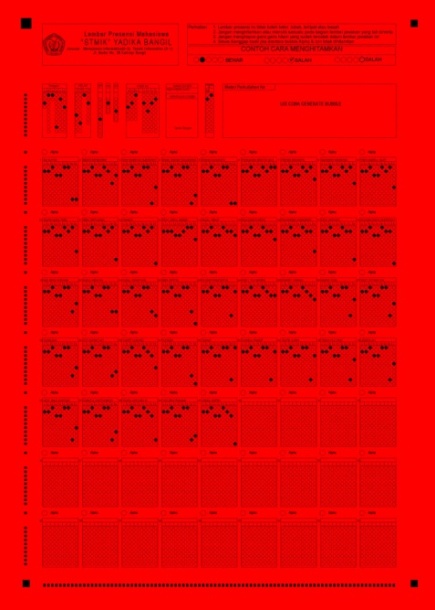
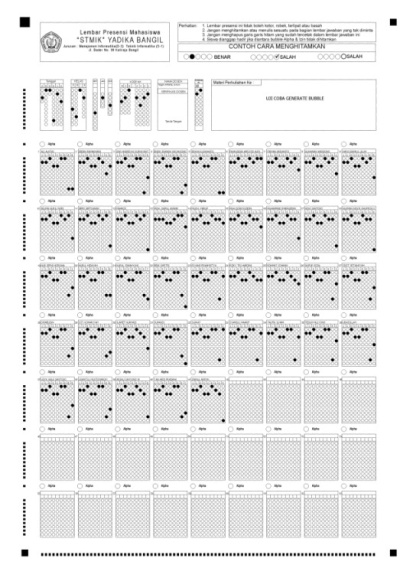
12563250 >> 16 = 000000000000000010111111

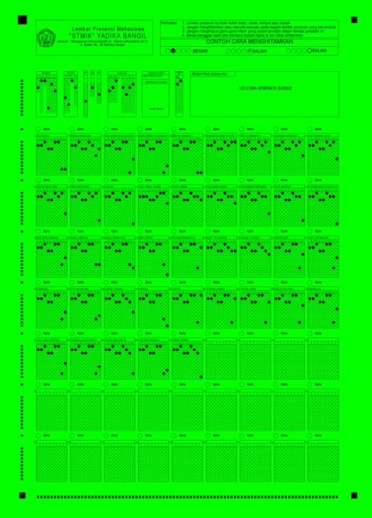
255 = 1111111 AND

191 = 10111111

sehingga didapat nilai warna merah 10111111b atau sama dengan 191.

Untuk mengubah citra berwarna yang mempunyai nilai matrik masing – masing r, g dan b menjadi citra gray scale dengan nilai s, maka konversi dapat dilakukan dengan mengambil rata – rata dari nilai r, g dan b sehingga dapat dituliskan menjadi:

…………………………………(4.2)



**Gambar 4.23.** Hasil pengambilan nilai RGB dan *Grayscale* citra LPM

Berikut ini adalah *flowchart* dari proses transformasi warna.

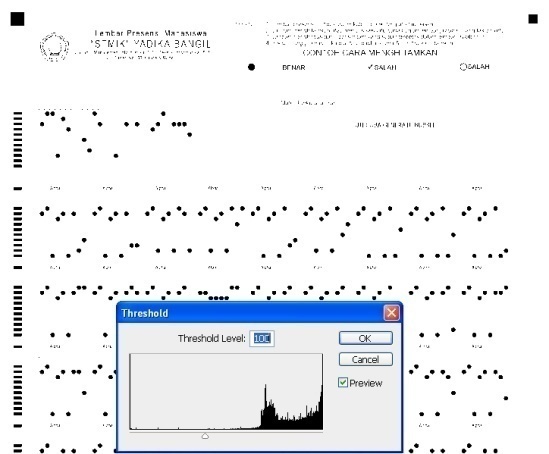
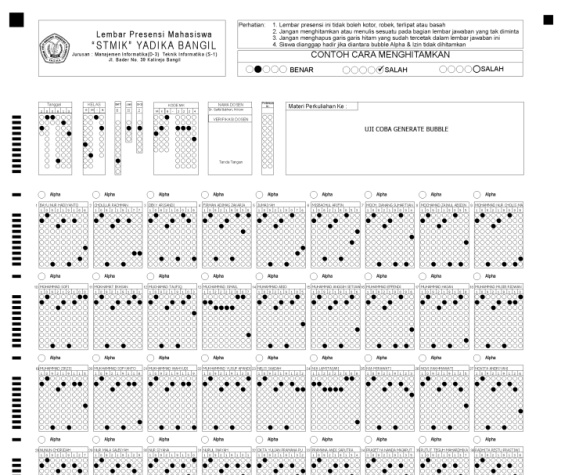


**Gambar 4.24.** *Flowchart* proses transofmasi warna

1. **Segmentasi Berdasarkan Histogram**

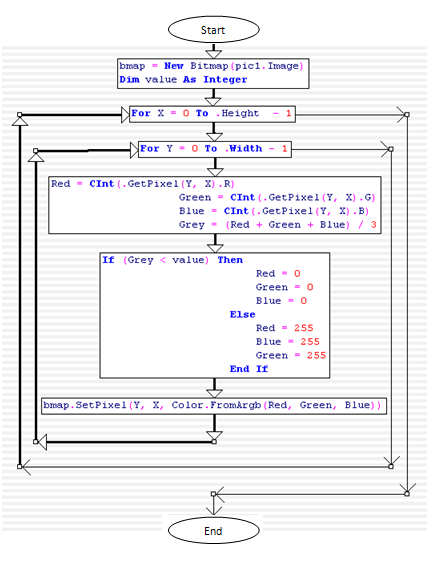
Segmentasi adalah suatu proses yang digunakan untuk mengelompokkan citra sesuai dengan obyek citranya. Setelah menentukan berapa banyak distribusi warna yang digunakan dengan melihat histogram citra *grayscale* maka tiap citra yang ada dilakukan segmentasi warna.

Langkah awal untuk proses segmentasi adalah mengkonversi dari citra *grayscale* ke citra biner dilakukan dengan operasi pengambangan (*thresholding*). Operasi pengambangan mengelompokkan nilai derajat keabuan setiap piksel ke dalam 2 kelas, yaitu hitam dan putih. Sehingga jika ingin memisahkan (segmentasi) obyek dari gambar latar belakangnya. Piksel-piksel obyek dinyatakan dengan nilai 1 sedangkan piksel lainnya dengan 0. Untuk memperoleh hasil yang bagus, obyek harus dapat dipisahkan dari gambar latar belakangnya, berikut ilustrasi dari proses segmentasi dengan nilai threshold 100.



**Gambar 4.25.** Hasil proses *Thresholding*

Berikut ini adalah *flowchart* dari proses segmentasi warna.



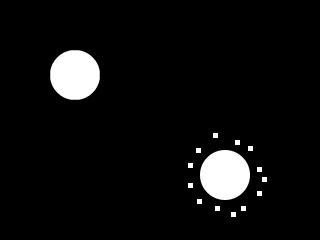
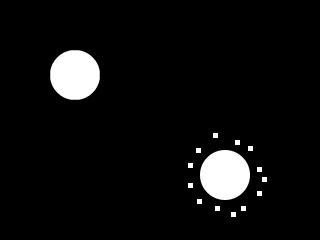
**Gambar 4.26.** *Flowchart* proses segmentasi

1. **Reduksi Noise**

Derau (*Noise*) adalah gambar atau piksel yang mengganggu kualitas citra. Derau dapat disebabkan oleh gangguan fisis (optik) pada alat akuisisi maupun secara disengaja akibat proses pengolahan yang tidak sesuai Dalam tahapan transformasi warna terdapat banyak derauyang menyertai hasil, maka reduksi deraudiperlukan untuk mengurangi sebanyak mungkin derau tersebut.

Operasi morfologi *– opening* berguna dalam menghilangkan obyek – obyek kecil yang terdapat dalam tanpa terjadinya penurunan ukuran obyek yang lainnya.

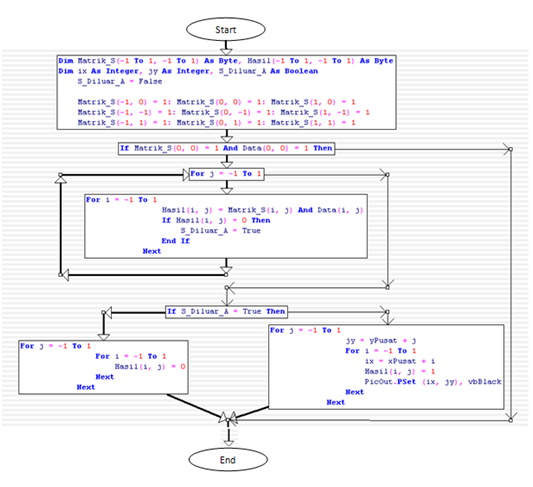
Berikut merupakan gambaran operasi morfologi *– opening.*

**

(b)

(a)

**Gambar 4.27.** (a) Citra awal (b) Operasi Opening

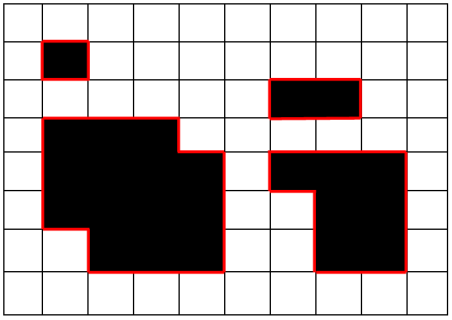
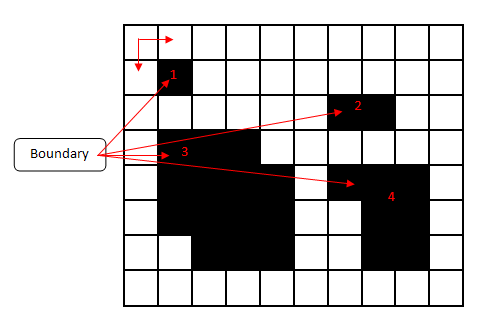
Berikut ini adalah *flowchart* dari operasi morfologi *– opening*.

**Gambar 4.28.** *Flowchart* proses morfologi *– opening*

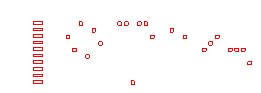
1. **Boundary Detection**

Operasi morfologi merupakan operasi yang digunakan untuk meningkatkan bentuk (struktur) sehingga lebih mudah untuk di kenali (Ahmad,2005).

Dengan pendekatan morfologi, kita memandang suatu citra sebagai himpunan posisi-posisi (x,y) yang bernilai 1 atau 0 . Berikut ilustrasi proses *boundary detection.*

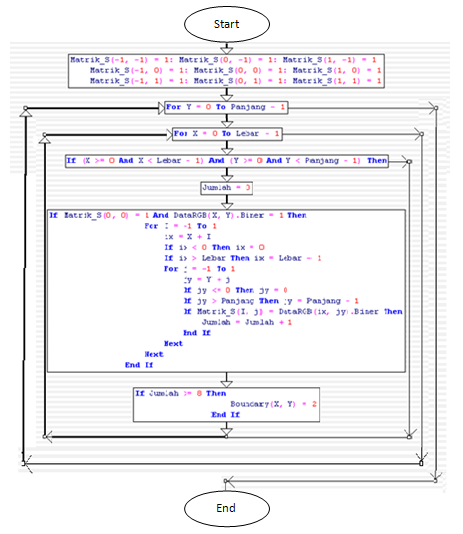
****

**Gambar 4.29.** Ilustrasi proses *boundary detection*

****

**Gambar 4.30.** Hasil proses *boundary detection*

Berikut ini adalah *flowchart* dari proses *boundary detection*.

****

**Gambar 4.31.** *Flowchart* *boundary detection*

1. **Deteksi dan Analisisa *Bubble* Menggunakan *Chain Code***

*Chain code* digunakan untuk menggambarkan batas obyek atau jumlah piksel yang berada dalam satu obyek. Batas obyek direpresentasikan dengan piksel-piksel yang saling terhubung dan memiliki nilai yang sama. *Chain code* mendeskripsikan sebuah obyek dengan segmen garis yang berurutan berdasarkan arah prioritas penelusuran yang telah ditetapkan. Arah dari tiap segmen direpresentasikan dengan angka tertentu. Elemen pertama pada sebuah urutan harus memberikan informasi mengenai posisinya sehingga rekonstruksi area atau perhitungan luas dapat dilakukan. *Chain code* berjalan dengan menelusuri piksel-piksel pada citra berdasarkan prioritas arah yang telah ditentukan, dalam penelitian ini digunakan *chain code* yangterdiri dari 8 arah mata angin.

Dalam mencari *chain code*, langkah pertama adalah menemukan titik awal obyek. Pencarian titik awal dilakukan dengan mengambil nilai titik awal obyek yang telah terdeteksi pada proses *boundary detection* dan penentuan prioritas arah pencarian seperti gambar 4.32.a. Arah pencarian pada *chain code* dimulai dari titik awal dan berakhir pada titik yang sama seperti pada gambar 4.12.b.

1

2

3

4

6

7

8

5

1

2

7

8

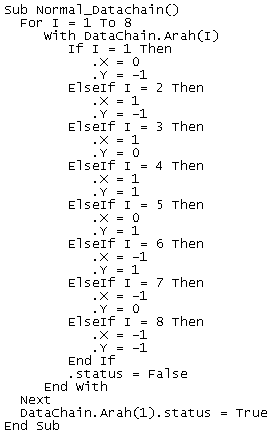
3

(a)

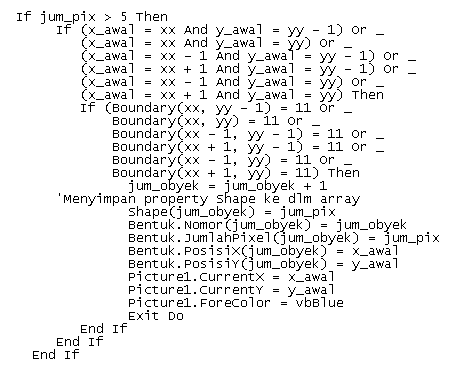
(b)

**Gambar 4.32.** (a) *Chain code* arah 8 mata angin, (b) Arah pencarian titik awal pada akhir proses *chain code*

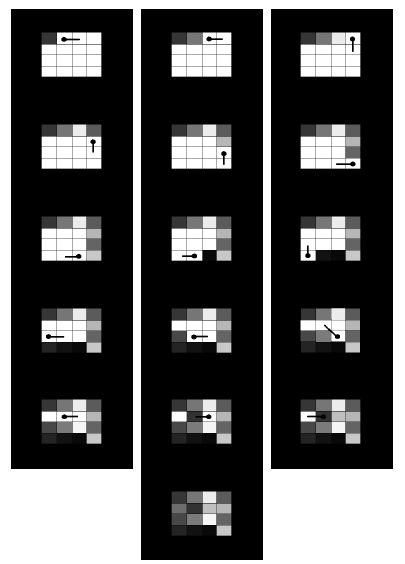
Inisial 8 arah chain code



Arah proses pencarian titik awal pada akhir proses *chain code*



Proses penghitungan luas obyek dilakukan dengan cara menghitung banyaknya piksel pada tiap area obyek, sedangkan *chain code* digunakan sebagai penentu prioritas arah penghitungan luas. Berikut ilustrasi proses penghitungan luas area obyek.



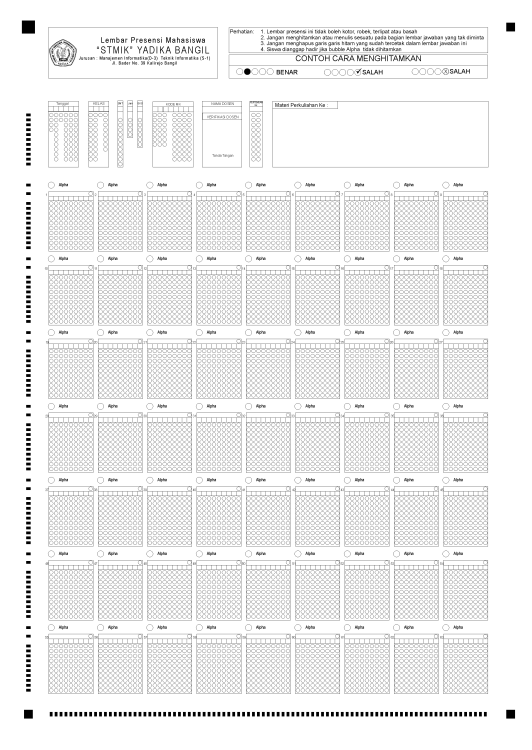
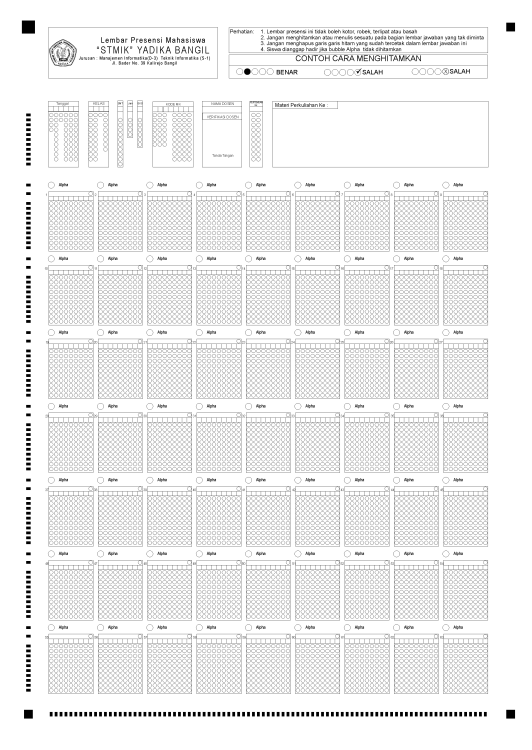
**Gambar 4.33.** Ilustrasi proses hitung luas

1. **Normalisasi Data Citra**

Setelah dilakukan proses penghitungan luas area obyek, maka selanjutnya dilakukan proses pencarian titik acuan dengan cara mengurutkan semua obyek yang telah terdeteksi berdasarkan luas area terbesar atau area dengan jumlah piksel terbanyak. Setelah diketahui obyek dengan luas area terbesar, selanjutnya adalah proses pencarian titik acuan yang kedua yaitu dengan cara mencari obyek yang mempunyai posisi y terdekat dengan obyek dengan luas area terbesar seperti pada gambar berikut.



2





2

**Gambar 4.34.** (1) Obyek dengan luas area terbesar (2) Obyek yang mempunyai posisi y terdekat dengan obyek 1

Dalam proses pencarian 2 buah obyek yang dijadikan acuan diatas, dilakukan pengambilan posisi (x,y) kedua obyek tersebut. Posisi (x,y) kedua obyek tersebut berguna dalam proses pencarian sudut kemiringan data citra sehingga data citra yang miring dapat di-*reposition* (rotasi)menjadi data citra normal. Untuk melakukan proses *reposition* (rotasi)terlebih dahulu dilakukan pencarian derajat kemiringan (gradien) data citra berdasarkan posisi (x,y) kedua obyek tersebut.

Bentuk umum persamaan garis lurus adalah seperti berikut.

y = mx + c …………………………………. (4. 3)

dimana :

m : gradien garis lurus

c : konstanta

sehingga dapat diperoleh rumus ,

m = y / x

dimana,

x ≠ 0

bila garis lurus tersebut melewati kedua obyek tersebut, maka diperoleh rumus,

m = (y2-y1) / (x2-x1) ……………………(4.4)

dimana,

(x2-x1) ≠ 0

sedangkan untuk mendapatkan derajat sudut kemiringan menggunakan rumus,

θ = arctan (m) …………………………….. (4.5)

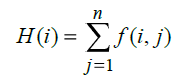
dimana,

θ = derajat sudut kemiringan

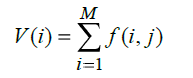
Dalam proses perhitungan derajat sudut kemiringan data citra berlaku ketentuan berikut:

1. Untuk (y2-y1) < 0dan (x2-x1)> 0, maka θ = θ + 360 karena θ bernilai negatif.
2. Untuk (x2-x1)< 0, maka θ = θ + 180 karena berada di kuadran III.
3. Untuk (x2-x1) = 0 dan (y2-y1) > 0, maka θ = 90.
4. Untuk (x2-x1) = 0 dan (y2-y1) < 0, maka θ = 270.

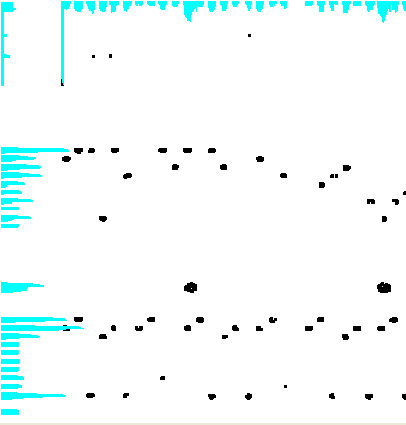
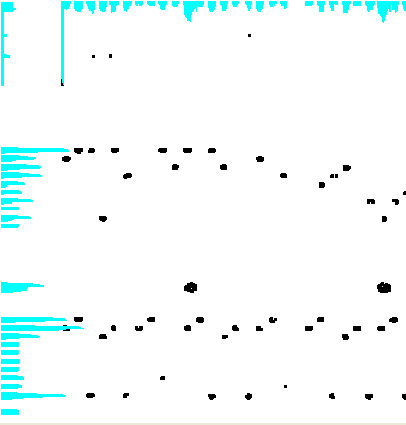
Selanjutnya adalah proses penentuan status terbalik atau tidaknya data citra menggunakan metode *horizontal integral projection* dan v*ertical integral projection*. *Horizontal integral projection* adalah penjumlahan matrik citra dari baris ke *m* sampai pada kolom ke *n* atau dengan kata lain penjumlahan matrik citra dengan arah baris. *Horizontal projection* bertujuan untuk mencari posisi baris maksimum.

 ………………………(4.6)

Sedangkan *vertical integral projection* adalah penjumlahan matrik citra dari kolom ke *n* sampai pada baris ke *m*. Atau dengan kata lain penjumlahan matrik citra dengan arah kolom. *Vertical projection* bertujuan untuk mencari posisi kolom maksimum.

……………………...(4.7)

Setelah didapatkan nilai tertinggi dari *horizontal integral projection* dan *vertical integral projection,* maka dicari baris / posisi Y dari nilai tertinggi pada *horizontal integral projection* dan kolom / posisi X dari nilai tertinggi pada *vertical integral projection* seperti gambar berikut.

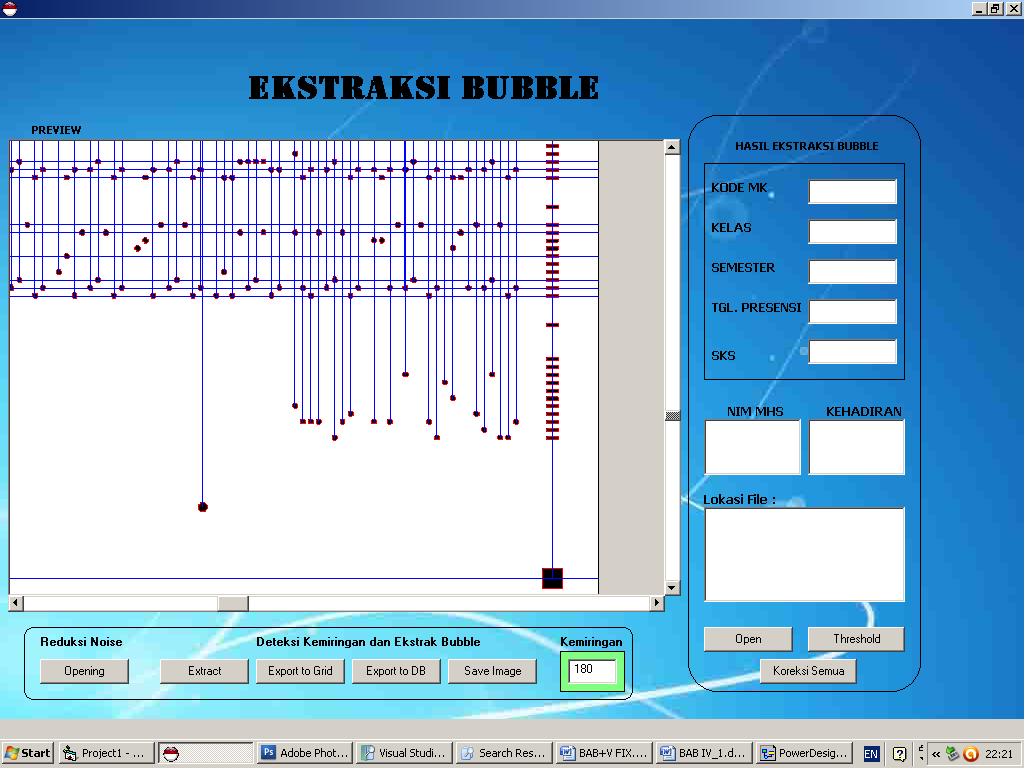


* + - 1. (b)

**Gambar 4.35** Hasil *integral projection* data citra setelah segmentasi dan reduksi *noise*  (a) data citra normal (b) data citra terbalik

Apabila kita amati gambar 4.28.a. nilai tertinggi *horizontal integral projection* pada data citra normal berada di pojok kiri bawah, hal ini berarti posisi Y dari nilai tertinggi *horizontal integral projection* mendekati nilai tinggi citra data. Sedangkan nilai tertinggi *vertical integral projection* pada data citra normal berada di pojok kiri atas, hal ini berarti posisi X dari nilai tertinggi *vertical integral projection* mendekati pangkal nilai X ( X = 0 ) citra data. Kesimpulan dari hal ini adalah jika posisi Y dari nilai tertinggi *horizontal integral projection*  lebih kecil dari setengah nilai tinggi data citra dan posisi Y dari nilai tertinggi *vertical integral projection*  lebih besar dari setengah nilai lebar data citra, maka data citra tersebut terbalik seperti pada gambar 4.35.b.

Berikut adalah contoh hasil deteksi derajat kemiringan data citra terbalik.

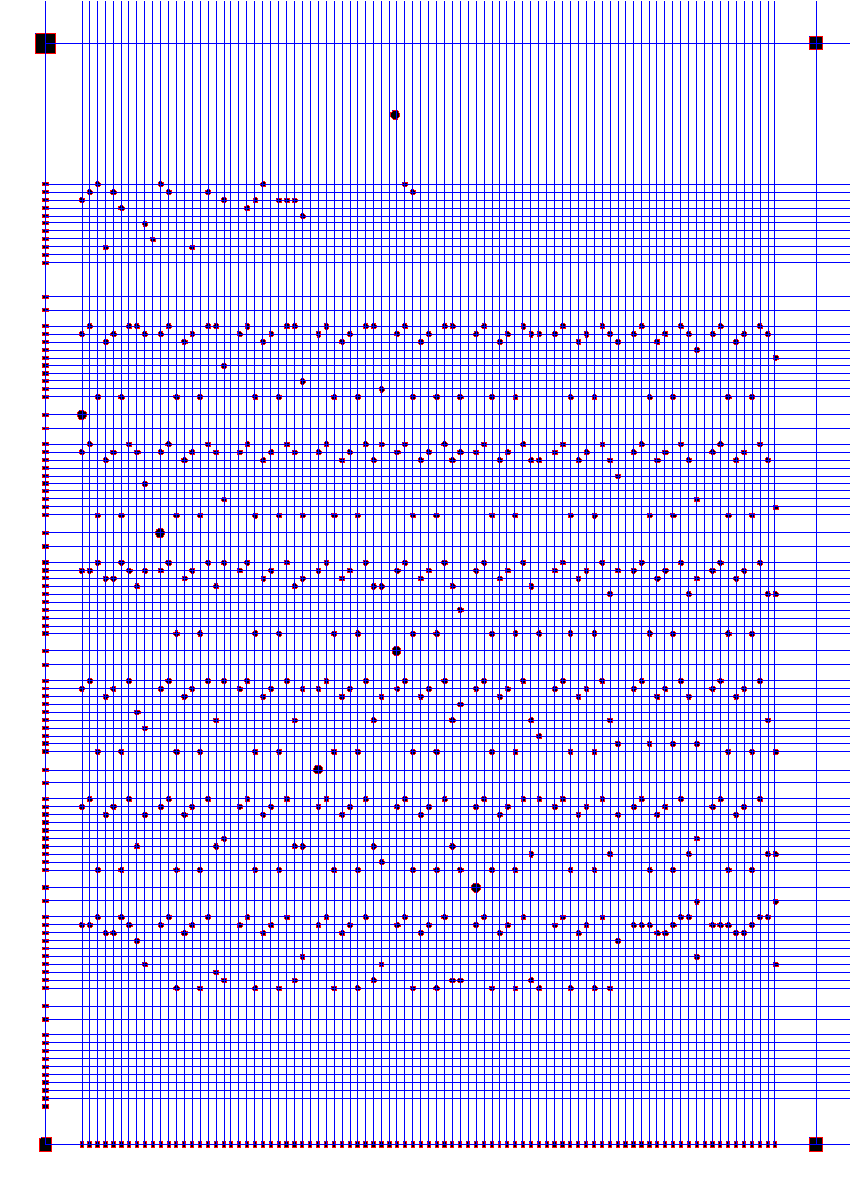


**Gambar 4.36.** Hasil deteksi derajat kemiringan data citra terbalik.

1. **Ekstraksi Identitas Presensi Mahasiswa**

Jika memperhatikan masing-masing area pada lembar presensi mahasiswa, masing-masing daerah yang diarsir akan terdiri lebih dari satu area, tetapi mempunyai lokasi yang saling berdekatan yang dibatasi oleh sebuah *frame* yang disebut kolom sehingga dalam satu lembar presensi mahasiswa terbagi 2 bagian yaitu identitas presensi dan data mahasiswa. Identitas presensi terdiri dari kolom Tanggal, Kelas, Semester, Jam, SKS, Kode MK, Nama Dosen, dan Pertemuan absensi, sedangkan presensi mahasiswa terdiri dari Nim dan Nama Mahasiswa yang berada di posisi bawah dari daerah identitas presensi.

Setelah diketahui bagian – bagian dari lembar presensi mahasiswa, langkah selanjutnya adalah proses ekstraksi *bubble* (bulatan arsiran) meliputi membaca dan menampilkan data pada tabel. Hal ini dilakukan dengan membentuk koordinat posisi berdasarkan posisi *mark* vertikal yang berjumlah 97 buah dan *mark* horizontal yang berjumlah 91 buah, sebagai contoh *markvert(1),markhorz(1)* = “ ” (spasi), *markvert(2),markhorz(1)* = “M”, *markvert(3),markhorz(1)* = “K”, dan seterusnya. Untuk lebih jelasnya perhatikan gambar 4.15.



mark horisontal

M a r k V e r t I k a l Berjumlah 97 Mark

M a r k V e r t I k a l berjumlah 91

**Gambar4.37.** Koordinat obyek berdasarkan *mark* horisontal dan *mark* vertikal.

* 1. **Desain *Interface***

Dalam perancangan pembuatan perangkat lunak untuk pemeriksa lembar presensi mahasiswa digunakan program Microsoft Visual Basic.Net 2008. Aplikasi ini akan memproses data input berupa citra LPM dalam format *bitmap.* Hasil dari program aplikasi ini berupa laporan hasil ekstraksi data presensi mahasiswa dalam file berformat *excel.* Berikut desain input dan output program.

* + - 1. Tampilan form utama program aplikasi.

Pada form ini user bisa memilih folder citra LPM dan secara otomatis *path* citra LPM tampil pada *Scanned image list*, kemudian user melakukan praproses sebelum proses ekstraksi.

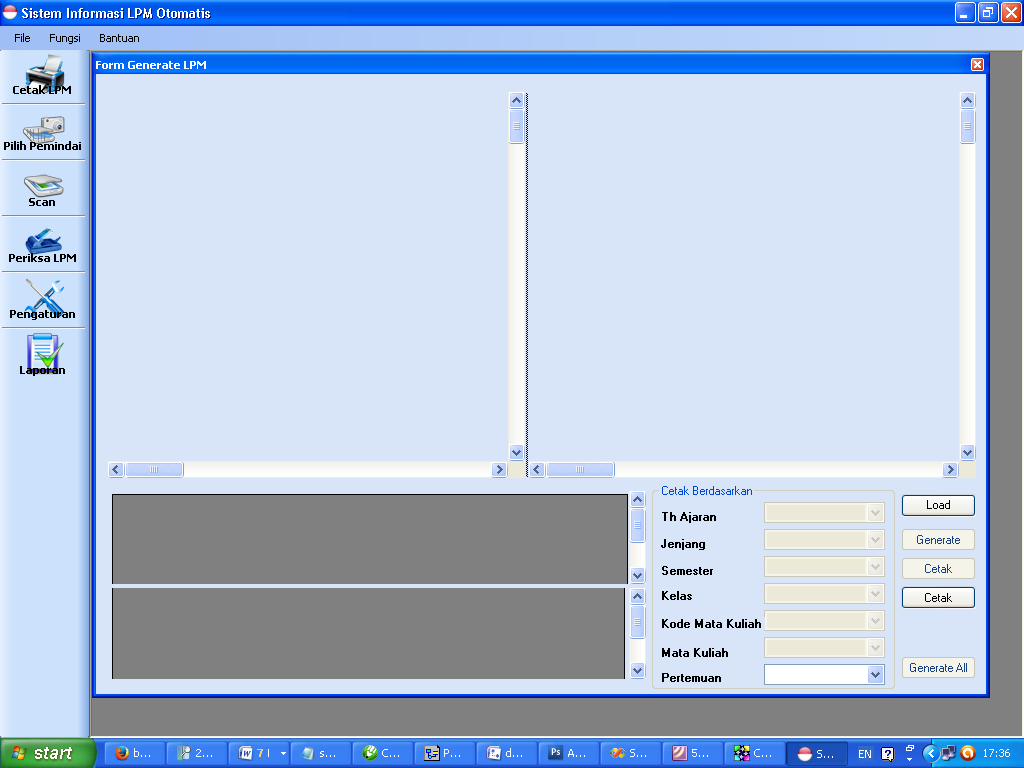


**Gambar 4.38**. Desain form utama

Desain form utama pada perangkat lunak ini terdiri dari beberapa bagian, antara lain:

* 1. *Cetak LPM,* digunakan untuk mengenerate Lembar Presensi mahasiswa
  2. *Pilih Pemindai,* digunakan untuk memilih scanner yang digunakan
  3. *Scan,* digunakan untuk menscan Lembar Presensi Mahasiswa
  4. *Periksa LPM* digunakan untuk mengekstrak data Lembar Presensi Mahasiswa
  5. *Pengaturan*  digunakan untuk mengatur koneksi database, nilai threshold dan ukuran resize citra
  6. *Laporan* digunakan untuk melihat rekap data presensi dari tabel t\_perkuliahan
     + 1. Form Generate LPM

Pada form ini user dapat mengenerate berdasarkan filter database secara otomatis kemudian mencetak citra LPM hasil Generate



**Gambar 4.39**. Desain form Generate LPM

* + - 1. Form Ekstraksi

Pada form ini user bisa memilih folder citra LPM setelah praproses kemudian user melakukan proses reduksi noise sebelum proses ekstraksi dan secara otomatis informasi hasil ekstraks citra lembar presensi tampil pada kolo hasil ekstraksi.

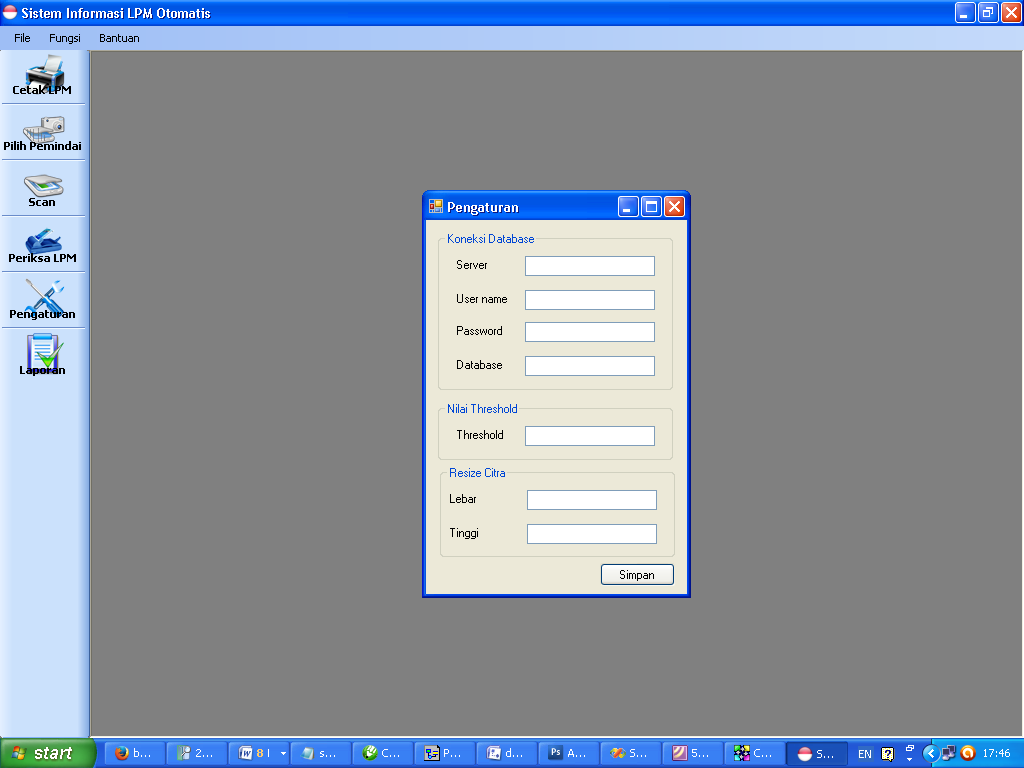


**Gambar 4.40**. Desain Form Ekstraksi

Desain form ekstraksi pada perangkat lunak ini terdiri dari beberapa bagian, antara lain:

* 1. *Previewbox,* digunakan untuk menampilkan data citra presensi yang akan diekstrak.
  2. Kolom Hasil Ekstraksi*,* digunakan untuk menampilkan informasi hasil ekstraksi berupa identitas dan data presensi mahasiswa
  3. Tombol Open digunakan untuk membuka LPM yang akan diekstrak
  4. Tombol Threshold digunakan untuk mengambil citra biner dari LPM
  5. Reduksi *Noise,* digunakan untuk menfilter *noise* data citra LPM sebelum proses ekstraksi.
  6. Ekstrak*,* digunakan untuk mengekstrak identitas data presensi mahasiswa pada data citra lembar LPM.
  7. *Save Image,* digunakan untuk menyimpan citra pada *Previewbox.*
  8. Kolom NIM dan Kehadiran digunakan untuk menampilkan NIM dan Kehadiran mahasiswa yang terdapat pada lembar presensi mahasiswa
  9. Export to DB digunakan untuk menyimpan hasil ekstraksi ke dalam database
  10. Koreksi Semua digunakan untuk mengekstrak data presensi di citra LPM secara masal.
      + 1. Form Pengaturan

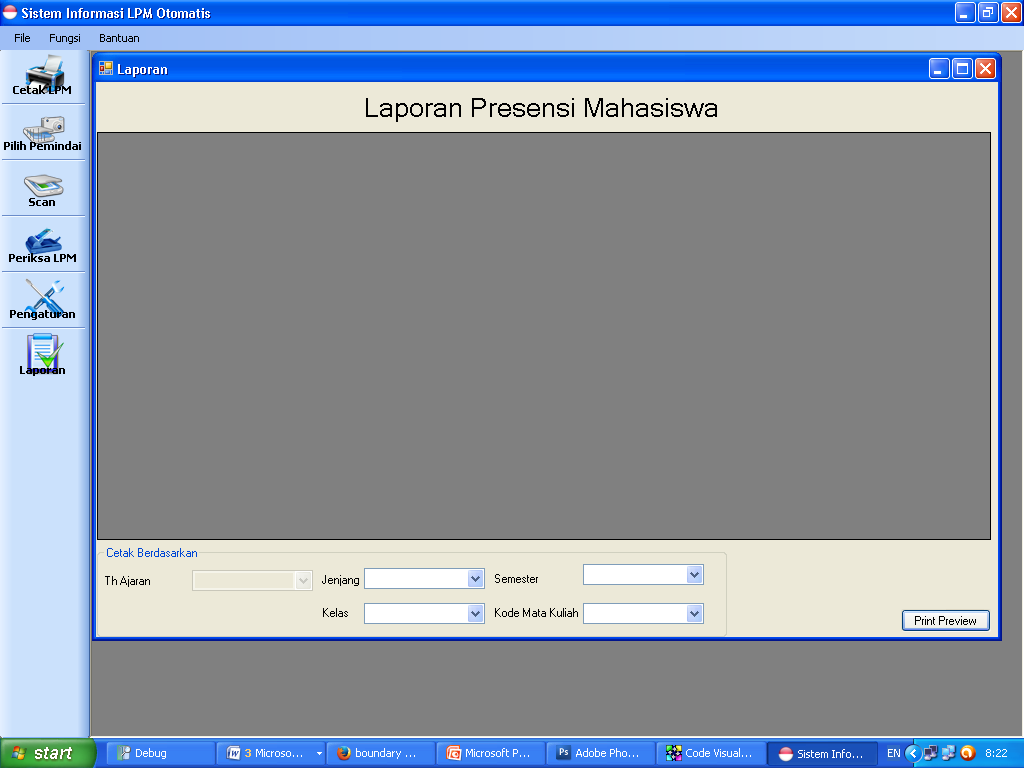
Pada form ini user bisa mengatur koneksi database, nilai threshold dan ukuran resize citra



**Gambar 4.41**. Desain form Pengaturan

* + - 1. Form Laporan

Pada form ini user dapat melihat hasil perekapan data presensi Mahasiswa



**Gambar 4.42**. Desain form Pengaturan

* 1. **Analisa Kebutuhan Sistem**

Alat pendukung dalam pembuat sistem ini terdiri dari perangkat keras dan perangkat lunak, diantaranya :

Spesifikasi perangkat keras yang digunakan:

1. Satu unit laptop dengan spesifikasi sebagai berikut:

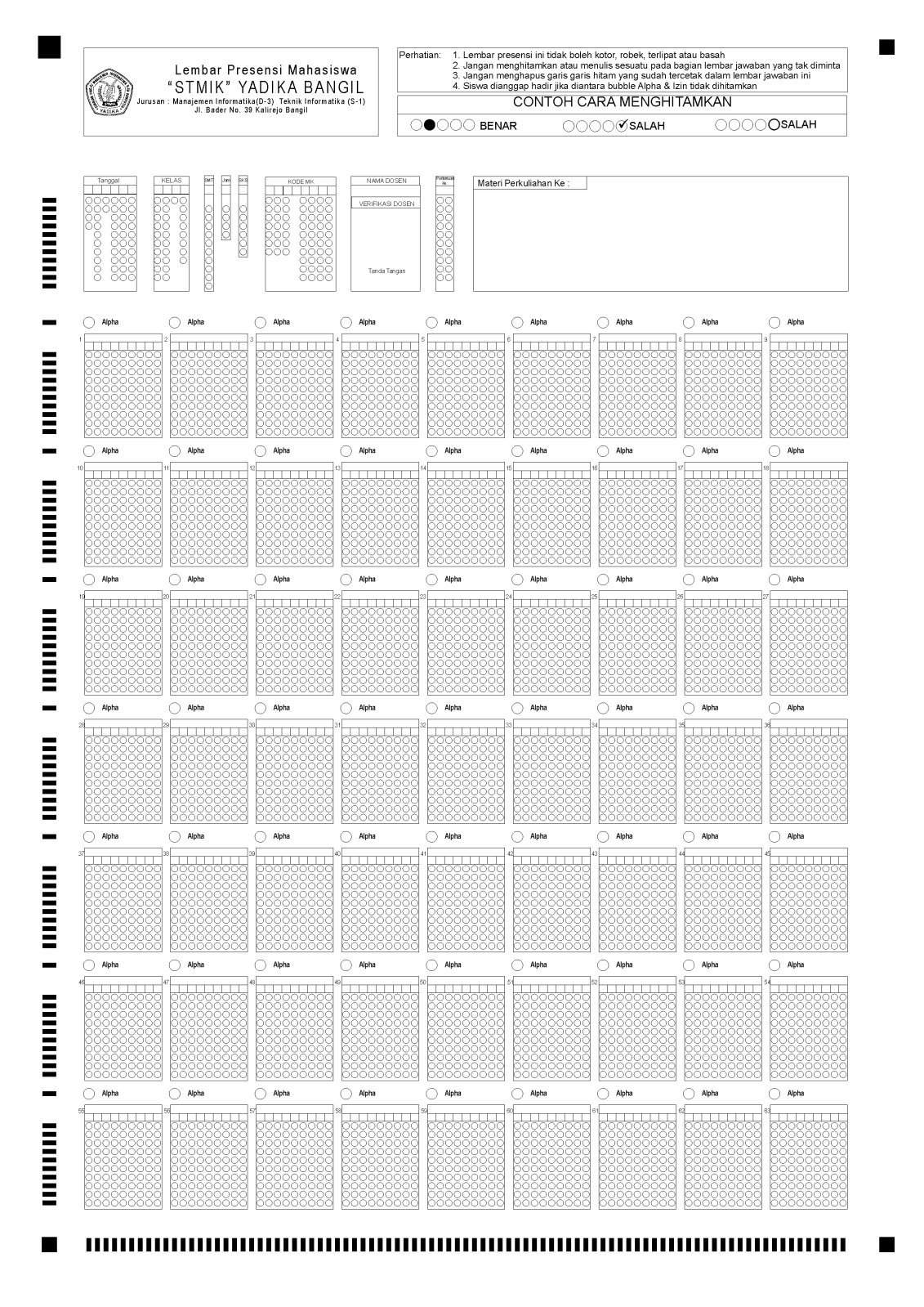
* Prosesor : Intel® Atom ™ CPU 330 1,60GHz
* RAM : 2GB DDR2
* Resolusi Monitor : 12”
* Harddisk : 160GB HDD

1. Scanner *Automatic Document Feeder (ADF)*
2. Kabel USB penghubung PC dengan *scanner.*

Spesifikasi perangkat lunak yang digunakan:

1. *Visual Basic.Net 2008 Professional Edition*
2. *Microsoft Visual Basic 6.0*
3. *CorelDraw X3*
4. *Xampp for Windows*
5. *MySQL Connector Net 6.5.4*
6. *MySQL ODBC Connector*
7. *Driver scanner*
8. Sistem operasi *Windows XP Professional* SP 3

Desain Template Lembar Presensi Mahasiswa:

* Lembar Presensi mahasiswa ukuran A4 (210 x 297 mm).

**Gambar 4.44** Form Lembar Presensi Mahasiswa

**BAB V**

**IMPLEMENTASI DAN EVALUASI SISTEM**

Bab ini membahas tentang pengujian citra yang digunakan dalam mengenerate citra Template Lembar Presensi mahasiswa dan mengekstraksinya untuk dijadikan sebagai entri data presensi mahasiswa. Pengujian dilakukan pada komputer dengan spesifikasi yang telah disebutkan pada bab III.

Pengujian dilakukan dengan menggunakan citra Template LPM dan LPM hasil *scan*. Aplikasi mendeteksi kemiringan citra, kolom identitas dan data mahasiswa, kemudian memberikan informasi kehadiran yang diekstrak, serta memasukkan hasil ekstraksi ke laporan dalam database.

**5.1 Lingkungan Implementasi**

Lingkungan implementasi merupakan kondisi uji coba dilaksanakan. Kondisi tersebut meliputi perangkat lunak yang digunakan dan file yang diuji coba.

**5.1.1 Sistem Operasi**

Perangkat lunak ini diimplementasikan pada lingkungan sistem operasi Windows. Sistem operasi ini digunakan karena pertimbangan dipakai oleh banyak orang serta *user friendly*.

**5.1.2 Bahasa Pemrograman**

Bahasa pemrograman yang dipakai dalam membangun aplikasi ini adalah Visual Basic. Program ini di pilih karena pertimbangan bahwa program tersebut banyak digunakan dalam membangun suatu aplikasi dan telah diajarkan dalam materi perkuliahan tentang fungsinya dalam pembuatan program berbasis pengolahan citra (*image processing*).

**5.2 Lingkungan Pengujian**

Pengujian akan dilakukan dengan citra yang berbeda-beda dan sesuai dengan batasan masalah yang telah dijabarkan sebelumnya.

**5.3 Spesifikasi Lembar Presensi**

Pencetakan lembar Presensi dilakukan dengan spesifikasi yang ada di bab III (lihat point 3.2 Pembentukan Citra Masukan). Template Lembar Presensi degenerate secara otomatis kemudian LPM yang sudah tercetak digunakan untuk proses presensi kemudian di scan untuk mengubah menjadi data citra lembar Presensi digital. Pada tabel di bawah ini ditampilkan sample hasil Generate per kolom pada Lembar Presensi Mahasiswa, dan hasil scan lembar Presensi serta keberhasilan uji coba terhadap sample setelah dipakai sebagai masukan dari aplikasi.

**Tabel 5.1** Hasil uji coba *Generate* lembar Presensi

|  |  |  |  |
| --- | --- | --- | --- |
| **Identitas** | **Citra** | **Jumlah Data** | **Berhasil/**  **tidak** |
| Kolom Tanggal | tgl.jpg | 2 | Berhasil |
| Kolom Kelas | kelas.jpg | 2 | Berhasil |
| Kolom Semester | smt.jpg | 2 | Berhasil |
| Koklom Jam | jam.jpg | 2 | Berhasil |
| Kolom SKS | sks.jpg | 2 | Berhasil |
| Kolom Kode MK | mk.jpg | 2 | Berhasil |
| Pertemuan | pertemuan.jpg | 2 | Berhasil |
| NIM MHS | nim.jpg | 2 | Berhasil |
| NIM MHS Transfer |  | Berhasil |
| Nama MHS |  | Berhasil |
| **Rata-rata Keberhasilan Generate Bubble** | |  | 100% |

.

**Tabel 5.2** Hasil uji coba *scan* lembar Presensi

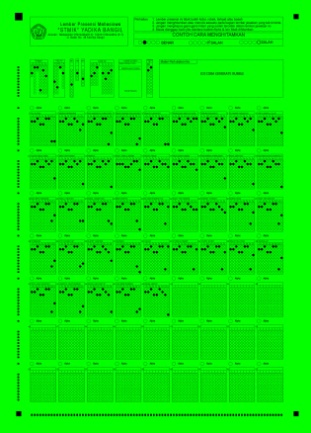
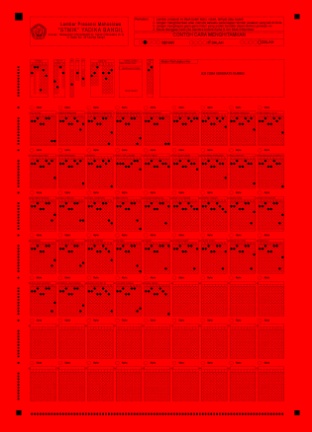
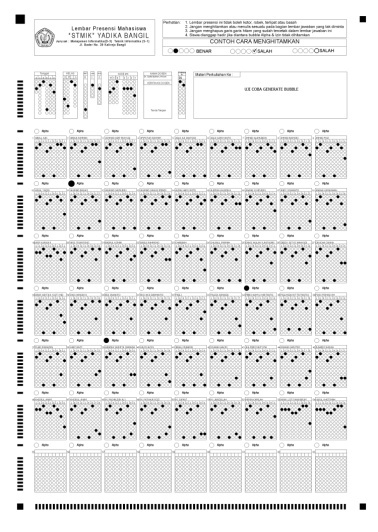
|  |  |  |  |
| --- | --- | --- | --- |
| **DPI** | **Ukuran** | **Resolusi** | **Berhasil/tidak** |
| 72 | 100 KB | 320 x 457 | Tidak berhasil |
| 96 | 375 KB | 530 x 757 | Tidak berhasil |
| 200 | 1200 KB | 700 x 990 | Berhasil |
| 300 | 3242 KB | 1235 x 1686 | Berhasil |
| 400 | 3445 KB | 2409 x 3436 | Berhasil |
| 600 | 3559 KB | 2480x 3508 | Berhasil |

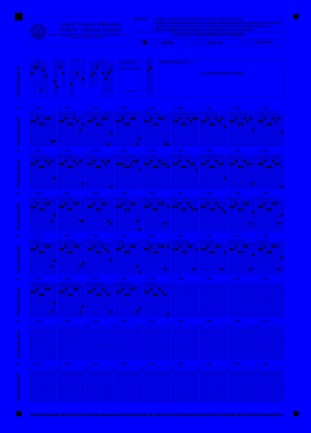
*Scanning* dengan 700 dpi menggunakan *scanner* ADF tidak dapat dilakukan. Dari hasil uji coba, ditentukan bahwa pen-*scan*-an lembar Presensi ditetapkan dengan 300 dpi dengan pertimbangan merupakan *default* *setting* dari *scanner* dan ukuran citra hasil tidak terlalu besar. Resolusi gambar hasil scan adalah dan ukuran file sekitar 3.5 MB. Pada beberapa kali uji coba proses ekstraksi tidak berjalan yaitu saat terdapat objek yang lebih besar dari objek acuan utama baik berasal dari kertas kotor, proses pencetakan atau pada saat proses segmentasi.

**5.4 Analisa Hasil Pengujian**

**5.4.1 Implementasi Transformasi Warna**

Proses awal yang dilakukan dalam pengolahan citra ini adalah transformasi citra berwarna menjadi citra grayscale. Tujuan dari transformasi warna ini adalah untuk menyerdehanakan model citra. Citra berwarna terdiri dari 3 layer matrik yaitu Rlayer, Glayer dan Blayer. Transformasi warna ini dapat dilakukan dengan mengambil rata – rata dari nilai Red, Green dan Blue data citra. Gambar berikut adalah ilustrasi mendapatkan nilai Red, Green dan Blue serta hasil transformasi warna dari citra lembar Presensi.





**Gambar 5.1** (a) hasil pengambilan nilai RGB (b) citra grey

**5.4.2 Segmentasi Citra dengan *Thresholding***

*Thresholding* dilakukan untuk mengetahui apakah warna piksel dari citra lembar Presensi digital termasuk latar belakang (warna putih) atau objek isian (arsiran). Nilai *threshold* yang digunakan sangat menentukan keberhasilan pada proses selanjutnya.

Berikut adalah tabel data yang didapat dari hasil pengujian dengan nilai *threshold* antara 0 – 100. Karena jika diuji satu persatu dari 0 hingga 255, akan menghasilkan data terlalu banyak. Maka nilai *threshold* yang diambil adalah 0, 20, 30, 40, 50, 60, 70, 90, 100

**Tabel 5.3** Hasil uji coba segmentasi citra Grey dari ScanLembar Presensi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nilai *threshold*** | **Alat pengisian** | **Gambar** | | **Warna hitam** |
| **Asli** | **Setelah proses *threshold* (perbesaran 8x)** |
| **0** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** |  | Kurang dari setengah *bubble* |
| **20** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** |  | Kurang dari setengah *bubble* |
| **30** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | **bullpoint20.jpg** | Lebih dari setengah *bubble* |
| **40** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | **bullpoint30.jpg** | Lebih dari setengah *bubble* |
| **50** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | **bullpoint40.jpg** | Lebih dari setengah *bubble* |
| **60** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | **bullpoint50.jpg** | Lebih dari setengah *bubble* |
| **70** | Pensil | **pensil1.jpg** |  | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | **bullpoint60.jpg** | Lebih dari setengah *bubble* |
| **80** | Pensil | **pensil1.jpg** | pensil90.jpg | Kurang dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | bullpoint70.jpg | Lebih dari setengah *bubble* |
| **90** | Pensil | **pensil1.jpg** | pensil100.jpg | Lebih dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | bullpoint80.jpg | Lebih dari setengah *bubble* |
| **100** | Pensil | **pensil1.jpg** | pensil101.jpg | Lebih dari setengah *bubble* |
| Ballpoint hitam | **ling1.bmp** | bullpoint100.jpg | Lebih dari setengah *bubble* |

Dari pengamatan terhadap sample *bubble* (bulatan arsiran) citra uji coba di atas diambil angka ambang untuk penentuan nilai *threshold* yaitu ketika warna hitam hasil proses *thresholding* lebih dari setengah *bubble* (bulatan arsiran) agar objek pada data citra tidak hilang dalam proses selanjutnya yaitu reduksi *noise*. Sedangkan untuk alat tulis yang diperbolehkan untuk mengisi *bubble* adalah pensil, pena warna hitam

Berikut ini adalah tabel data sampel yang didapat dari hasil pengujian *range* nilai *threshold* pada 3 tipe data citra yaitu data citra lembar Presensi diisi dengan pensil dan data citra lembar Presensi hasil scan ADF yang diisi dengan ballpoint hitam.

**Tabel 5.4** Hasil uji coba *range threshold* citra lembar Presensi

|  |  |  |
| --- | --- | --- |
| **alat pengisian** | ***Range Threshold*** | **Keterangan** |
| Pensil | ≤ 70 | Gagal |
| 80 - 100 | Berhasil |
| Bullpoint | ≤ 20 | Gagal |
| 30-100 | Berhasil |

**5.4.3 Reduksi *Noise* Citra dengan Operasi Morfologi *Opening***

Pada tahap reduksi menggunakan operasi morfologi opening berguna untuk menghilangkan obyek – obyek kecil yang terdapat dalam citra tanpa terjadinya penurunan ukuran obyek yang lainnya. Berikut ini adalah tabel data sampel yang didapat dari hasil pengujian operasi morfologi *opening* menggunakan *structuring elements* ukuran 3x3 dan sampel diambil dari kolom nomor peserta data citra yang telah tersegmentasi dengan nilai *threshold* 90.

**Tabel 5.5** Hasil uji coba operasi morfologi *opening* citra lembar Presensi

| **Warna cetakan, alat pengisian** | **Gambar hasil segmentasi (nilai *threshold* = 90)** | **Gambar hasil *opening*** |
| --- | --- | --- |
| Pensil | D:\Uji LPM 2\Segmentasi\lpm-Rabu.bmp | asli |
| Ballpoint hitam | D:\Uji LPM 2\Segmentasi\lpm-Rabu.bmp | *asli* |

**5.4.4 *Boundary Detection* Objek pada Data Citra**

Implementasi pada tahap ini bertujuan untuk memisahkan dan menandai piksel – piksel yang terhubung menjadi satu kesatuan objek pada citra. Proses ini dilakukan dengan memeriksa citra, piksel per piksel (dari kiri ke kanan dan atas ke bawah) untuk mengidentifikasi area piksel terhubung. Gambar berikut merupakan hasil proses *boundary detection*

**Gambar 5.2** Hasil proses *boundary detection*

**5.4.5 Normalisasi Data Citra**

Proses yang dilakukan sebelum proses ekstraksi informasi adalah normalisasi data citra. Proses ini terdiri dari proses pengecekan kemiringan dan proses *reposition* (rotasi)menjadi data citra normal bila data citra tersebut terdeteksi miring. Berikut ini adalah tabel data sampel yang didapat dari hasil pengujian normalisasi data citra lembar Presensi dan sampel diambil dari data citra lembar Presensi cetakan hitam dan diisi dengan ballpoint berwarna hitamyang di-*scan* secara terbalik dengan *scanner* ADF dan data citra yang di-*scan* dengan cara dimiringkan maksimal (± 20) atau terbalik dengan *scanner flatbed.*

**Tabel 5.4** Hasil uji coba normalisasi citra lembar Presensi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sample** | **Jumlah data** | **alat pengisian** | **Deteksi kemiringan** | **Hasil ekstraksi** | **Keterangan** |
| 1 | 2 | Ballpoint & Pensil hitam | 3550 | Salah | Gagal |
| 2 | 2 | Ballpoint & Pensil hitam | 50 | Salah | Gagal |
| 3 | 2 | Ballpoint & Pensil hitam | 00 | Benar | Berhasil |
| 4 | 2 | Ballpoint & Pensil hitam | 1800 | Benar | Berhasil |
| **Rata-rata Keberhasilan** | | |  |  | 50% |

**5.4.4 Kecepatan Proses dan Rekapitulasi Hasil Pemeriksaan LPM**

Pengukuran waktu proses dilakukan dengan menghitung waktu dari *scanning* lembar Presensi sampai pemasukan data ke database. Hal ini dibutuhkan untuk mengetahui lama proses rekapitulasi hasil presensi mahasiswa sehingga dapat dianalisa apakah aplikasi ini dapat diimplementasikan pada sistem informasi lembar presensi mahasiswa otomatis di STMIK Yadika Bangil.

Pengujian dilakukan dengan mengukur proses pada lembar Presensi dengan beberapa kasus. Rata-rata waktu proses didapat dari seluruh data waktu proses yang telah diujicobakan. Rumus untuk mendapatkan rata-rata waktu proses yaitu:

 ..............................*(5.1)*

n = jumlah uji coba yang dilakukan

WP = waktu proses

Sampel lembar Presensi yang akan diukur adalah:

b. Sampel lembar Presensi 2 : jumlah 2 lembar, di-*scan* dengan *scanner* ADF dengan posisi terbalik (diputar 3550), warna cetakan hitam, diisi dengan ballpoint hitam, dan jumlah mahasiswa 39.

b. Sampel lembar Presensi 2 : jumlah 2 lembar, di-*scan* dengan *scanner* ADF dengan posisi terbalik (diputar 50), warna cetakan hitam, diisi dengan ballpoint hitam, dan jumlah mahasiswa 39.

c. Sampel lembar Presensi 3 : jumlah 2 lembar, di-*scan* dengan *scanner flatbed,* warna cetakan hitam, diisi dengan ballpoint hitam, dan jumlah mahasiswa 61 mahasiswa.

d. Sampel lembar Presensi 4 : jumlah 2 lembar, di-*scan* dengan *scanner flatbed* dengan posisi normal dan terbalik (diputar 1800)*,* warna cetakan hitam, diisi dengan pensil dan ballpoint hitam dan jumlah data 61 mahasiswa.

Pengukuran waktu proses dilakukan dengan *stop watch* dimulai dari Generate Bubble, *scan* lembar Presensi, proses pengolahan citra, proses ekstraksi informasi sampai pelaporan ke database. Untuk waktu proses pada aplikasi sampai pemasukan data ke database dilakukan tiga kali berturut-turut untuk sampel yang sama.

**Tabel 5.5** Hasil uji coba waktu proses Generate hingga pemeriksaan lembar Presensi

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sampel** | **Jumlah data** | **Total waktupengujian (dalam detik)** | | | | | | **Total waktu** |
| **Generate Bubble** | **Scan** | **Deteksi Bubble** | | | |
| **Iterasi**  **ke- 1** | **Iterasi**  **Ke-2** | **Iterasi**  **ke-3** | **Rata2 Iterasi** |
| 1 | 2 | 6 | 28 | 256 | 257 | 255 | 256 | 290 |
| 2 | 2 | 6 | 28 | 411 | 410 | 412 | 411 | 445 |
| 3 | 2 | 6 | 33 | 321 | 1143 | 1110 | 858 | 897 |
| 4 | 2 | 6 | 33 | 305 | 412 | 452 | 390 | 429 |

Berdasarkan tabel diatas dapat dianalisa yaitu sampel 1 dan 2 dengan jumlah lembar Presensi yang sama yaitu 2 lembar ternyata selisih waktu yang dibutuhkan cukup banyak yaitu 155 detik, hal ini disebabkan oleh proses normalisasi data citra yang terdeteksi miring. Sedangkan perhitungan rata-rata waktu yang dibutuhkan untuk memeriksa satu lembar adalah sebagai berikut.

2061 / 8

= 258 detik

= 4 menit 3 detik

Berikut ini adalah tabel hasil rekapitulasi pemeriksaan lembar Presensi.

**Tabel 5.6** Hasil rekapitulasi pemeriksaan lembar Presensi.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sampel** | **Jumlah data** | **Prosentase Ekstraksi** | | **Keterangan** |
| **Identitas** | **Presensi** |
| 1 | 2 | 100% | \*\*50% | 50% |
| 2 | 2 | 100% | 100% | 100% |
| 3 | 2 | 100% | 100% | 100% |
| 4 | 2 | 100% | \*50% | 50% |
| **Prosentasi keakuratan** | | | | 75% |

Keterangan : \* kesalahan proses pengisian lembar Presensi

\*\* kesalahan proses ekstraksi lembar Presensi

Berdasarkan pengamatan tabel hasil rekapitulasi pemeriksaan lembar Presensi diatas, didapatkan prosentase keberhasilan ekstraksi lembar Presensi sebesar 75%. Beberapa kesalahan pada saat pengujian sampel yaitu pada proses ekstraksi identitas. Kesalahan ini terjadi 2 kemungkinan yaitu kesalahan pada saat pengisian lembar Presensi dan kesalahan pada saat proses pemeriksaan lembar Presensi.

Berdasarkan pengamatan pada saat proses pemeriksaan lembar Presensi, kesalahan tersebut terjadi karena proses deteksi kemiringan yang salah sehingga proses normalisasi data citra tidak akurat

* 1. **Tampilan Sistem**

Berikut ini akan dijelaskan tentang form-form yang ada pada sistem.

**5.5.1 Form Menu Utama**

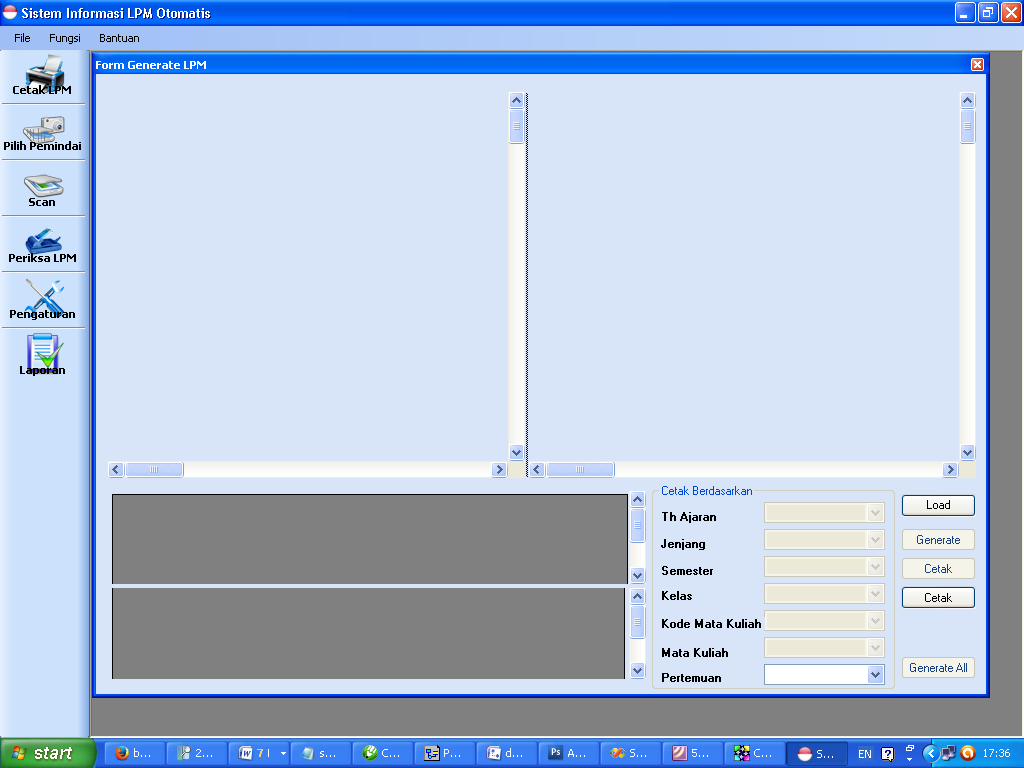
Gambar di bawah merupakan tampilan menu utama pada sistem pemeriksa lembar Presensi. User bisa memilih folder citra lembar Presensi dan secara otomatis *path* citra lembar Presensi tampil pada *Scanned image list*, kemudian user melakukan praproses sebelum proses ekstraksi dengan cara menekan tombol Threshold All dan untuk ke form ekstraksi dengan cara menekan tombol Extract Bubble.

**

**Gambar 5.3** Form menu utama

**5.5.2. Form Generate LPM**

Pada form ini user dapat mengenerate berdasarkan filter database dinataranya Tahun Ajaran, Jenjang, Semester, Kelas, Kode MK kemudian secara otomatis lembar presensi tergenerate pada Template Lembar Presensi mahasiswa, dan tercetak dengan sendirinya



**Gambar 5.4**. Desain form Generate LPM

**5.5.3 Form Ekstraksi**

Pada form ini user bisa memilih folder citra LPM setelah praproses kemudian user melakukan proses reduksi noise sebelum proses ekstraksi dan secara otomatis informasi hasil ekstraks citra lembar presensi tampil pada kolo hasil ekstraksi.



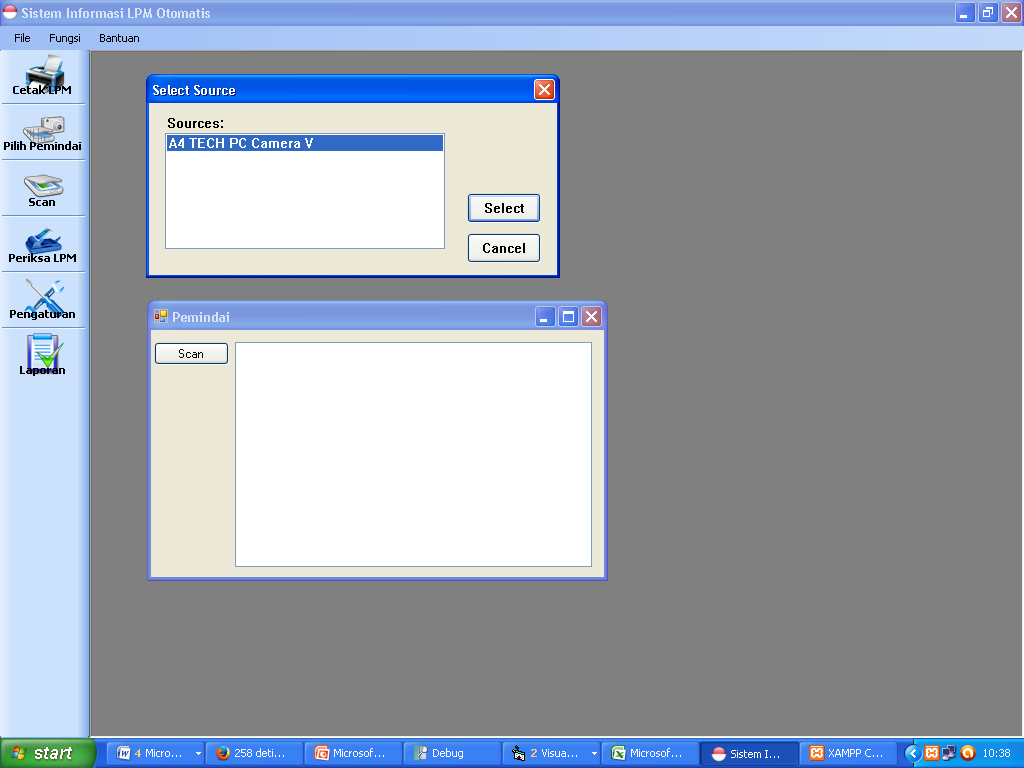
**Gambar 5.5**. Desain Form Ekstraksi

Desain form ekstraksi pada perangkat lunak ini terdiri dari beberapa bagian, antara lain:

Desain form ekstraksi pada perangkat lunak ini terdiri dari beberapa bagian, antara lain:

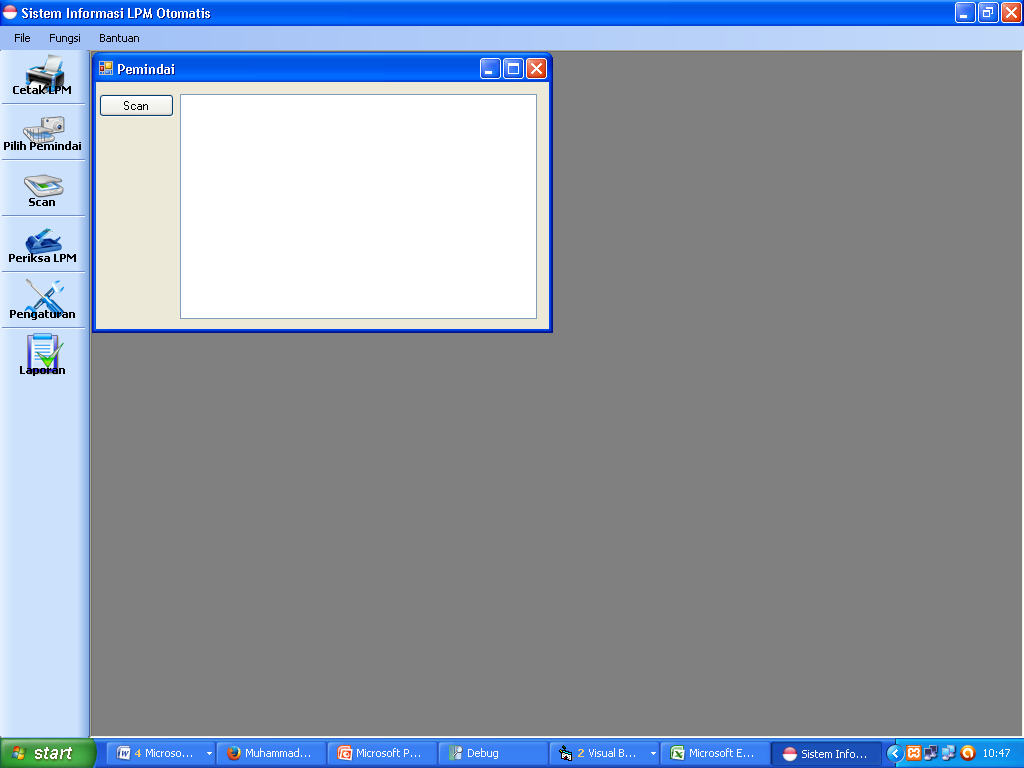
* 1. *Previewbox,* digunakan untuk menampilkan data citra presensi yang akan diekstrak.
  2. Kolom Hasil Ekstraksi*,* digunakan untuk menampilkan informasi hasil ekstraksi berupa identitas dan data presensi mahasiswa
  3. Tombol Open digunakan untuk membuka LPM yang akan diekstrak
  4. Tombol Threshold digunakan untuk mengambil citra biner dari LPM
  5. Reduksi *Noise,* digunakan untuk menfilter *noise* data citra LPM sebelum proses ekstraksi.
  6. Ekstrak*,* digunakan untuk mengekstrak identitas data presensi mahasiswa pada data citra LPM.
  7. *Save Image,* digunakan untuk menyimpan citra pada *Previewbox.*
  8. Kolom NIM dan Kehadiran digunakan untuk menampilkan NIM dan Kehadiran mahasiswa yang terdapat pada lembar presensi mahasiswa
  9. Export to DB digunakan untuk menyimpan hasil ekstraksi ke dalam database
  10. Koreksi Semua digunakan untuk mengekstrak data presensi di citra LPM secara masal.

**5.5.4 Form Pilih Pemindai**

****

**Gambar 5.6**. Desain Form Select Source Scanner

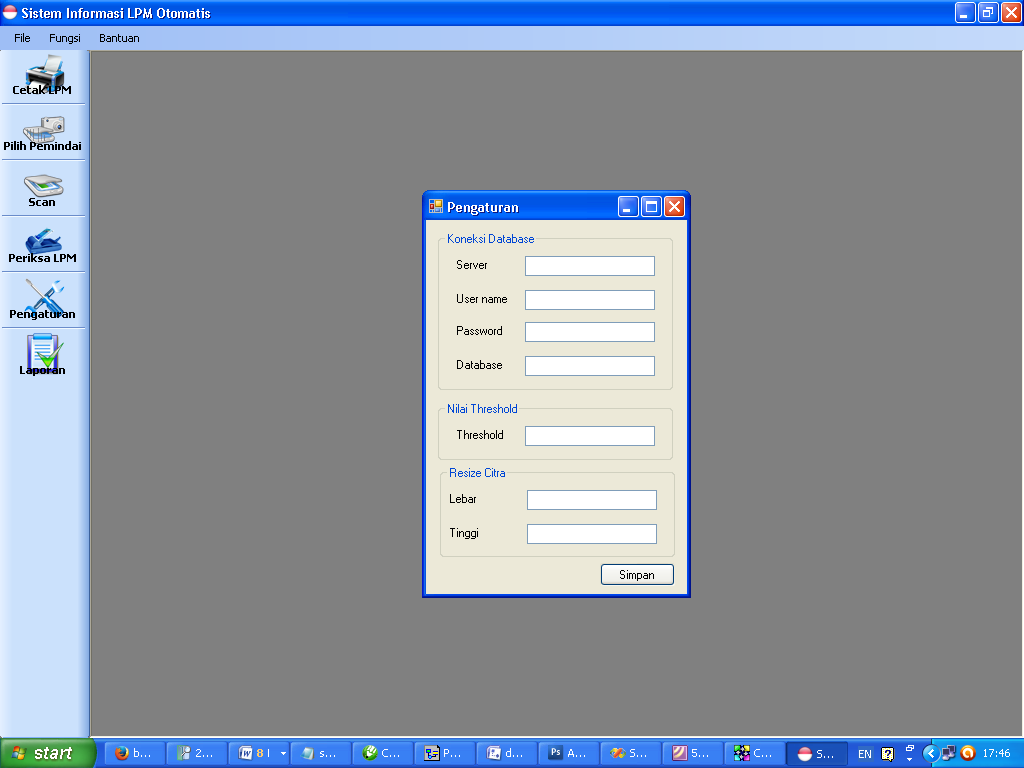
**5.5.5 Form Form Scan**

****

**Gambar 5.6**. Desain Form Scan

**5.5.6 Form Pengaturan**

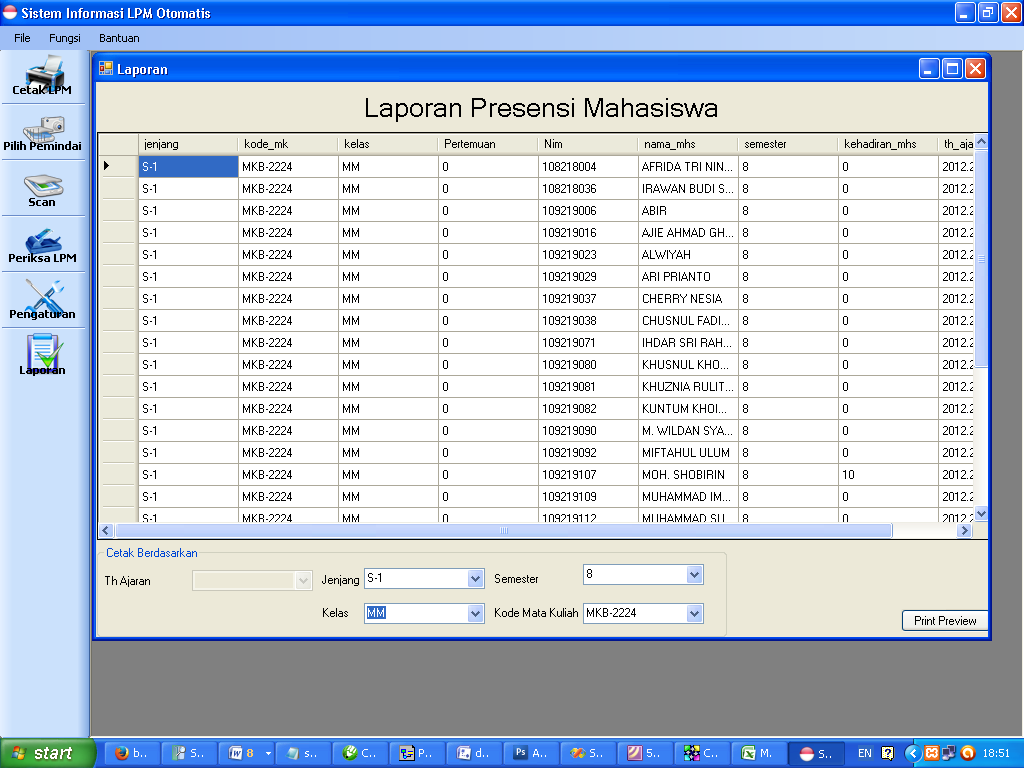
Pada form ini user bisa mengatur koneksi database, nilai threshold dan ukuran resize citra



**Gambar 5.7**. Desain form Pengaturan

**5.5.6 Laporan Hasil Pemeriksaan Lembar Presensi**

Gambaran di bawah merupakan tampilan laporan hasil pemeriksaan lembar Presensi. Struktur laporan hasil pemeriksaan lembar Presensi ini dapat dilihat pada Gambar 5.6



**Gambar 5.8** Laporan pemeriksaan lembar Presensi

**DAFTAR PUSTAKA**

**BUKU**

Amiruddin, KH. Hammam. 1963. *Ta’limul Muta’allim*. Magelang

Fadilsyah, ‘Computer Vision dan Pengolahan Citra’, Yogyakarta, Penerbit Andi. 2007.

Fadlisyah. *Computer Vision dan Pengolahan Citra*. Yogyakarta : CV. Andi Offset

Kusrini, M.Kom, Koniyo, Andri . 2007. *"Tuntunan Praktis Membuat Sistem Informasi Akuntansi dengan Visual Basic & Microsoft SQL Server"*, Yogyakarta : C.V. ANDI OFFSET.

Sutoyo, T, dkk. 2009.*Teori Pengolahan Citra Digital*. Yogyakarta : CV. Andi Offset

Jogiyanto, HM. 2001. *Analisis dan Desin Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Andi.

Jogiyanto, HM. 1999. *Pengenalan Komputer*. Yogyakarta: Andi.

Putra, Darma. 2009. *Pengolahan Citra Digital*. Yogyakarta: Andi.

Rinaldi, Munir. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritmik.* Bandung : Informatika.

McLeod, Jr, Raymond. George Shell. 2004. *Sistem Informasi Manajemen Edisi Kedelapan*. . Jakarta: PT. Indeks.

\_\_\_\_\_. 2010. *Membuat Aplikasi Client Server dengan Visual Basic 2008.* Yogyakarta : Penerbit Andi

Hirin, AM. 2011. *Belajar Tuntas VB.NET 2010*. Jakarta: Prestasi Pustakaraya.

Purnomo, Catur Hadi. 2008. *Panduan Belajar Otodidak Microsoft Excel 2007.* Jakarta : Mediakita

Kusrianto, Adi. 2004. *Panduan Lengkap Memakai CorelDraw 12.* Jakarta : Elexmedia Komputindo

**JURNAL**

Rijal, Yusron. 2010. Optimalisasi reduksi noise menggunakan chain-code termodifikasi pada pendeteksian wajah*. Jurnal CAUCHY* : 1-4.

Silviana, Renita. *Pengembangan Sistem Informasi Transaksional Akademik STMIK Yadika.* Unpublished Paper. Teknik Informatika STMIK YADIKA, Bangil. 2012

Masfran, dkk. 2012. Segmentasi tepi citra CT scan paru-paru menggunakan metode chain code dan operasi morfologi. *Jurnal Teknik Informatika Vol.1* *Politeknik Caltex Riau* : 1-6.

Wiliana, dkk. 2012. Perbandingan algoritma arithmetic dengan geometric mean filter untuk reduksi noise pada citra. *Jurnal Ilmu Komputer dan Teknologi Informasi USU* : 1-4.

Gunadi, Kartika. 1999. Algoritma midpoint untuk penggambaran grafik berkecepatan tinggi. *Jurnal Informatika Vol.1 No.1* : 34-42.

Basuki, Achmad 2006. *Mengenali Angka menggunakan Fitur Bntuk Integral Proyeksi.* Surabaya : PENS – ITS

Basuki, Achmad, Nana Ramadiajnti, Tri Harsono, *Deteksi Rambu-Rambu Batas Kecepatan menggunakan filter RGB dan Integral Proyeksi*. Surabaya : IES, EEPIS. 2007

Cahyana, Nur Heri, Hafsah, Andhika Noorindra, 2009. *Sistem Humidifier dan Temperatureizer Digunakan dalam Penyiraman Otomatis Tanaman*, Yogyakarta : FTI UPN Veteran Yogyakarta

Hanif Al Fatta, *Sistem Presensi Karyawan Berbasis Pengenalan Wajah dengan Algoritma Eigenface,* STMIK AMIKOM Yogyakarta, Seminar Nasional Sistem dan Informatika 2006; Bali

http://edipurwanto1988.blogspot.com/2009/06/keamanan-sistem-informasi-absensi.html

Novia, Windy, 2010. *Kamus Ilmiah Populer, Michigan* : the University of Michigan

Moody, J. 2004. *Public Perception of Biometric Devices: the Effect of Misinformation on Acceptance and Use. Journal of Issue in Informing Science and Information Technology*,753-761.

Riwinoto,S.T,M.Kom, 2012. *Penggunaan Algoritma Hough Tranforms Untuk Deteksi Bentuk Lingkaran pada Ruang 2D.* Batam: Teknik Informatika Politeknik Batam.

Sumarno, Eko. Hanugrah Probo Hasmoro 2012, *Implementasi dan Analisis Aplikasi Pencatatan Kehadiran dengan Sidik Jari Guru dan Karyawan Smp Negeri 2 Karanganyar*, Program Studi Teknik Informatika Universitas Surakarta, Seruni FTI UNSA 2012 Volume 1

Lewis, William E. 2005. Software Testing and Continuous Quality Improvement. Second Edition. Florida : Auerbach Publications.

Ibrahim, Azzam. 2006. A Chain Code Approach for Recognizing Basic Shapes . Rahmeh Omar Jabay King Abdullah II for Information Technology College University Of Jorda

**SURAT PERNYATAAN KETUA PENELITI**

Yang bertanda tangan dibawah ini:

Nama : **Yusron Rijal, S.Si, MT**

NIDN : 0716027302

Pangkat.Golongan : -

Jabatan Fungsional : -

Dengan ini menyatakan bahwa proposal penelitian saya dengan **PENGEMBANGAN SISTEM INFORMASI DETEKSI PRESENSI OTOMATIS MENGGUNAKAN BUBBLE DETECTOR** yang diusulkan dalam skema HIBAH PENELITIAN DOSEN tahun anggaran 2014 bersifat original dan belum pernah dibiayai oleh lembaga/sumber dana lain. Bilamana dikemudian hari ditemukan ketidak-sesuaian dengan pernyataan ini, maka saya bersedia dituntut dan diproses sesuai dengan ketentuan yang berlaku dan mengembalikan seluruh biaya penelitian yang sudah diterima ke kas negara.

Demikian pernyataan ini dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Pasuruan, 16 Maret 2013

Mengetahui, yang menyatakan,

Ketua LPPM STMIK Yadika Bangil

Materai Rp 6000

**M. Imron, ST Yusron Rijal, S.Si, MT**

NIK. 09110680007 NIDN. 0716027302

1. [↑](#footnote-ref-2)