

# SKRIPSI

## RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR JAWAB KOMPUTER MENGGUNAKAN *WEBCAM* BERBASIS *BORLAND DELPHI*



Disusun oleh :  
**CAHYO WIDYANTO**  
NIM. 1012912

**JURUSAN TEKNIK ELEKTRO  
KONSENTRASI TEKNIK KOMPUTER S-1  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2012**

**LEMBAR PERSETUJUAN**

**RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR  
JAWAB KOMPUTER MENGGUNAKAN *WEBCAM* BERBASIS  
*BORLAND DELPHI***

**SKRIPSI**

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh  
Gelar Sarjana Teknik*

**Disusun oleh :**

**CAHYO WIDYANTO**

**10.12.912**

**Mengetahui,**

**Ketua Program Studi Teknik Elektro S-1**

**Ir. Yusuf Ismail Nakhoda, MT**

**NIP.Y.1018800189**

**Diperiksa dan Disetujui**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Dr. Eng. Aryuanto S, ST, MT**  
**NIP.P.1030800417**

**Ir. Yusuf Ismail Nakhoda, MT**  
**NIP.Y.1018800189**

**PROGRAM STUDI TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG**

**2012**

## SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini :

Nama : Cahyo Widyanto  
NIM : 1012912  
Program Studi : Teknik Elektro  
Konsentrasi : Teknik Komputer S-1

Dengan ini menyatakan bahwa Skripsi yang saya buat adalah hasil karya sendiri, tidak merupakan plagiasi dari karya orang lain. Dalam Skripsi ini tidak memuat karya orang lain, kecuali dicantumkan sumbernya sesuai dengan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat, dan apabila di kemudian hari ada pelanggaran atas surat pernyataan ini, saya bersedia menerima sanksinya.

Malang, 03-Agustus -2012

Yang membuat Pernyataan,



CAHYO WIDYANTO  
NIM : 1012912

## ABSTRAK

### RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR JAWAB KOMPUTER MENGGUNAKAN *WEBCAM* BERBASIS *BORLAND DELPHI*

Cahyo Widyanto, NIM 1012912

Dosen Pembimbing : Dr. Eng.Aryuanto S,ST,MT dan Ir. Yusuf Ismail Nakhoda, MT

Selama ini proses penilaian atau Pengoreksian lembar jawab komputer yang telah diaplikasikan oleh beberapa instansi pendidikan umumnya menggunakan mesin *scanner*. Namun masih banyak yang tidak mampu menggunakan mesin *scanner* dikarenakan faktor mahalnya harga mesin *scanner*. Sehingga memunculkan ide untuk merancang aplikasi Penilaian Pada Lembar Jawab Komputer Menggunakan *Webcam* Berbasis *Borland Delphi*. Dengan proses pengolahan citra untuk mendeteksi jawaban pada lembar jawab computer, menggunakan metode *region merging*, *filtering*, dan metode *threshold*.

Dengan adanya aplikasi penilaian pada lembar jawab komputer menggunakan *webcam* ini diharapkan bisa diterapkan di semua kalangan instansi khususnya bidang pendidikan. Sistem terdiri dari *hardware* dan *software*. *Hardware* terdiri dari *webcam*. Langkah-langkah yang dilakukan yaitu analisis dan perancangan sistem, pembangunan sistem dan pengujian sistem. Sistem yang dibangun dalam penelitian ini telah dapat membaca tanda hitam pada lembar jawab dan mampu mengolah nilai sesuai dengan jumlah benar dan jumlah salah. Tingkat keakuratan ditentukan oleh posisi kertas pada proses penilaian dan pencahayaan pada saat penilaian.

Kata Kunci: *Webcam*, *pixel*, *scanner*, *region merging*, *filtering*, *threshold*.

## KATA PENGANTAR

Puji Syukur ke hadirat Tuhan Yang Maha Kuasa atas segala limpahan berkat dan rahmat-Nya sehingga pembuatan skripsi yang berjudul Rancang Bangun Aplikasi Penilaian Pada Lembar Jawab Komputer Menggunakan *Webcam* Berbasis *Borland Delphi* dapat terselesaikan. Skripsi ini dibuat untuk memenuhi salah satu syarat dalam memperoleh gelar sarjana teknik. Ucapan terima kasih yang sebesar-besarnya kami sampaikan pada:

1. Bapak Rektor Institut Teknologi Nasional Malang
2. Bapak Dekan Fakultas Teknologi Industri ITN Malang
3. Bapak Ketua Jurusan Elektro
4. Rekan-rekan angkatan 2010 Teknik Elektro ITN Malang
5. Semua pihak yang telah membantu dalam penulisan dan penyusunan tugas akhir ini.

Penulis menyadari bahwa penelitian ini masih jauh dari sempurna, untuk itu kritik dan saran dari pembaca sangat penulis harapkan untuk perbaikan penelitian ini.

Malang, Agustus 2012

Penulis

## DAFTAR ISI

|                                 |             |
|---------------------------------|-------------|
| <b>Halaman Judul</b> .....      | <b>i</b>    |
| <b>Lembar Persetujuan</b> ..... | <b>ii</b>   |
| <b>Surat Pernyataan</b> .....   | <b>iii</b>  |
| <b>Abstrak</b> .....            | <b>iv</b>   |
| <b>Kata Pengantar</b> .....     | <b>v</b>    |
| <b>Daftar Isi</b> .....         | <b>vi</b>   |
| <b>Daftar Tabel</b> .....       | <b>viii</b> |
| <b>Daftar Gambar</b> .....      | <b>ix</b>   |

### **Bab I Pendahuluan**

|                                  |   |
|----------------------------------|---|
| 1.1 Latar Belakang .....         | 1 |
| 1.2 Rumusan Masalah .....        | 2 |
| 1.3 Batasan Masalah.....         | 2 |
| 1.4 Tujuan Penelitian.....       | 2 |
| 1.5 Manfaat Penelitian.....      | 2 |
| 1.6 Metode Penelitian.....       | 2 |
| 1.7 Sistematika Pembahasan ..... | 3 |

### **Bab II Tinjauan Pustaka**

|  |    |
|--|----|
| 2.1 Webcam .....                                     | 5  |
| 2.2 Personal Computer (PC).....                      | 7  |
| 2.2.1 Unit Masukan ( <i>Input device</i> ).....      | 8  |
| 2.2.2 Unit Pemrosesan ( <i>Process device</i> )..... | 9  |
| 2.2.3 Unit Keluaran ( <i>Output Device</i> ).....    | 9  |
| 2.3 Bahasa Pemrograman <i>Delphi</i> .....           | 9  |
| 2.3.1 Komponen.....                                  | 11 |
| 2.3.2 Lingkungan Kerja <i>Delphi</i> .....           | 13 |
| 2.3.3 AlphaControls 2010 v7.63 .....                 | 14 |
| 2.4 Metode <i>Thresholding</i> .....                 | 14 |
| 2.5 Metode <i>filtering</i> .....                    | 15 |
| 2.6 Pengolahan Citra.....                            | 16 |

|   |           |
|---|-----------|
| 2.6.1 Pengolahan Citra Warna.....                 | 19        |
| 2.6.2 Pengolahan Citra Digital.....               | 21        |
| 2.6.3 Operasi Pengolahan Citra.....               | 23        |
| <b>Bab III Perancangan Dan Pembuatan Aplikasi</b> |           |
| 3.1 Perancangan Perangkat Keras.....              | 24        |
| 3.1.1 Blok Diagram Sistem dan Prinsip Kerja.....  | 24        |
| 3.1.2 Batasan Sistem.....                         | 25        |
| 3.2 Perancangan Perangkat Keras.....              | 25        |
| 3.2.1 Kebutuhan Perangkat Keras.....              | 26        |
| 3.3 Perancangan Perangkat Lunak.....              | 27        |
| 3.3.1 Kebutuhan Perangkat Lunak.....              | 29        |
| 3.4 Desain Antar Muka.....                        | 46        |
| <b>Bab IV Implementasi dan Pembahasan</b>         |           |
| 4.1 Pengujian <i>Hardware</i> .....               | 48        |
| 4.1.2 Pengujian <i>Webcam</i> .....               | 48        |
| 4.2 Pengujian <i>Software</i> .....               | 49        |
| 4.3 Pembahasan.....                               | 59        |
| <b>Bab V Kesimpulan dan Saran</b>                 |           |
| 5.1 Kesimpulan.....                               | 60        |
| 5.2 Saran.....                                    | 60        |
| <b>Daftar Pustaka</b> .....                       | <b>61</b> |
| <b>Lampiran-lampiran</b> .....                    | <b>62</b> |

## DAFTAR TABEL

|   |    |
|---|----|
| Tabel 2.1 Contoh-contoh warna hexadesimal.....  | 20 |
| Tabel 3.1 Komponen tambahan .....   | 29 |
| Tabel 3.2 Daftar komponen <i>delphi</i> yang digunakan dalam pembuatan aplikasi ..... | 29 |
| Tabel 4.1 Pengujian mendeteksi jumlah <i>pixel</i> pada jawaban.....                  | 56 |
| Tabel 4.2 Pengujian jika terdapat bayangan pada lembar jawab .....                    | 57 |
| Tabel 4.3 Pengujian Bentuk Pengisian Jawaban.....                                     | 58 |
| Tabel 4.3 Pengujian keseluruhan sistem .....  | 60 |



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dewasa ini, perkembangan teknologi sangat pesat. Perkembangan teknologi khususnya bidang pendidikan telah membawa perubahan yang begitu cepat dan canggih. Kemajuan teknologi dibidang pendidikan yang semakin pesat telah mendorong berbagai macam kreasi dan inovasi. Salah satu contoh hasil teknologi yang diterapkan di dunia pendidikan adalah *scanner* yang digunakan dalam proses pengkoreksian lembar jawab komputer pada ujian sekolah.

Selama ini proses koreksi atau penilaian nilai pada lembar jawab komputer umumnya menggunakan *scanner* yang harganya cukup mahal, pada proses koreksi menggunakan *scanner* lembar jawab komputer harus diberi tanda hitam terlebih dahulu dengan menggunakan pensil 2B dan harus diisi bulat penuh. Selain itu peserta ujian juga dituntut untuk berhati-hati dalam mengisi bulatan-bulatan pada lembar jawab komputer, karena jika tanda hitam melebihi batas yang sudah ditentukan maka pada proses pengkoreksian menggunakan *scanner*, jawaban tersebut dianggap salah. Peserta ujian juga harus benar-benar teliti dalam mengubah jawaban mereka, seperti mengubah tanda hitam dari jawaban pertama A ke jawaban yang kedua B, jika tidak benar-benar bersih dalam menghapus tanda hitam pada jawaban pertama, maka pada proses pengkoreksian akan menganggap terdapat 2 (dua) jawaban pada 1 (satu) soal. Selama ini hal tersebut harus diterima apa adanya karena itulah batasan teknologi yang digunakan. Akibatnya, peserta yang sebenarnya bisa menjawab benar, tapi tidak bisa mengisi bulatan-bulatan jawaban dengan benar sehingga kurang penuh, akhirnya tidak lulus dan dirugikan.

Berdasarkan permasalahan diatas, timbul pemikiran untuk membuat aplikasi penilaian pada lembar jawab komputer dengan memanfaatkan *webcam* sebagai media penangkap objek (lembar jawab komputer). Peserta ujian tidak lagi terlalu berhati-hati dalam mengisi bulatan-bulatan dari jawaban yang mereka pilih. Proses pengisian maupun penggantian jawaban dapat dilakukan lebih cepat jika dibandingkan dengan cara lama. Dengan dibuatnya alat ini diharapkan peserta ujian dapat lebih cepat dalam proses pengisian atau mengganti jawaban dan yang lebih penting dengan cara ini dapat

mengurangi beban psikologi para peserta karena takut salah dalam proses pengisian bulatan pada lembar jawab komputer.

### 1.2 Rumusan Masalah

Dari uraian sebelumnya, maka diambil permasalahan yang akan dibahas dalam penelitian ini yaitu bagaimana merencanakan dan membuat suatu Aplikasi penilaian nilai pada lembar jawab komputer menggunakan *webcam* berbasis *Borland Delphi 7.0*.

### 1.3 Batasan Masalah

Batasan masalah pada skripsi ini antara lain:

1. Menggunakan Lembar jawaban komputer dengan warna dasar putih dan warna cetakan merah dengan ukuran kwarto / A4.
2. Data yang di ambil dari hasil *capture* lembar jawab komputer berupa data peserta ( nama, tanggal, kode soal, jurusan ) dan jawaban.
3. Membutuhkan pencahayaan yang terang.
4. Sistem hanya mengenali satu jawaban.
5. Bahasa pemrograman yang digunakan adalah *Borland Delphi*

### 1.4 Tujuan Penelitian

Penelitian bertujuan untuk merancang dan membangun perangkat lunak yang mampu membaca jawaban dari lembar jawaban yang di-*capture* oleh *web cam* dan mampu mencocokkan antara jawaban dengan kunci jawaban, dan mampu menghitung nilai.

### 1.5 Manfaat Penelitian

Dengan perancangan dan pembuatan alat ini maka manfaat yang akan diperoleh yaitu mempermudah peserta ujian dalam mengisi bulatan pada lembar jawaban komputer, peserta ujian tidak harus mengisi lembar jawab menggunakan pensil 2B . Selain itu waktu yang tersedia dapat dimanfaatkan lebih optimal untuk menjawab soal dengan benar karena waktu untuk menghapus dan mengganti jawaban lebih cepat.

### 1.6 Metode Penelitian

Adapun metode penelitian yang digunakan adalah sebagai berikut:

---

#### 1. Studi literatur

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

#### 2. Analisa Kebutuhan Sistem

Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem di mana nantinya akan digunakan sebagai acuan perancangan sistem.

#### 3. Perancangan dan Implementasi

Berdasarkan data dan informasi yang telah diperoleh serta analisa kebutuhan untuk membangun sistem ini, akan dibuat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat dan diimplementasikan kedalam sistem.

#### 4. Pengujian dan Evaluasi

Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, yaitu pengujian berdasarkan fungsionalitas program dimana pengujian langsung dilakukan ditempat dimana aplikasi akan diterapkan, dan akan dilakukan koreksi dan penyempurnaan program jika diperlukan.

### 1.7 Sistematika Pembahasan

Untuk mempermudah pembaca memahami tulisan ini, penulis membagi tulisan ini ke dalam beberapa bab sebagai berikut:

#### **BAB I PENDAHULUAN**

Bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian dan sistematika penulisan.

#### **BAB II DASAR TEORI**

Bab ini menjelaskan tentang teori-teori yang terkait dengan permasalahan yang diambil. Seperti pengolahan citra warna, metode

---

*thresholding*, metode *filtering*, metode *region merging*, *webcam* dan bahasa pemrograman *delphi*.

### **BAB III DESAIN DAN PERANCANGAN SISTEM**

Bab ini membahas tentang desain dan perancangan aplikasi penilaian pada lembar jawab komputer menggunakan *webcam* berbasis *delphi*.

### **BAB IV HASIL DAN PEMBAHASAN**

Bab ini memaparkan pembahasan dan hasil pengujian tentang aplikasi yang dibangun.

### **BAB V PENUTUP**

Bab ini berisi kesimpulan dan saran yang diharapkan dapat bermanfaat untuk pengembangan pembuatan program aplikasi tersebut.

---

## BAB II

### TINJAUAN PUSTAKA

Dalam penyusunan tugas akhir ini digunakan dua jenis perangkat yaitu *hardware* dan *software*. Peralatan untuk *hardware* adalah *webcam* yang terhubung ke *PC* (*personal computer*) melalui *port USB*. Untuk peralatan *software* digunakan bahasa pemrograman *Borland Delphi 7*.

#### 2.1 Kamera Digital (*Webcam*)

*Webcam* (singkatan dari *web camera*) adalah sebutan bagi kamera *real-time* (bermakna keadaan pada saat ini juga) yang gambarnya bisa diakses atau dilihat melalui *World Wide Web*, program *instant messaging*, atau aplikasi *video call*. Istilah *webcam* juga merujuk kepada jenis kamera yang digunakan untuk keperluan ini. Sebuah *webcam* yang sederhana terdiri dari sebuah lensa standar, dipasang di sebuah papan sirkuit untuk menangkap sinyal gambar dan *casing* (*cover*) termasuk *casing* depan dan *casing* samping untuk menutupi lensa standar dan memiliki sebuah lensa di *casing* depan yang berguna untuk memasukkan gambar, kabel support yang dibuat dari bahan fleksibel, salah satu ujungnya dihubungkan dengan papan sirkuit dan ujung satu lagi memiliki *connector*, kabel ini dikontrol untuk menyesuaikan ketinggian, arah dan sudut pandang *webcam*.



Gambar 2.1 *Webcam*

Sekarang ini *web camera* yang ada di pasaran umumnya terbagi ke dalam dua tipe : *web camera permanent* (*fixed*) dan *revolving web camera*. Pada *web camera permanent* terdapat pengapit lensa standar di posisi yang diinginkan untuk menangkap gambar pengguna. Sedangkan pada *revolving web camera* terdapat landasan dan lensa

standar dipasang di landasan tersebut sehingga dapat disesuaikan ke sudut pandang yang terbaik untuk menangkap gambar pengguna.

*Webcam* memiliki fitur dan setting yang bermacam-macam, diantaranya adalah:

1. *Motion sensing*

*Web camera* akan mengambil gambar ketika kamera mendeteksi gerakan.

2. *Image archiving*

Pengguna dapat membuat sebuah *archive* yang menyimpan semua gambar dari *web camera* atau hanya gambar-gambar tertentu saat interval pre-set.

3. *Video messaging*

Beberapa program *messaging* mendukung fitur ini.

4. *Advanced connection*

Menyambungkan perangkat *home theater* ke *web camera* dengan kabel maupun nirkabel.

5. *Automotion*

Kamera robotic yang memungkinkan pengambilan gambar secara *pan* atau *tilt* dan setting program pengambilan frame berdasarkan posisi kamera.

6. *Streaming media*

Aplikasi professional setup *web camera* dapat menggunakan kompresi MPEG4 untuk mendapatkan streaming audio dan video yang sesungguhnya.

7. *Custom coding*

Mengimport kode komputer pengguna untuk memberi tahu *web camera* apa yang harus dilakukan (misalnya *automatically refresh*).

8. *AutoCam*

Memungkinkan pengguna membuat *web page* untuk *web camera* secara gratis di server perusahaan pembuat *camera*.

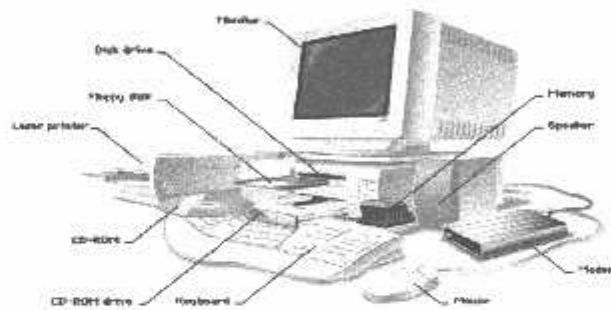
Ada beberapa permasalahan yang dihadapi *webcam*. Secara fisik, kamera-kamera yang berada di pasaran memiliki kesulitan untuk memenuhi kebutuhan personal

---

pengguna karena desainnya yang cukup bergaya namun hanya memiliki sedikit variasi. Lalu, sudut pandang *webcam* disesuaikan secara tidak langsung dalam cara yang tidak nyaman. Pengguna juga banyak menemui kesulitan ketika akan menyesuaikan posisi atau sudut pandang *webcam* pada saat proses pengambilan gambar. (<http://id.wikipedia.org/wiki/webcam>).

## 2.2 PC (Personal Computer)

Komputer adalah suatu alat elektronik yang dapat menerima *input* data, mengolah data, memberikan informasi, menggunakan suatu program yang tersimpan di memori komputer (*stored program*), dapat menyimpan program dan hasil pengolahan, serta bekerja secara otomatis.



Gambar 2.2 Personal Computer ( PC )

Suatu proses pengolahan data terdiri dari 3 tahapan dasar, yaitu *input*, *processing* dan *output*. Ketiga tahap ini disebut dengan siklus pengolahan data (*data processing cycle*). Pada siklus pengolahan data, *input* merupakan tahap awal dimana data dimasukkan ke dalam proses komputer lewat alat *input* (*input device*). *Processing* merupakan tahap berikutnya yaitu proses pengolahan terhadap data yang sudah masuk yang dilakukan oleh alat pemroses (*processing device*), yang dapat berupa proses menghitung, membandingkan, mengklasifikasikan, mengurutkan, mengendalikan atau mencari di *storage*. Kemudian tahap terakhir adalah *Output* yang merupakan proses menghasilkan *output* dari hasil pengolahan data ke alat *output* (*output device*), yaitu berupa informasi dan lainnya.



Gambar 2.3 Siklus Pengolahan Data

### 2.2.1 Unit Masukan (*Input device*)

Unit ini berfungsi untuk memasukkan data dari luar ke dalam suatu memori dan *processor* untuk diolah guna menghasilkan informasi yang diperlukan. *Input device* atau unit masukan yang umumnya digunakan oleh *personal computer* (PC) adalah keyboard dan mouse. Keyboard dan mouse adalah unit yang menghubungkan user (pengguna) dengan komputer. Selain itu terdapat *joystick* yang biasa digunakan untuk bermain game atau permainan dengan komputer. *Input device* berfungsi sebagai media untuk memasukkan data dari luar sistem ke dalam suatu memori atau *processor* untuk diolah dan menghasilkan informasi yang diperlukan. Data yang dimasukkan ke dalam sistem komputer dapat berbentuk *signal input* dan *maintenance input*. *Signal input* berbentuk data yang dimasukkan ke dalam sistem komputer, sedangkan *maintenance input* berbentuk program yang digunakan untuk mengolah data yang dimasukkan. Jadi *Input device* selain digunakan untuk memasukkan data dapat pula digunakan untuk memasukkan program. Berdasarkan sifatnya, peralatan input dapat digolongkan menjadi dua yaitu:

1. Peralatan input langsung, yaitu input yang dimasukkan langsung diproses di alat pemrosesan.

Contohnya: Keyboard, mouse, *touch screen*, light pen, *scanner*.

2. Peralatan input tidak langsung, yaitu input yang melalui media tertentu sebelum suatu input diproses oleh alat pemroses.

Contohnya: punched card, disket, harddisk.

---



### 2.2.2 Unit Pemrosesan (*Process device*)

Otak sebuah komputer berada pada unit pemrosesan (*Process device*). Unit pemrosesan dinamakan CPU (*Central Processing Unit*). Fungsi CPU adalah sebagai pemroses dan pengolah data yang selanjutnya dapat menghasilkan suatu informasi yang diperlukan. Pada komputer mikrounit, pemrosesan ini disebut dengan *micro-processor* atau *processor* yang berbentuk *chip* yang terdiri dari ribuan sampai jutaan *IC*. Fungsi utama dari CPU bekerja dengan aritmatika dan logika terhadap data yang terdapat dalam memori atau yang dimasukkan melalui unit masukan. Perkembangan *processor* yang pertama kali muncul tahun 1990-an adalah pentium dengan kecepatan 75 Mega Hertz, dan saat ini kecepatannya sudah mencapai 3 Giga Hertz lebih dengan *processor* pentium IV. Seiring dengan kecepatan Pentium IV, telah pula diperkenalkan *processor* dengan teknologi *mobile* yaitu Centrino (Pentium M-Centrino) jenis *processor* ini baru terdapat pada komputer-komputer *built-up*, laptop, notebook. Saat ini, *processor* yang terbaru adalah *Dual Core (Core duo)*. *Processor* ini memiliki dua kecepatan seperti memiliki dua *processor*. Beberapa produsen *processor* yang terkenal adalah Intel, AMD, dan Cyrix.

### 2.2.3 Unit Keluaran (*Output Device*)

*Output device* bisa diartikan sebagai peralatan yang berfungsi untuk mengeluarkan hasil pemrosesan ataupun pengolahan data yang berasal dari CPU ke dalam suatu media yang dapat dibaca oleh manusia ataupun dapat digunakan untuk penyimpanan data hasil proses. Jenis dan media dari *output device* yang dimiliki oleh komputer cukup banyak. Macam-macam *output device* (unit keluaran) antara lain monitor, printer, dan speaker.

## 2.3 Bahasa Pemrograman *Delphi*

*Delphi* adalah paket bahasa pemrograman yang bekerja dalam sistem windows. *Delphi* merupakan bahasa pemrograman yang mempunyai cakupan kemampuan yang luas dan sangat canggih. Berbagai aplikasi dapat dibuat dengan *delphi*, termasuk aplikasi untuk mengolah teks, grafik, angka, *database* dan aplikasi web.

Secara umum, kemampuan *delphi* adalah menyediakan komponen-komponen dan bahasa pemrograman yang canggih, sehingga memungkinkan untuk membuat program aplikasi sesuai dengan keinginan, dengan tampilan dan kemampuan yang

---

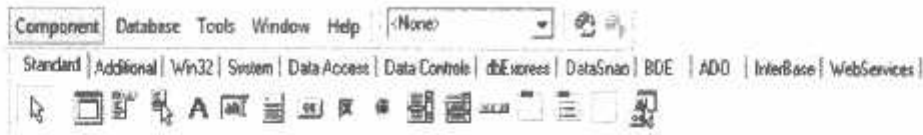
canggih. Untuk mempermudah dalam membuat program aplikasi, *delphi* menyediakan fasilitas pemrograman yang sangat lengkap. Fasilitas pemrograman tersebut dibagi dalam dua kelompok, yaitu obyek dan bahasa pemrograman. Secara ringkas, obyek adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat (visual). Obyek biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman secara singkat dapat disebut sebagai kumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan serta untuk menjalankan tugas tertentu.

*Delphi* menggunakan bahasa obyek Pascal sebagai bahasa dasar. Sehingga shintax dari kode *Delphi* hampir sama dengan Pascal. Selain itu *Delphi* juga mendukung penulisan kode bahasa assembly dalam penulisan kodenya sehingga memudahkan dalam membuat program bahasa mesin. Khusus untuk pemrograman *database*, *delphi* menyediakan obyek yang sangat kuat, canggih dan lengkap, sehingga mudah dalam merancang, membuat dan menyelesaikan aplikasi *database* yang diinginkan (M.Agus J.Alam, jakarta,2003).

Pada dasarnya tujuan akhir memprogram adalah menghasilkan sebuah program aplikasi berbentuk file yang dapat dieksekusi (*executable file* yang berekstensi *exe*) dari MS-DOS *prompt*, *Windows Explorer* atau menu *Run* pada MS Windows. Pemrogram harus melakukan serangkaian langkah untuk menyusun komponen – komponen program yang tersimpan dalam sejumlah file yang disebut dengan *source code*. Proses pembuatan sebuah file *exe* dari sejumlah file *source code* disebut dengan kompilasi. Penggunaan *delphi* akan mempermudah pemrogram dalam proses kompilasi ini, karena proses ini sudah dilakukan *delphi* secara langsung pada saat pemrogram melakukan *running* terhadap *source code*. *Source code* sebuah program aplikasi terdiri dari file, *dfm*, file *pas* dan file *dpr*. File *dpr* pasti hanya sebuah, sedangkan file lainnya dapat lebih dari sebuah. Tentu saja apabila dalam sebuah program yang diinginkan melibatkan lebih dari satu form, file, *dfm* dan *pas* akan sebanyak form yang dilibatkan. Di samping itu, pemrogram dapat menambahkan lagi file unit tertentu jika diinginkan. Semua file *source code* dapat dilihat pada *window project manager* (Zainudin Zuhri, 2003).

---

### 2.3.1 Komponen



Gambar 2.4. *Component palette*

Dalam membuat program, *delphi* telah menyediakan banyak kemudahan, yaitu dengan disediakan komponen-komponen. Komponen ini merupakan sebuah prosedur/program yang sudah di kompilasi dan langsung dapat digunakan, sesuai dengan fungsinya masing-masing.

Untuk menggunakan komponen ini kita dapat meng-klik komponen yang diinginkan, kemudian kita klik di form, maka komponen tersebut akan muncul di form.

Kegunaan beberapa komponen :

#### 3. Button/ Bitbtn

Biasa digunakan sebagai tombol kendali. Perbedaan antara bitbtn dengan btn : pada bitbtn dapat disisipkan warna pada tombol dan icon tertentu, lain halnya bila menggunakan btn.

#### 4. Panel

Panel berfungsi untuk mengelompokkan komponen-komponen didalamnya.

#### 5. Label

Kita dapat menamakan atau memberi keterangan pada program.

#### 6. Edit

Edit berfungsi sebagai masukan data (input) dalam bentuk string, dari bentuk string dapat mengolahnya menjadi bentuk integer atau bentuk lainnya. Yang kemudian dapat digunakan untuk operasi selanjutnya.

#### 7. Chart

Data-data yang telah kita analisa, dapat ditampilkan ke dalam grafik, sehingga mudah untuk menganalisisnya.

8. Stringgrid  
Stringgrid berguna untuk menaruh data string kedalam bentuk kolom tabel, seperti pada Excel. Maka harus mengubah type data ke dalam bentuk string bila data yang ingin ditampilkan data bukan string.
  9. PopupMenu  
PopupMenu berfungsi sebagai perintah yang aktif bila kita meng-klik kanan mouse, Untuk mengaktifkannya harus mengaktifkan popup menu pada komponen yang diinginkan, caranya : ubah pada object inspector.
  10. MainMenu  
Contoh main menu adalah Option pada tiap aplikasi program, dengan komponen ini, maka bisa menaruh fungsi-fungsi program seperti pada aplikasi umumnya.
  11. ComboBox  
Combo Box berfungsi sebagai petunjuk untuk pemilihan berbagai masukan.
  12. CheckBox  
Bila komponen ini di check maka ada aplikasi yang bisa disetting untuk bekerja dibawahnya.
  13. RadioButton  
*Prinsip kerjanya hampir sama dengan check box, cuma tampilannya saja yang berbeda.*
  14. MediaPlayer  
Biasa digunakan untuk menyalakan atau memainkan musik (format wav atau midi) dan menjalankan film (format avi).
  15. Timer  
Timer berfungsi sebagai jam yang telah disediakan Delphi. Timer dapat mendecode time, sehingga dapat terjadi akuisisi data.
  16. Serversocket  
Digunakan di server, yang akan menyediakan layanan koneksi ke client.
-

### **Component Palette**

Berisi ikon-ikon komponen visual dan nonvisual yang dapat digunakan untuk merancang antarmuka bagi pemakai aplikasi. Component Palette terdiri atas beberapa page yang dipakai sebagai pengelompok jenis komponen misalnya, Standard, Additional, System, Data Access, dan sebagainya.

### **Object Inspector**

Object inspector digunakan untuk mengubah karakteristik sebuah komponen. Pada object inspector terdapat 2 tab yaitu Properties dan Event. Tab tersebut dapat diaktifkan salah satu dengan mengklik Properties atau Event. Pada tab properties, bisa digunakan untuk mengubah property dari komponen yang telah dibuat. Sedangkan pada tab event, dapat digunakan untuk menyisipkan kode dalam menanggapi kejadian tertentu. Kejadian bisa dibangkitkan karena beberapa hal, seperti pengklikan mouse, penekanan tombol keyboard, penutupan jendela, dan sebagainya. Misalnya onClick, yang dibangkitkan bila mengklik form.

### **Object TreeView**

Object TreeView berisi daftar komponen yang telah diletakkan pada Form Designer. Sebagai contoh, jika meletakkan 2 komponen, TgroupBox dan TLabel, Object TreeView.

### **2.3.3 AlphaControls 2010 v7.63**

Alpha Controls adalah suatu komponen tambahan yang didalamnya terdapat kumpulan library-library skin dan juga komponen-komponen yang dapat digunakan pada Borland Delphi 5, 6, 7, 2007, 2010, XE2 and C++ Builder. Alpha Controls secara umum digunakan untuk memperindah tampilan dari form dan juga interface dari aplikasi yang dibuat.

Disamping untuk memperindah tampilan dari aplikasi yang dibuat, dalam AlphaControls juga terdapat komponen-komponen lainya seperti Dialog (*path, open, save*).

## **2.4. Metode Thresholding**

Thresholding digunakan untuk mengatur jumlah derajat keabuan yang ada pada citra. Dengan menggunakan thresholding maka derajat keabuan bisa diubah sesuai

---

keinginan, misalkan diinginkan menggunakan derajat keabuan 16, maka tinggal membagi nilai derajat keabuan dengan 16. Proses thresholding ini pada dasarnya adalah proses perubahan kuantisasi pada citra, sehingga untuk melakukan thresholding dengan derajat keabuan dapat digunakan rumus:

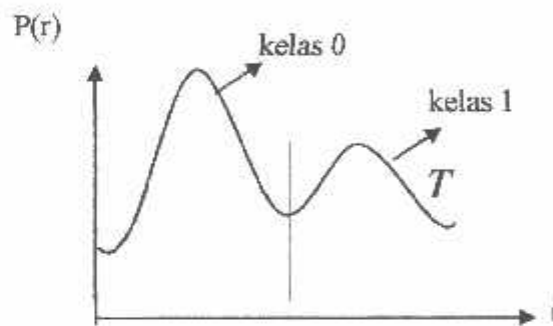
$$x = \frac{w}{b}$$

dimana :

w adalah nilai derajat keabuan sebelum thresholding

b adalah jumlah derajat keabuan yang diinginkan

x adalah nilai derajat keabuan setelah thresholding



Gambar 2.6 Penentuan nilai ambang  $T$

(Sumber : Rinaldi, 2004)

Pengembangan secara local dilakukan terhadap daerah-daerah di dalam citra. Dalam hal ini citra dipecah menjadi bagian-bagian kecil, kemudian proses pengembangan dilakukan secara local. Nilai ambang untuk setiap bagian belum tentu sama dengan bagian lain. Sebagai contoh, pengembangan dilakukan terhadap daerah citra yang berukuran  $5 \times 5$  atau  $10 \times 10$  *pixel*. Nilai ambangnya ditentukan sebagai fungsi rata-rata derajat keabuan di dalam daerah citra tersebut. Intensitas *pixel* yang berbeda secara signifikan dari nilai rata-rata tersebut dianggap mengandung informasi kontras dan ini harus dipertahankan di dalam citra biner.

## 2.5 Metode Filtering

*Filtering* adalah suatu proses dimana diambil sebagian sinyal dari frekwensi tertentu, dan membuang sinyal pada frekwensi yang lain. *Filtering* pada citra juga

menggunakan prinsip yang sama, yaitu mengambil fungsi citra pada frekwensi-frekwensi tertentu dan membuang fungsi citra pada frekwensi-frekwensi tertentu.

Dari sifat-sifat citra pada bidang frekwensi, maka prinsip-prinsip *filtering* dapat dikembangkan adalah sebagai berikut:

- (1) Bila ingin mempertahankan gradiasi atau banyaknya level warna pada suatu citra, maka yang dipertahankan adalah frekwensi rendah dan frekwensi tinggi dapat dibuang atau dinamakan dengan Low Pass Filter. Hal ini banyak digunakan untuk reduksi noise dan proses blur.
- (2) Bila ingin mendapatkan threshold atau citra biner yang menunjukkan bentuk suatu gambar maka frekwensi tinggi dipertahankan dan frekwensi rendah dibuang atau dinamakan dengan High Pass Filter. Hal ini banyak digunakan untuk menentukan garis tepi (*edge*) atau sketsa dari citra.
- (3) Bila ingin mempertahankan gradiasi dan bentuk, dengan tetap mengurangi banyaknya bidang frekwensi (*bandwidth*) dan membuang sinyal yang tidak perlu maka frekwensi rendah dan frekwensi tinggi dipertahankan, sedangkan frekwensi tengahan dibuang atau dinamakan dengan Band Stop Filter. Teknik yang dikembangkan dengan menggunakan Wavelet Transform yang banyak digunakan untuk kompresi, restorasi dan denoising.

## 2.6 Pengolahan Citra (*Image Processing*)

Pengolahan citra merupakan proses pengolahan dan analisa citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra.

Meskipun sebuah citra kaya informasi, namun seringkali citra yang dimiliki mengalami penurunan mutu (*degradasi*), misalnya mengandung cacat atau derau (*noise*), warnanya terlalu kontras, kurang tajam, kabur (*blurring*), dan sebagainya. Tentu saja citra semacam ini menjadi lebih sulit diinterpretasi karena informasi yang disampaikan oleh citra tersebut menjadi berkurang. Agar citra yang mengalami gangguan mudah diinterpretasi (baik oleh manusia maupun mesin), maka citra tersebut perlu dimanipulasi menjadi citra lain yang kualitasnya lebih baik. Bidang studi yang

---

menyangkut hal ini adalah pengolahan citra (*image processing*). Pengolahan citra adalah pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya lebih baik.

Pengolahan citra bertujuan:

1. memperbaiki kualitas gambar agar atau lebih mudah diinterpretasi oleh manusia atau mesin komputer, dilihat dari aspek radiometrik (peningkatan kontras, transformasi warna, restorasi citra) sedangkan dari aspek geometrik (rotasi, translasi, skala, transformasi geometrik).
2. melakukan proses penarikan informasi atau deskripsi obyek atau pengenalan obyek yang terkandung pada citra.
3. melakukan kompresi atau reduksi data untuk tujuan penyimpanan data, transmisi data, dan waktu proses data.

Berikut ini adalah definisi dasar yang dipergunakan dalam pengolahan citra:

- a. Citra adalah gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang kontinu menjadi gambar diskrit melalui proses sampling yang diwakili oleh bit-bit tertentu.
- b. Sampling adalah proses untuk menentukan warna pada piksel tertentu pada citra dari sebuah gambar yang kontinu. Pada proses sampling biasanya dicari warna rata-rata dari gambar analog yang kemudian dibulatkan. Proses sampling sering disebut juga dengan proses digitisasi.
- c. Gambar analog dibagi menjadi  $N$  baris dan  $M$  kolom sehingga menjadi gambar diskrit. Persilangan antara baris dan kolom tertentu tersebut disebut dengan piksel. Contohnya adalah gambar/titik diskrit pada baris  $n$  dan kolom  $m$  disebut dengan piksel  $(n,m)$ .

Ada kalanya, dalam proses sampling, warna rata-rata yang didapat direlasikan ke level warna tertentu. Contohnya apabila citra hanya terdapat 16 tingkatan warna abu-abu, maka nilai rata-rata yang didapat dari proses sampling harus direlasikan ke 16 tingkatan tersebut. Format data citra berhubungan erat dengan warna. Pada kebanyakan kasus, terutama untuk keperluan penampilan visual, nilai data merepresentasikan warna dari citra yang diolah. Format citra yang banyak dipakai adalah Citra biner. Proses

---



mengasosiasikan warna rata-rata dengan tingkatan warna tertentu disebut dengan kuantisasi.

Secara matematis, citra merupakan fungsi terus menerus (*continue*) dari intensitas cahaya pada dua bidang. Citra terbagi menjadi dua jenis:

1. Citra diam (*still image*)  
Citra tunggal yang tidak bergerak.
2. Citra bergerak (*moving images*)

Peralatan Pengolahan Citra terdiri dari:

1. Sistem Akuisisi Citra (*Image Acquisition System*) yang diperlukan untuk mendapatkan citra, terdiri dari sensor (mengubah gelombang elektro- magnetik menjadi sinyal listrik) dan pendigitasi (*digitizer*) dimana mengubah sinyal listrik analog menjadi sinyal digital. Contohnya kamera dan *scanner*.
2. Penayang berupa monitor TV untuk menampilkan citra pada layar kaca  
Pencetak untuk mendapatkan hasil dalam bentuk cetakan seperti foto, transparansi atau slide.
3. Pengolahan dalam bentuk algoritma.
4. Penyimpanan citra dibedakan 3 jenis: a. untuk jangka pendek berupa card, b. untuk tempat terpasang berbentuk disket magnetik, c. untuk arsip berupa pita magnetik dan disket optis.
5. Komunikasi untuk pengiriman maupun penerimaan citra. Contoh dengan satelit.

Beberapa operasi-operasi dalam pengolahan citra:

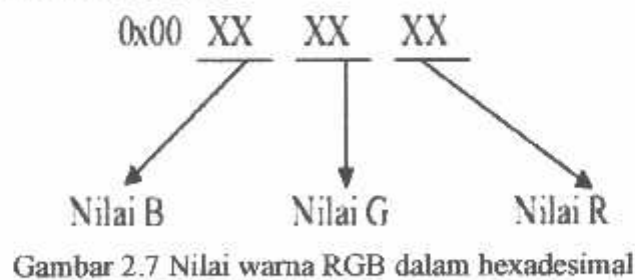
1. Perbaikan kualitas citra (*image enhancement*) dengan memanipulasi parameter citra. Contohnya :
    - perbaikan kontras gelap/terang
    - penajaman (*sharpening*)
    - perbaikan tepian obyek (*edge enhancement*)
    - pemberian warna semu (*pseudocoloring*)
    - penapisan derau (*noise filtering*)
-

2. Pemugaran Citra (*image restoration*) untuk menghilangkan/meminimumkan cacat pada citra. Contohnya: penghilangan kesamaran (*deblurring*), penghilangan derau (*noise*)
3. Pemampatan citra (*image compression*) dari memori besar ke memori kecil dengan metode JPEG. Contohnya, citra boat.bmp (258 KB) dimampatkan menjadi citra boat.jpg (49 KB).
4. Segmentasi Citra (*image segmentation*) bertujuan untuk memecah suatu citra ke dalam beberapa segmen dengan kriteria tertentu.
5. Pengorakan Citra (*image analysis*) bertujuan menghitung besaran kuantitatif dari citra untuk menghasilkan deskripsinya. Tekniknya dengan mengekstraksi ciri tertentu yang membantu dalam identifikasi obyek. Kadangkala dengan proses segmentasi. Contoh :
  - pendeteksian tepi obyek (*edge detection*)
  - ekstraksi batas (*boundary*)
  - representasi daerah (*region*)
6. Rekonstruksi citra (*image reconstruction*) bertujuan membentuk ulang obyek dari beberapa citra hasil proyeksi. Misalnya beberapa foto rontgen dengan sinar X digunakan untuk membentuk ulang gambar organ tubuh.

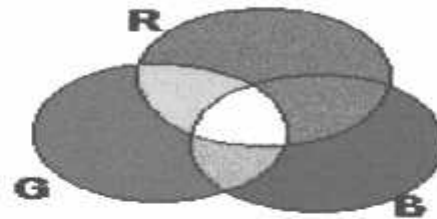
### 2.6.1 Pengolahan Citra Warna

Dasar pengolahan citra adalah pengolahan warna RGB pada posisi tertentu. Dalam pengolahan citra warna dipresentasikan dengan nilai hexadesimal dari 0x00000000 sampai 0x00ffffff. Warna hitam adalah 0x00000000 dan warna putih adalah 0x00ffffff.

Definisi nilai warna diatas seperti gambar 2.6, variable 0x00 menyatakan angka dibelakangnya adalah hexadesimal.



Terlihat bahwa setiap warna mempunyai range nilai 00 (angka desimalnya adalah 0) dan ff (angka desimalnya adalah 255), atau mempunyai nilai derajat keabuan  $256 = 2^8$ . Dengan demikian range warna yang digunakan adalah  $(28)(28)(28) = 224$  (atau yang dikenal dengan istilah *True Colours* pada *Windows*). Nilai warna yang digunakan diatas merupakan gabungan warna cahaya merah, hijau dan biru seperti yang terlihat pada gambar 2.7. Sehingga untuk menentukan nilai dari suatu warna yang bukan warna dasar digunakan gabungan skala kecerahan dari setiap warna.



Gambar 2.8 Komposisi warna RGB

Dari definisi diatas untuk menyajikan warna tertentu dapat dengan mudah dilakukan, yaitu dengan mencampurkan ketiga warna dasar RGB. Berikut memperlihatkan contoh-contoh warna yang bisa digunakan.

Untuk mengetahui kombinasi warna, perlu dibuat suatu program yang dapat menampilkan sesuai dengan nilai yang dimasukkan sehingga dapat dicoba berbagai warna kombinasi RGB seperti gambar 2.7.

Tabel 2.2 Contoh-contoh Warna dalam Hexadesimal

|                      |                          |
|----------------------|--------------------------|
| Hitam = 0x00000000   | Orange = 0x0000A0FF      |
| Merah = 0x000000FF   | Abu-abu = 0x00888888     |
| Hijau = 0x00FF00     | Ungu muda = 0x00FF00AA   |
| Biru = 0x00FF0000    | Hijau muda = 0x00A0FF00  |
| Kuning = 0x0000FFFF  | Merah muda = 0x00A000FF  |
| Magenta = 0x00FF00FF | Kuning muda = 0X00A0FFFF |
| Cyan = 0x00FFFF00    | Coklat = 0x000088AA      |
| Putih = 0x00FFFFFF   | Ungu = 0x00A00088        |

### 2.6.2 Pengolahan Citra Digital

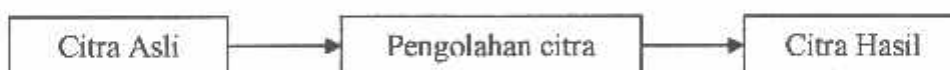
Meski sebuah citra kaya informasi, namun seringkali citra yang kita miliki mengalami penurunan mutu (degradasi), misalnya mengandung cacat atau derau (*noise*), warnanya terlalu kontras, kurang tajam, kabur (*blurring*), dan sebagainya. Tentu saja citra semacam ini menjadi lebih sulit diinterpretasi karena informasi yang disampaikan oleh citra tersebut menjadi berkurang.

Agar citra yang mengalami gangguan mudah diinterpretasi (baik oleh manusia maupun mesin), maka citra tersebut perlu dimanipulasi menjadi citra lain yang kualitasnya lebih baik. Hal ini berkaitan dengan pengolahan citra (*image processing*). Pengolahan citra adalah pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya lebih baik.

Umumnya, operasi-operasi pada pengolahan citra diterapkan pada citra bila :

1. perbaikan atau memodifikasi citra perlu dilakukan untuk meningkatkan kualitas penampakan atau untuk menonjolkan beberapa aspek informasi yang terkandung di dalam citra,
2. elemen di dalam citra perlu dikelompokkan, dicocokkan, atau diukur,
3. sebagian citra perlu digabung dengan bagian citra yang lain.

Teknik-teknik pengolahan citra mentransformasikan citra menjadi citra lain. Jadi, masukannya adalah citra dan keluarannya juga citra, namun citra keluaran mempunyai kualitas lebih baik daripada citra masukan. Blok diagram pengolahan citra, sebagai berikut:



Gambar 2.9 Blok Diagram Pengolahan Citra

Citra digital mengandung sejumlah elemen-elemen dasar. Elemen-elemen dasar inilah yang dimanipulasi dalam pengolahan citra. Elemen-elemen dasar yang penting diantaranya adalah sebagai berikut :

- a. Kecerahan (*brightness*)

Kecerahan disebut juga sebagai intensitas cahaya. Kecerahan pada suatu titik (piksel) di dalam suatu citra sebenarnya adalah intensitas rata-rata dari suatu area yang melingkupinya.

b. Kontras (*contrast*)

Kontras menyatakan sebaran terang (*lightness*) dan gelap (*darkness*) dalam suatu citra. Citra dengan kontras rendah dicirikan oleh sebagian besar komposisi citranya adalah terang atau sebagian besar gelap. Citra dengan kontras yang baik, komposisi gelap dan terangnya tersebar secara merata.

c. Kontur (*contour*)

Kontur adalah keadaan yang ditimbulkan oleh perubahan intensitas pada *pixel-pixel* yang bertetangga. Karena adanya perubahan intensitas inilah, maka tepi-tepi (*edge*) objek pada citra dapat dideteksi.

d. Warna (*color*)

Warna adalah persepsi yang dirasakan oleh sistem visual manusia terhadap panjang gelombang cahaya yang dipantulkan oleh objek. Setiap warna mempunyai panjang gelombang yang berbeda-beda. Warna yang diterima oleh sistem visual manusia (mata) merupakan hasil kombinasi cahaya dengan panjang gelombang yang berbeda-beda. Kombinasi warna yang memberikan rentang warna yang paling lebar adalah *red (R)*, *green(G)*, dan *blue(B)*.

e. Bentuk (*shape*)

Bentuk adalah properti intrinsik dari objek tiga dimensi. Bentuk merupakan properti intrinsik utama untuk sistem visual manusia karena manusia lebih sering menginterpretasikan suatu objek berdasarkan bentuknya daripada elemen lainnya.

f. Tekstur (*texture*)

Tekstur dicirikan sebagai distribusi spasial dari derajat keabuan di dalam sekumpulan *pixel-pixel* yang bertetangga. Sehingga, tekstur tidak dapat didefinisikan untuk sebuah *pixel*. Tekstur merupakan karakteristik untuk menganalisa permukaan berbagai jenis citra objek.

Dasar pengolahan citra adalah pengolahan warna RGB pada posisi tertentu. Dalam pengolahan citra, warna dipresentasikan dengan nilai hexadesimal dari 0x00000000 sampai 0x00ffffff. Warna hitam adalah 0x00000000 dan warna putih adalah 0x00ffffff.

### 2.6.3 Operasi Pengolahan Citra

Secara umum, operasi pengolahan citra dapat diklasifikasikan dalam beberapa jenis sebagai berikut:

1. Perbaikan kualitas citra (*image enhancement*)  
Bertujuan untuk memperbaiki kualitas citra dengan cara memanipulasi parameter-parameter citra.
  2. Pemugaran citra (*image restoration*)  
Bertujuan menghilangkan / meminimumkan cacat pada citra.
  3. Pemampatan citra (*image compression*)  
Bertujuan agar citra dapat direpresentasikan dalam bentuk yang lebih kompak sehingga memerlukan memori yang lebih sedikit.
  4. Segmentasi citra (*image segmentation*)  
Bertujuan untuk memecah suatu citra ke dalam beberapa segemn dengan suatu kriteria tertentu. Metode segmentasi citra dapat dibagi menjadi empat jenis yaitu teknik thresholding, metode berbasis boundary, metode berbasis region, dan teknik hybrid yang menggabungkan kriteria boundary dan region.
  5. Pengorakan citra (*image analysis*)  
Bertujuan menghitung besaran kuantitatif dari citra untuk menghasilkan deskripsinya.
  6. Rekonstruksi citra (*image reconstruction*)  
Bertujuan untuk membentuk ulang objek dari beberapa citra hasil proyeksi.
-

## BAB III

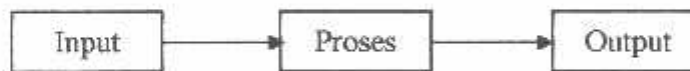
### PERANCANGAN DAN PEMBUATAN APLIKASI

Untuk dapat menghasilkan rangkaian yang dapat bekerja secara optimal harus dipahami karakteristik komponen utama dan konteks kerjanya dalam suatu rangkaian. Rangkaian sistem penilaian lembar jawab komputer ini secara umum dibagi menjadi dua bagian, yaitu *hardware* dan *software*. *Hardware* merupakan sistem perangkat keras yang meliputi sistem kontrol berupa PC (*Personal Computer*), Sedangkan *software* berupa program yang menggunakan bahasa pemrograman *Delphi 7.0*. Oleh karena itu direncanakan terlebih dahulu perangkat-perangkat yang mendukung sistem kerja alat ini.

#### 3.1 Perancangan Perangkat Keras

##### 3.1.1 Blok Diagram Sistem dan Prinsip Kerja

Untuk merencanakan suatu alat diperlukan bagian-bagian pendukung. Untuk itu dibuat suatu diagram blok untuk perancangan alat pengkoreksi lembar jawab soal berbasis PC (*Personal Computer*) menggunakan *webcam*. Pada Gambar 3.1 di bawah ini menggambarkan tentang blok diagram sistem secara keseluruhan.



Gambar 3.1 Konsep Dasar Sistem

Dari diagram blok di atas dapat dijelaskan fungsi dari masing-masing komponen, yaitu

1. Input data berupa lembar jawaban.
2. Proses pengolahan citra dengan data yang diolah berupa gambar (hasil *capture*) dalam format *bitmap*, kemudian diproses oleh sistem sehingga menghasilkan output berupa nilai.
3. Outputnya adalah jumlah benar, jumlah salah, nilai dan selanjutnya akan disimpan dalam *database*.

Prinsip kerja dari alat ini adalah sebagai berikut :

Pada langkah awal lembar jawaban soal diletakkan pada kotak yang telah disediakan. Kotak tersebut dirancang supaya lembar jawaban tepat (tidak bergeser) , dan

diatas kotak tersebut dipasang sebuah kamera (*WebCam A4TECH*) dengan jarak yang telah disesuaikan (29,4 cm) sehingga hasil penangkapan gambar dapat terlihat dengan jelas dan mencakup area lembar jawab yang akan dideteksi. Kemudian melalui *software, webcam* akan menangkap gambar lembar jawaban dalam format *bitmap* kemudian diolah *software* (pengolahan citra warna) untuk mengetahui jawaban yang telah diberi tanda, serta menangkap data. Dari data tersebut akan diketahui jawaban dari masing-masing lembar jawab yang kemudian dibandingkan dengan kunci jawaban. Semua data pengolahan tersebut disimpan sebagai *database* sehingga data-data tersebut dapat disimpan dan dipergunakan kembali jika dibutuhkan.

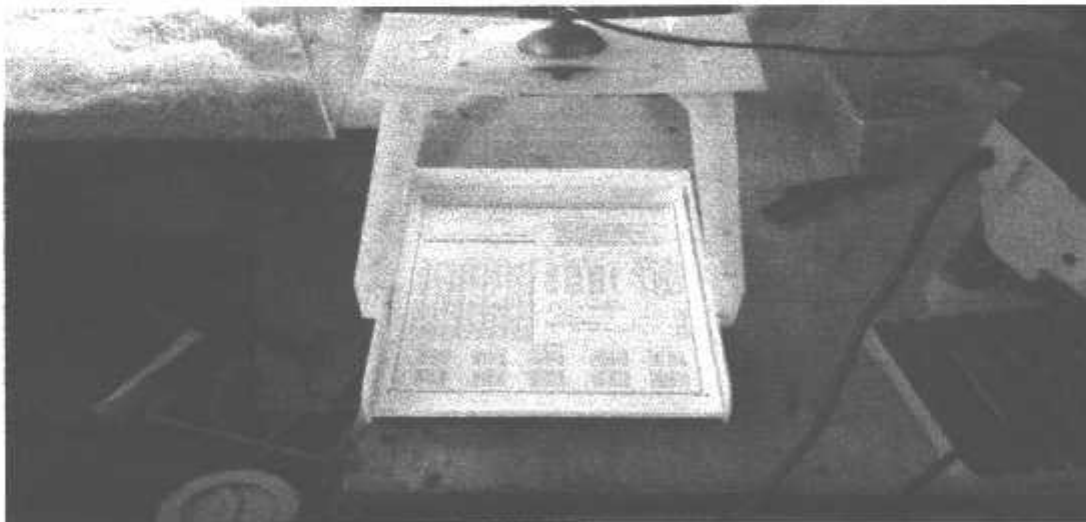
### 3.1.2 Batasan Sistem

Batasan-batasan sistem adalah:

- a. Lembar jawab menggunakan ukuran kwarto A4
- b. Hasil *capture webcam* berupa format BMP (*bitmap*)
- c. Sistem hanya mengenali satu (1) jawaban benar.
- d. *Database* disimpan dalam *format file .csv*.

### 3.2 Perancangan Perangkat Keras (mekanik)

Perancangan mekanik ini sangat diperlukan supaya penangkapan gambar oleh *webcam* mendapatkan hasil yang sesuai dengan keinginan serta agar letak lembar jawaban selalu tepat antara lembar jawaban satu dengan lembar jawaban yang lain. Adapun rancangan mekanik dapat dilihat dalam gambar 3.2 dibawah ini:



Gambar 3.2 Perancangan Perangkat Keras



### 3.2.1 Kebutuhan Perangkat Keras

Kebutuhan perangkat keras yang dibutuhkan dalam pembuatan sistem penilaian lembar jawab komputer ini adalah sebagai berikut:

1. Perangkat PC atau laptop minimal Pentium III
2. Lembar jawab komputer standart ujian nasional ukuran kwarto / A4 (210 x 297 mm).seperti pada gambar 3.3:

The image shows a computer answer sheet form. At the top, it is titled "LEMBAR JAWAB" and "LEMBAR JAWAB KOMPUTER". Below the title, there are several sections: a header area with fields for "No. Peserta", "No. Urut", "No. Soal", and "No. Jawaban"; a section for "Jawaban" with a grid of boxes for marking answers; and a section for "Keterangan" with a grid of boxes for marking correct or incorrect answers. The form is designed for computer-based grading of multiple-choice questions.

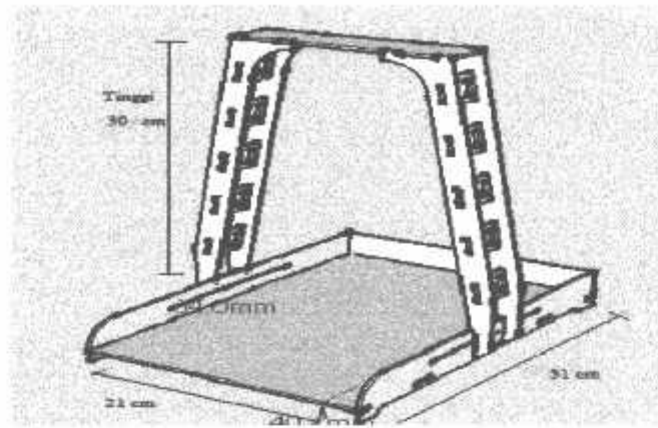
Gambar 3.3 Form Lembar Jawab Komputer

3. Satu Buah webcam.



Gambar 3.4 USB Webcam

4. Wadah atau tempat lembar jawaban soal yang akan dipotret

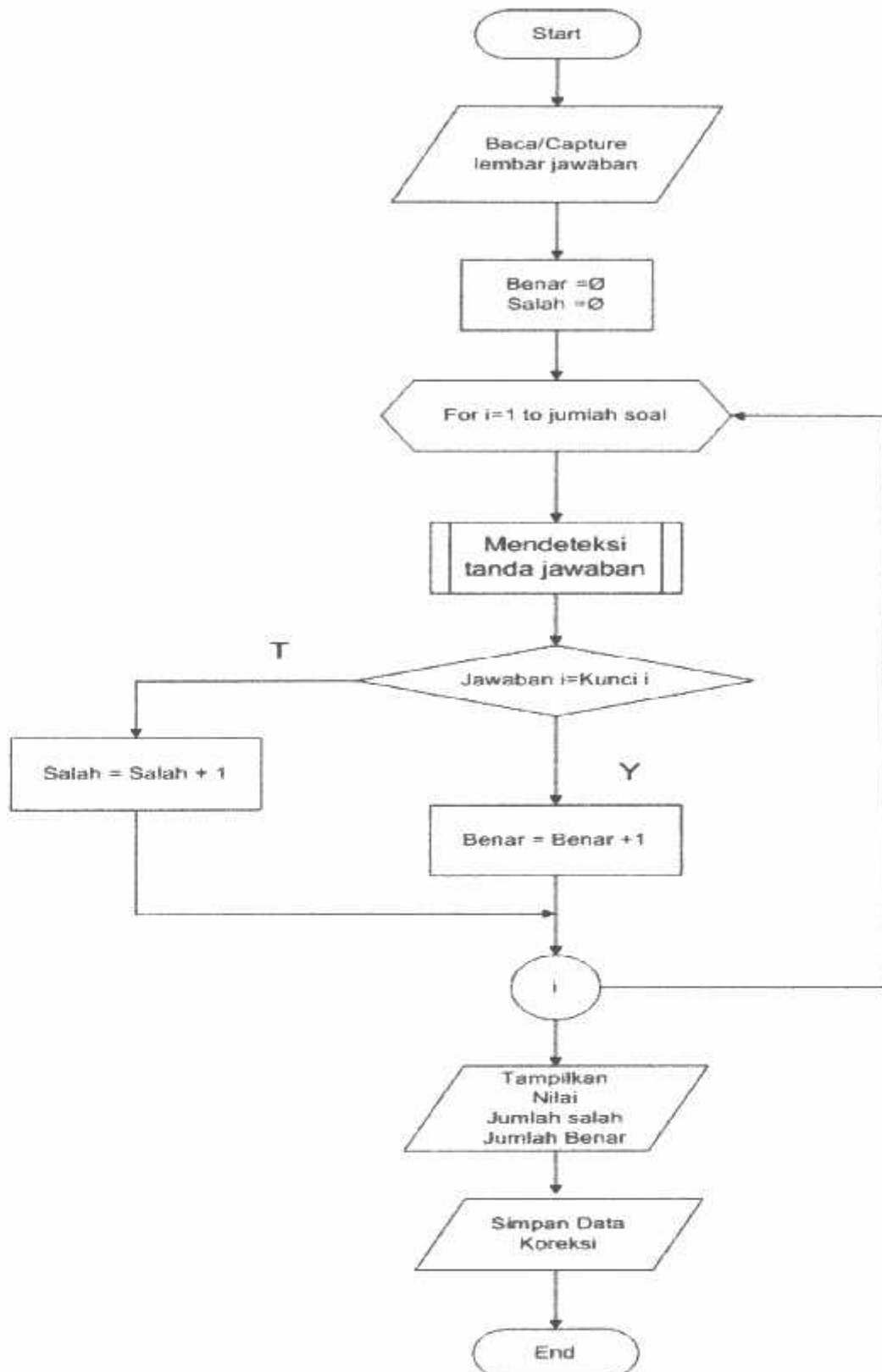


Gambar 3.5 Tempat Lembar Jawaban Komputer

### 3.3 Perancangan Perangkat Lunak

Perancangan *software* untuk menjalankan alat ini adalah dalam bentuk *flowchart* yang kemudian diimplementasikan menjadi program menggunakan bahasa pemrograman *Borland Delphi 7*. *Delphi* merupakan perangkat pengembangan aplikasi yang sangat terkenal di lingkungan Windows. Dengan menggunakan perangkat lunak ini dapat dibangun berbagai aplikasi Windows (permainan, multimedia, grafis, *database* dan sebagainya) dengan cepat dan mudah. Dengan menggunakan pendekatan visual, maka dapat diciptakan aplikasi yang canggih tanpa banyak menuliskan kode. *Delphi* menggunakan bahasa *object Pascal* sebagai bahasa dasar. Jika telah menguasai Pascal, maka akan dengan mudah memahami program *Delphi*.

Adapun *flowchart* untuk alat penilaian lembar jawab komputer berbasis menggunakan *webcam* berbasis *Borland Delphi* terlihat pada Gambar 3.3 :



Gambar 3.6 *Flowchart* Sistem Kerja program

### 3.3.1 Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak yang dibutuhkan dalam pembuatan sistem ini adalah sebagai berikut:

1. Untuk menangkap objek pada *delphi* menggunakan kamera maka dibutuhkan komponen tambahan yaitu DSPack.
2. Agar tampilan pada program *delphi* lebih menarik maka dibutuhkan komponen tambahan berupa AlphaControls.

Tabel 3.1 Komponen tambahan

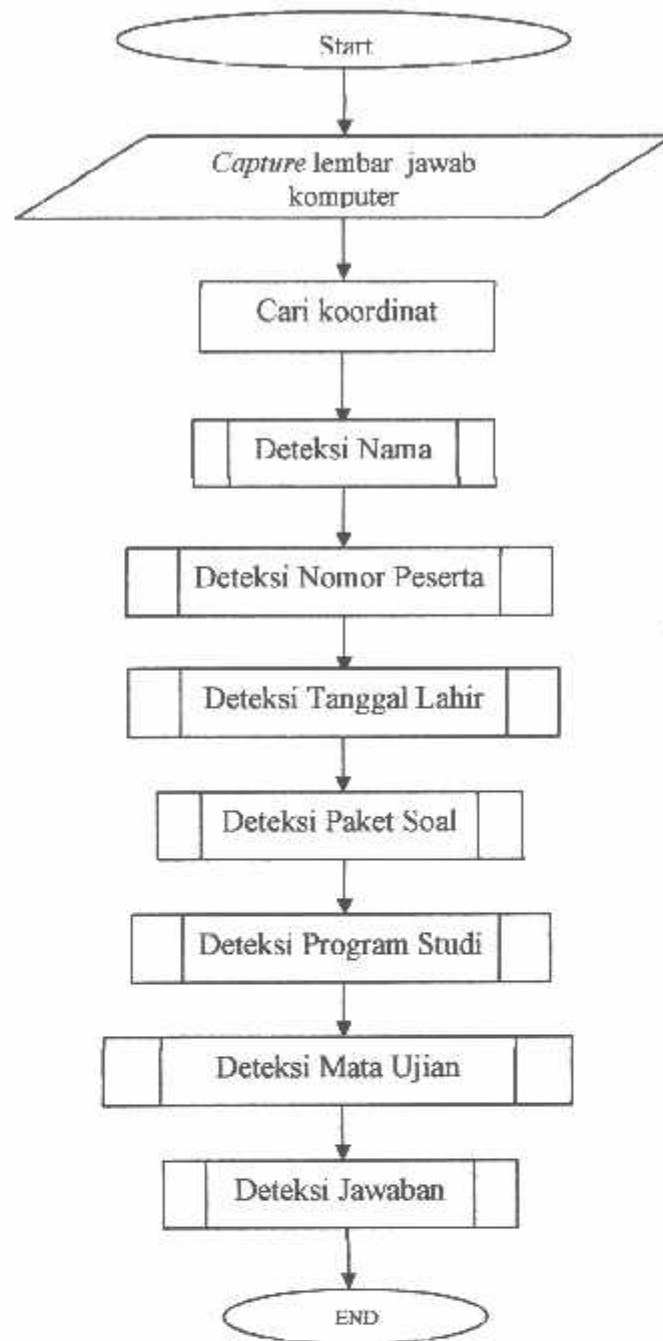
| Nama Komponen | Keterangan                        |
|---------------|-----------------------------------|
| DSPack        | Untuk mengaktifkan <i>Camera</i>  |
| AlphaControl  | Untuk Tampilan pada <i>delphi</i> |

Sedangkan komponen-komponen dari *delphi* sendiri yang digunakan dalam pembuatan aplikasi ini diantaranya adalah :

Tabel 3.2 Daftar komponen *delphi* yang digunakan dalam pembuatan aplikasi

| Nama Komponen  | Keterangan  |
|----------------|---|
| Label          | Digunakan dalam penamaan keterangan               |
| Edit           | Digunakan sebagai Inputan                         |
| BitBtn         | Digunakan sebagai Tombol bergambar                |
| TrackBar       | Digunakan dalam Zoom                              |
| DateTimePicker | Digunakan untuk pewaktuan jam dan tanggal         |
| CheckBox       | Menyatakan suatu kondisi Boolean.                 |
| RadioButton    | Digunakan sebagai pilihan menu                    |
| GroupBox       | Digunakan untuk mengelompokkan komponen           |
| Shape          | Digunakan sebagai indicator motion detection      |
| Timer          | Digunakan sebagai pewaktu countdown dan juga jam. |

Selanjutnya untuk proses pengolahan citra, diagram alirnya sebagai berikut:

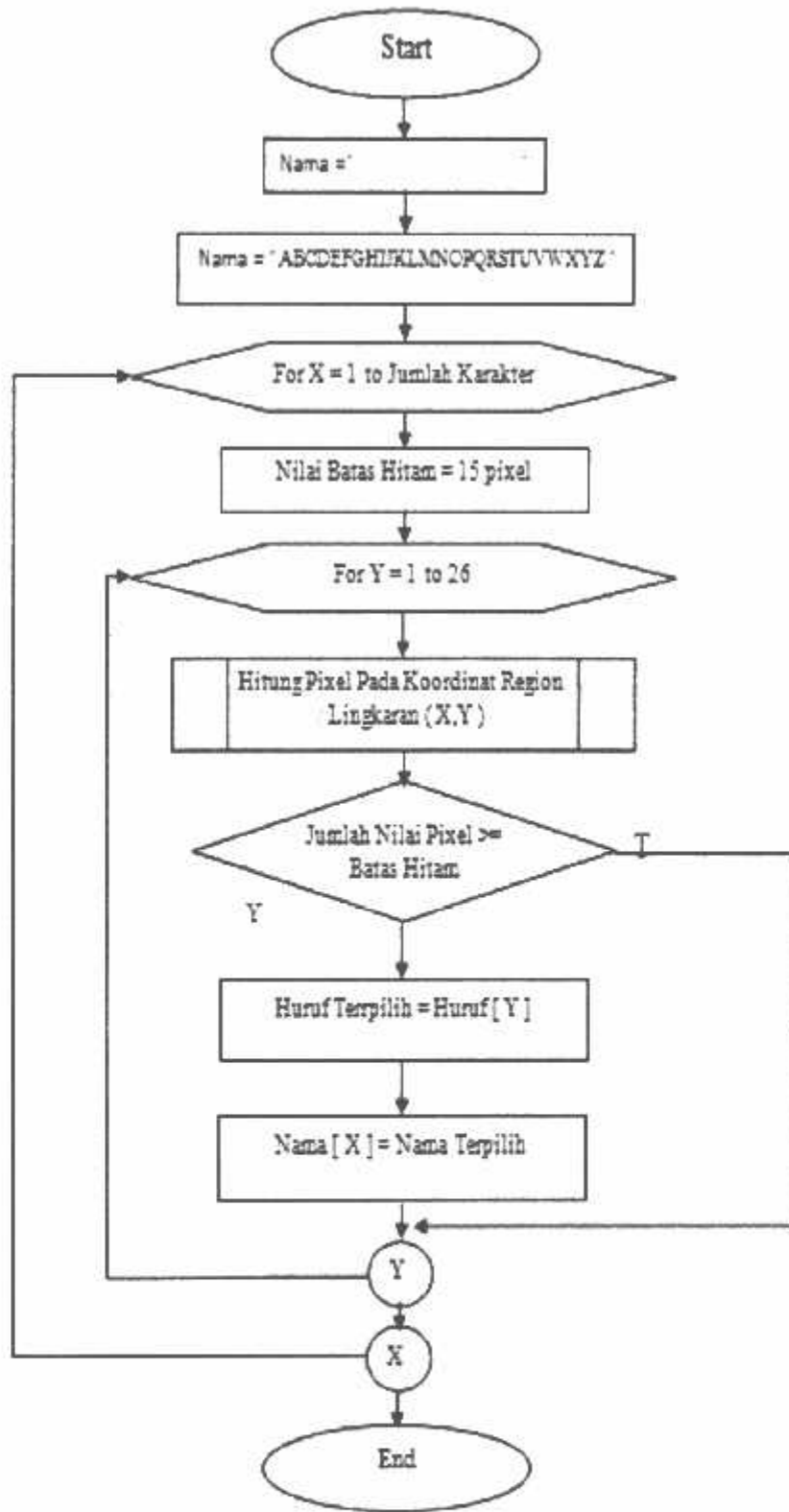


Gambar 3.7 Flowchart proses pengolahan citra

Pertama lembar jawab komputer telah di-capture oleh *webcam*. Proses selanjutnya mencari posisi koordinat yang sudah ditentukan. Kemudian sistem mendeteksi bagian – bagian yang terdapat di lembar jawab komputer antara lain nama, nomor peserta, tanggal lahir, paket soal, program studi, mata ujian, dan jawaban. Bagian – bagian lembar jawab komputer ini dideteksi dengan perulangan (*looping*) yang berbeda – beda.

- Keterangan *flowchart* Deteksi Nama.

1. Flowchart ini diawali dengan memberi inisialisasi Nama = ' '.
  2. Pada bagian nama, terdapat 20 karakter (isian nama) yang diasumsikan sebagai baris x dan 26 huruf yang terdiri dari A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z sebagai kolom y.
  3. Untuk kolom pertama, baris pertama dicari jumlah nilai *pixel* sesuai batas hitam.
  4. Jika terdapat nilai jumlah *pixel* lebih besar dari batas hitam maka sistem akan mengenali tanda hitam tersebut sesuai huruf yang terpilih( huruf A-Z).
  5. Jika tidak ditemukan maka sistem akan melanjutkan pendeteksian jumlah nilai *pixel* pada kolom pertama baris kedua, begitu seterusnya sampai baris ke dua puluh enam (26).
  6. Kemudian terjadi perulangan proses hingga baris ke dua puluh enam.
-

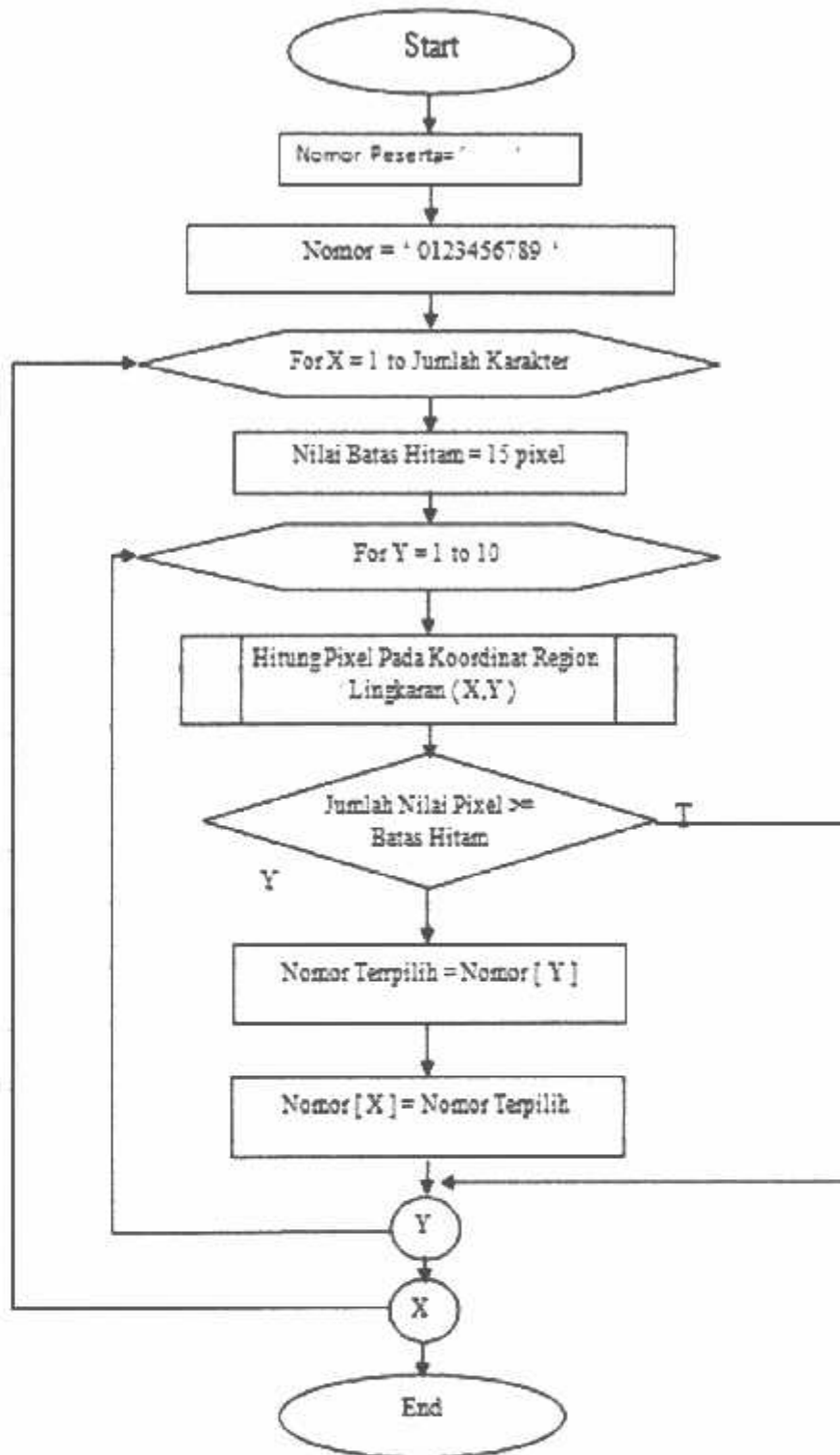


Gambar 3.8 *Flowchart* Deteksi Nama

- Keterangan *flowchart* Deteksi Nomor Peserta

1. Flowchart ini diawali dengan memberi inisialisasi Nomor Peserta = ' '.
  2. Pada bagian nomor, terdapat 9 karakter (isian nomor) yang diasumsikan sebagai baris x dan 10 nomor yang terdiri dari 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 sebagai kolom y.
  3. Untuk kolom pertama, dicari 'jumlah nilai pixel'.
  4. Nilai batas hitam = 15 *pixel* adalah jumlah nilai *pixel* minimum yang dianggap sebagai jawaban, jika jumlah nilai *pixel* tidak mencapai 15 *pixel*, maka *software* tidak menganggap sebagai jawaban.
  5. Selanjutnya pada kolom pertama baris pertama, setiap *region* nomor dihitung jumlah nilai *pixel* warna. Setelah diketahui, dibandingkan dengan nilai dengan batas hitam. Apakah jumlah nilai *pixel* kurang dari batas hitam yang telah ditentukan, jika tidak maka proses menuju ke *region* nomor kedua kolom pertama.
  6. Jika nilai nilai *pixel* kurang dari batas hitam yang telah ditentukan, jika tidak maka proses menuju ke *region* nomor kedua kolom pertama.
  7. Jika ya, maka nomor terpilih sebagai nomor pertama yang ditampilkan.
  8. Sehingga nomor peserta pada karakter pertama diisi dengan nomor yang terpilih.
  9. Kemudian terjadi perulangan proses hingga baris ke sepuluh (10) dan kolom ke sembilan (9).
-

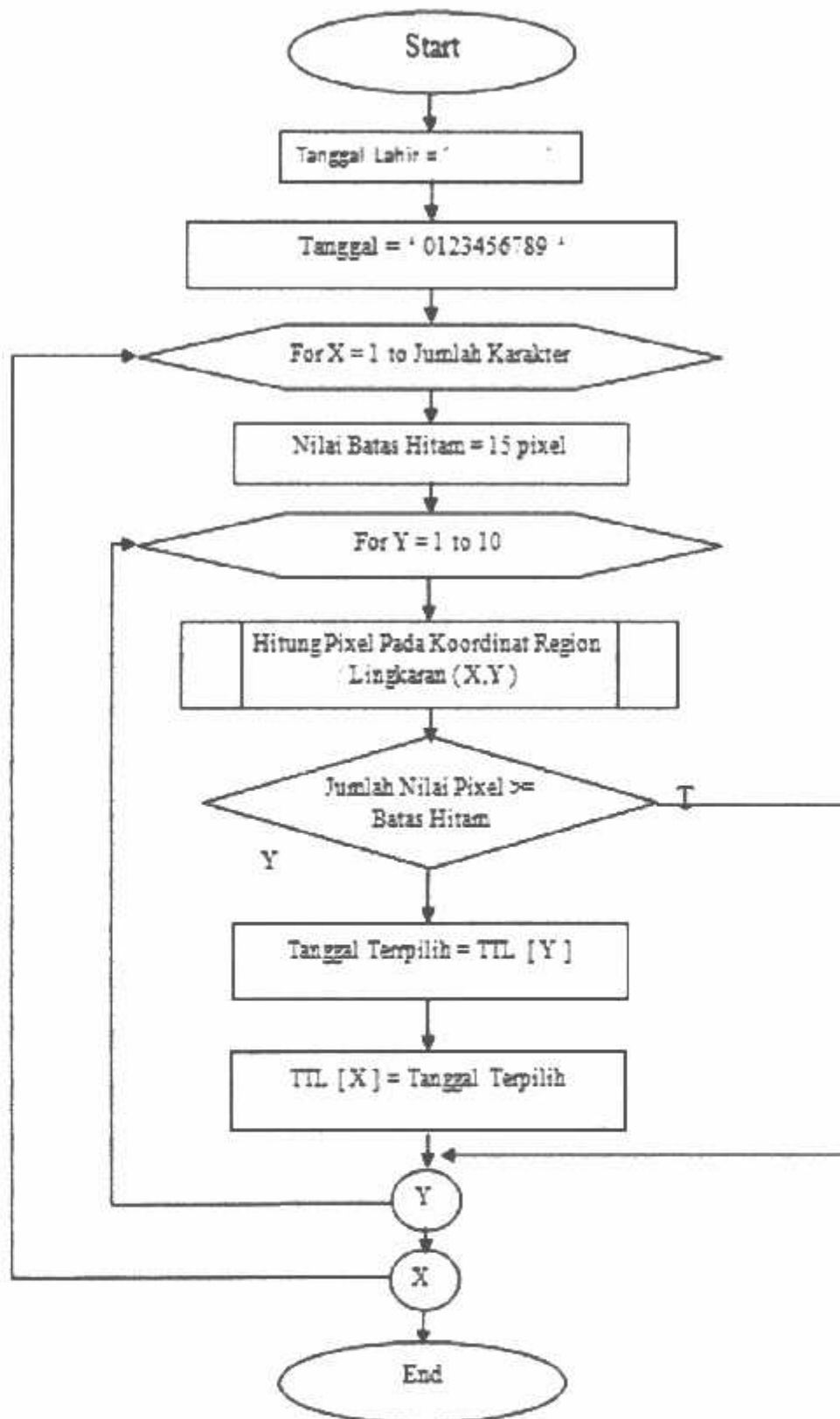




Gambar 3.9 Flowchart Deteksi Nomor Peserta

- Keterangan *flowchart* Deteksi Tanggal Lahir

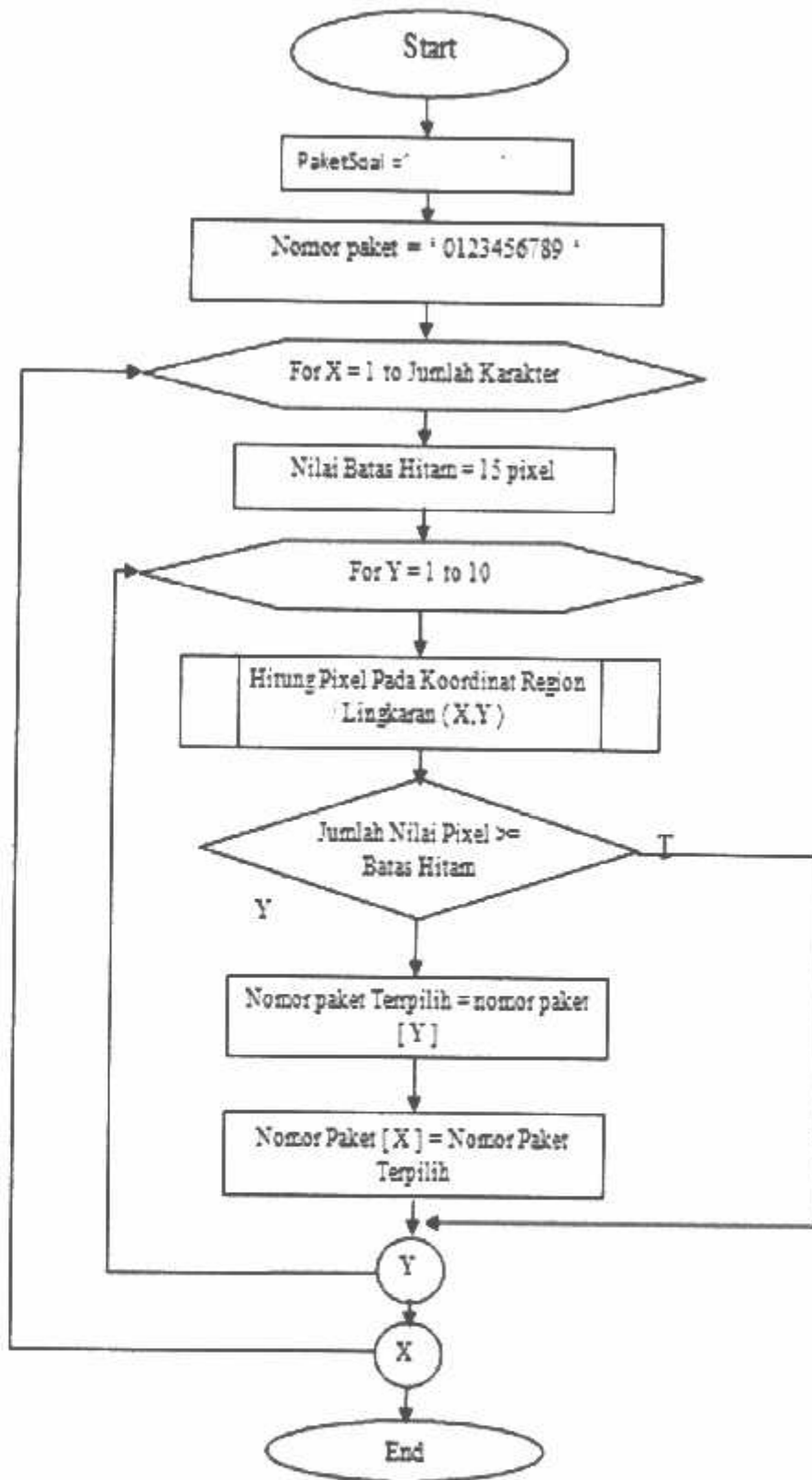
1. Flowchart ini diawali dengan memberi inisialisasi TTL = ' '.
  2. Pada bagian tanggal, terdapat 8 karakter (isian tanggal) yang diasumsikan sebagai baris x dan 10 nomor yang terdiri dari 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 sebagai kolom y.
  3. Untuk kolom pertama, dicari 'jumlah nilai *pixel*'.
  4. Nilai batas hitam = 15 *pixel* adalah jumlah nilai *pixel* minimum yang dianggap sebagai jawaban, jika jumlah nilai *pixel* tidak mencapai 15 *pixel*, maka *software* tidak menganggap sebagai jawaban.
  5. Selanjutnya pada kolom pertama baris pertama, setiap *region* tanggal dihitung jumlah nilai *pixel* warna hitam. Setelah diketahui, dibandingkan dengan nilai batas. Apakah jumlah nilai *pixel* warna hitam kurang dari nilai batas hitam, jika tidak maka proses menuju ke *region* tanggal kedua kolom pertama.
  6. Jika nilai nilai *pixel* kurang dari batas hitam yang telah ditentukan, jika tidak maka proses menuju ke *region* nomor kedua kolom pertama.
  7. Jika ya, maka nomor terpilih sebagai nomor pertama yang ditampilkan.
  8. Sehingga angka tanggal lahir pada karakter pertama diisi dengan nomor yang terpilih
  9. Kemudian terjadi perulangan proses hingga baris ke sepuluh (10) dan kolom ke delapan (8).
-



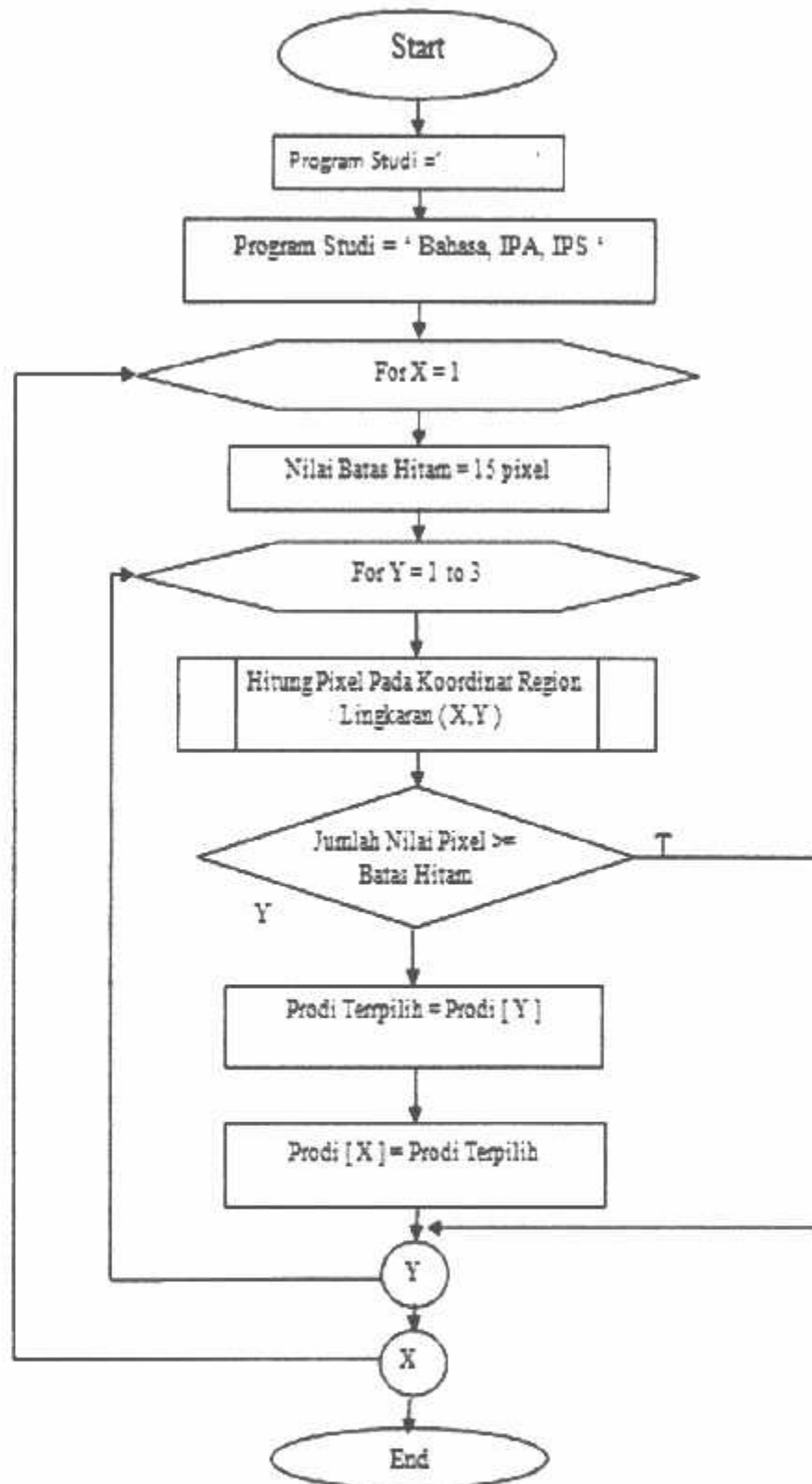
Gambar 3.10 *Flowchart* Deteksi Tanggal Lahir

- Keterangan *flowchart* Deteksi Paket Soal

1. Flowchart ini diawali dengan memberi inisialisasi PaketSoal = ' '.
  2. Pada bagian Paket Soal, terdapat 2 karakter (isian nomor) yang diasumsikan sebagai baris x dan 10 angka yang terdiri dari 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 sebagai kolom y.
  3. Untuk kolom pertama, jumlah nilai *pixel*.
  4. Nilai batas hitam = 15 *pixel* adalah jumlah nilai *pixel* minimum yang dianggap sebagai jawaban, jika jumlah nilai *pixel* tidak mencapai 15 *pixel*, maka *software* tidak menganggap sebagai jawaban.
  5. Selanjutnya pada kolom pertama baris pertama, setiap *region* paket soal dihitung jumlah nilai *pixel* warna hitam. Setelah diketahui, dibandingkan dengan nilai batas hitam. Apakah jumlah nilai *pixel* warna hitam kurang dari nilai batas hitam, jika tidak maka proses menuju ke *region* paket soal kedua kolom pertama.
  6. Jika nilai nilai *pixel* kurang dari batas hitam yang telah ditentukan, jika tidak maka proses menuju ke *region* nomor kedua kolom pertama.
  7. Jika ya, maka nomor terpilih sebagai nomor pertama yang ditampilkan.
  8. Sehingga angka paket soal pada karakter pertama diisi dengan nomor yang terpilih.
  9. Kemudian terjadi perulangan proses baris ke sepuluh (10) dan kolom ke dua (2).
-



Gambar 3.11 Flowchart Deteksi Paket Soal



Gambar 3.12 *Flowchart* Deteksi Program Studi

- Keterangan *flowchart* Deteksi Mata Ujian

1. Flowchart ini diawali dengan memberi inisialisasi Mata Ujian – ‘ ‘.
  2. Pada bagian Mata Ujian, hanya x sebagai kolom isian yang harus dipilih salah satu. Karena pada kolom mata ujian hanya terdapat 9 pilihan yang terdiri dari Bahasa Indonesia, Bahasa Inggris, Matematika, Fisika, Kimia, Biologi, Ekonomi, Sosiologi, Geografi.
  3. Untuk kolom pertama, dicari jumlah nilai *pixel*.
  4. Selanjutnya pada kolom pertama, setiap *region* mata ujian dihitung nilai *pixel* warna hitam. Setelah diketahui, dibandingkan dengan nilai batas hitam. Apakah rata – rata intensitas warna kurang dari nilai batas hitam, jika tidak maka proses menuju ke *region* mata ujian kedua kolom pertama.
  5. Jika nilai *pixel* warna hitam kurang dari nilai batas hitam. Jika tidak, maka proses menuju ke *region* mata ujian kedua.
  6. Jika ya, maka mata ujian yang terpilih sebagai *region* sama dengan mata ujian pertama.
  7. Sehingga Mata ujian pada karakter pertama diisi dengan mata ujian yang terpilih
  8. Kemudian terjadi perulangan proses hingga kolom ke sembilan (9).
-

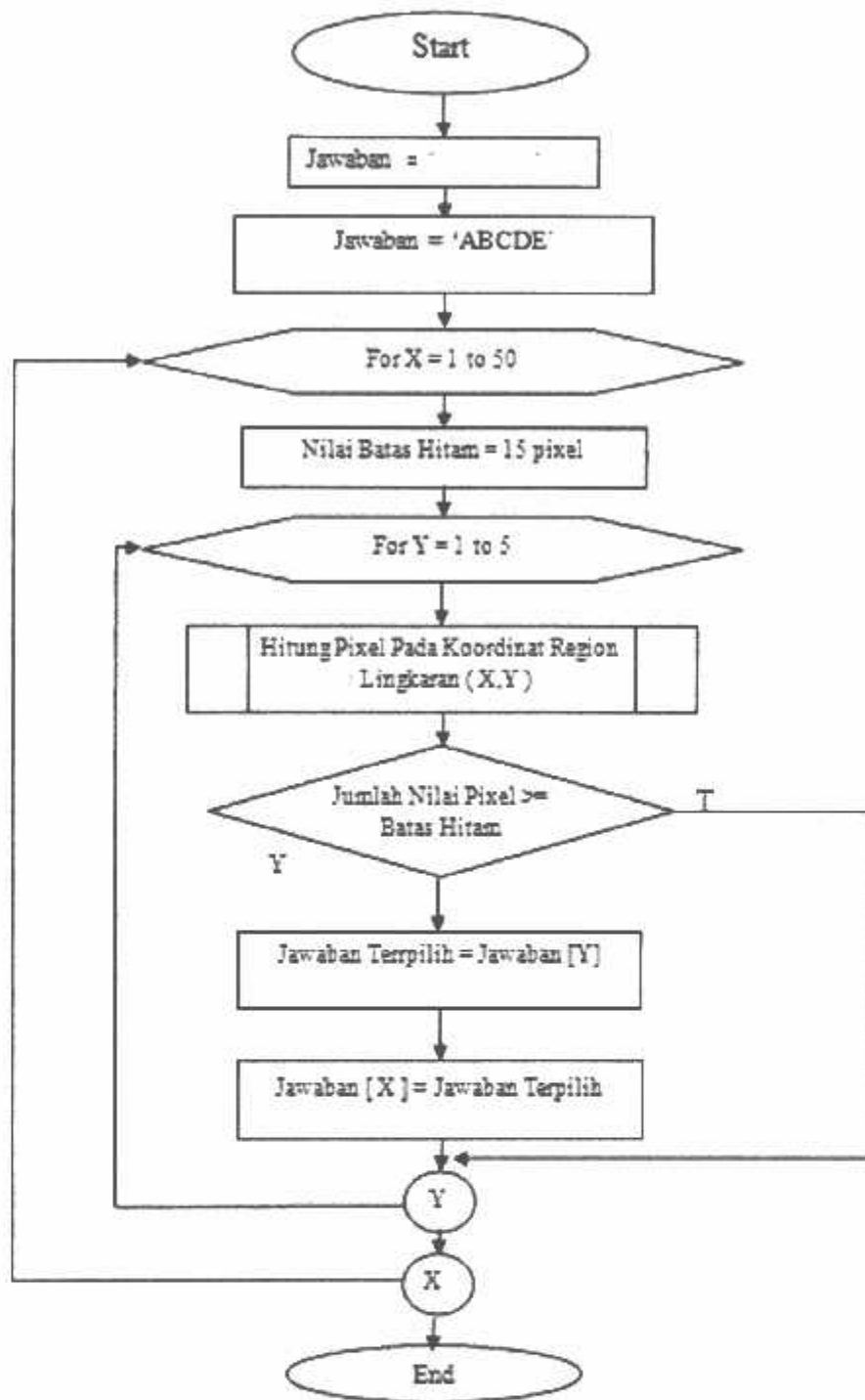
- Keterangan *flowchart* Deteksi Jawaban

1. Flowchart diawali dengan memberi inisialisasi Jawaban = ' '.
2. Pada bagian Jawaban, terdapat 50 karakter (isian jawaban) yang diasumsikan sebagai baris y dan 5 opsi yang terdiri dari A, B, C, D, E sebagai kolom x.
3. Untuk baris pertama, dicari jumlah nilai *pixel* pada *region* pertama.



4. Selanjutnya pada baris pertama kolom pertama, pada *region* pertama di hitung jumlah nilai *pixel*. Setelah diketahui, dibandingkan batas hitam. Jika nilai *pixel* pada *region* pertama kurang dari nilai batas hitam, maka proses menuju ke *region* opsi kedua kolom pertama.
5. Jika jumlah nilai *pixel* lebih besar atau sama dengan nilai batas hitam, maka proses akan memilih huruf terpilih tersebut dan melanjutkan ke *region* selanjutnya.
6. Jika pada *region* selanjutnya pada baris yang sama terdapat jumlah nilai *pixel* lebih besar atau sama dengan nilai batas hitam maka sistem akan memilih karakter tersebut sebagai jawaban ( jawaban yang terdeteksi bias lebih dari satu jawaban ).
7. Kemudian terjadi perulangan proses hingga kolom ke lima (5) dan baris ke lima puluh (50).

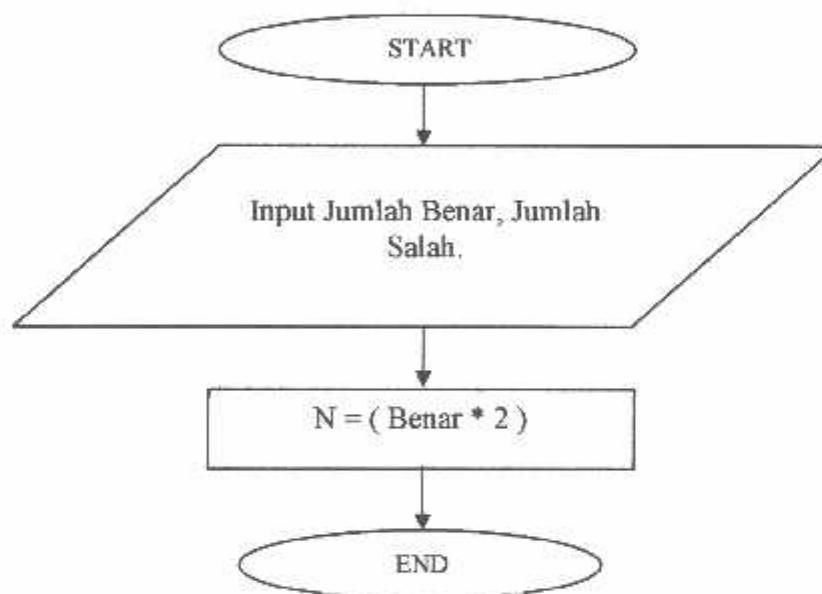




Gambar 3.14 *Flowchart* Deteksi Jawaban

Proses pemberian nilai pada sistem ini dilakukan dengan cara mencocokkan jawaban yang sudah terdeteksi dengan kunci jawaban yang tersedia, sistem kerja pemberian nilai sebagai berikut:

1. Jawaban terdiri dari dua kategori (S), benar (B), sebelum proses pendeteksian jawaban dilakukan variabel bernilai  $S = 0, B = 0$ .
2. kemudian sistem akan membaca soal dari urutan pertama sampai ke jumlah soal.
3. Jika jawaban pertama sama dengan (cocok) kunci jawaban maka nilai 'benar' sebelumnya ditambah satu, jika tidak sama maka nilai 'salah' sebelumnya ditambah satu, dan jika tidak terdapat jawaban maka variable nilai benar dan salah tetap. Berlaku seterusnya hingga jumlah soal.
4. Jika telah memenuhi jumlah soal, program akan menampilkan jumlah masing-masing benar, salah.
5. Kemudian program akan mengolah nilai sesuai jumlah jawaban benar dan jumlah jawaban salah dengan rumus penghitungan nilai = ( benar \* 2 ) = nilai.

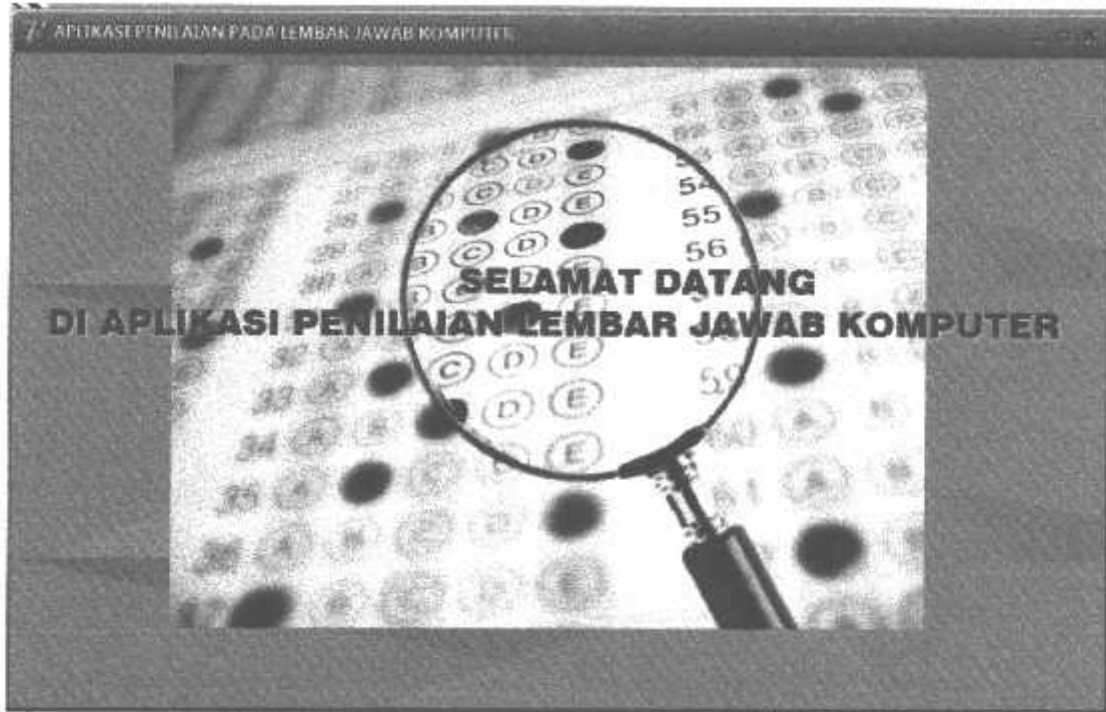


Gambar 3.15 *Flowchart* Olah Nilai

Keterangan *flowchart* sebagai berikut:

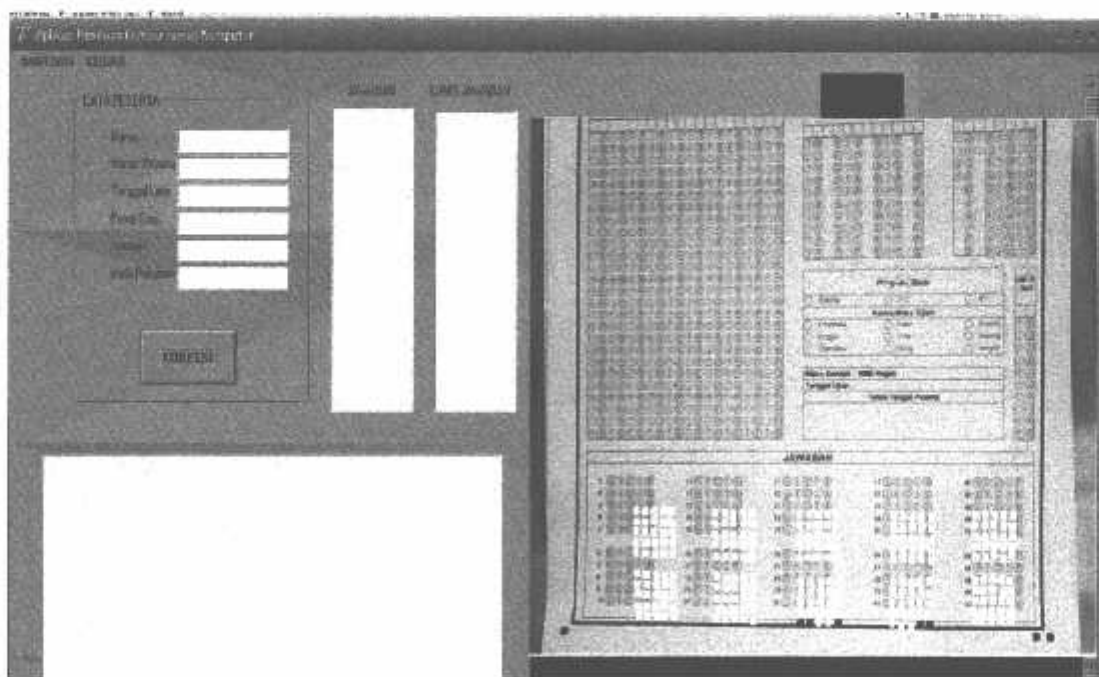
1. Inputan berupa Jumlah Benar, Jumlah Salah, dan Nilai
2. Benar bernilai 1, salah bernilai 1. Sehingga perhitungan nilai total dirumuskan dengan ( Jumlah Benar \* 2 ) = nilai.

### 3.4 Desain Antar Muka



Gambar 3.16 Desain Tampilan Awal Program

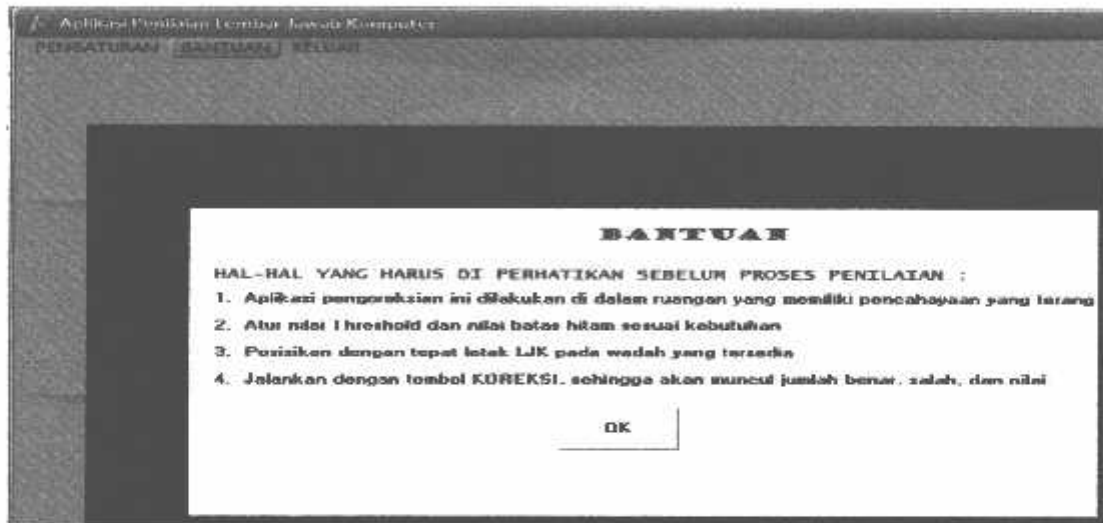
Gambar diatas merupakan tampilan awal dari program dimana secara otomatis sistem akan masuk pada menu utama program.



Gambar 3.17 Desain Menu Utama

Gambar diatas merupakan tampilan awal dari program dimana terdapat beberapa menu diantaranya menu pertolongan, bantuan, dan keluar. Pada sistem ini hasil *capture* dari lembar jawab komputer akan di tampilkan ada *form image* yang selanjutnya akan diolah oleh sistem sehingga dapat menghasilkan nilai sesuai jumlah jawaban salah dan jawaban benar.

Selanjutnya terlihat di tampilan awal, terdapat menu “Bantuan”, yang tampilannya sebagai berikut:



Gambar 3.19 Desain Menu Bantuan

Pada menu bantuan ditampilkan langkah-langkah sebelum dilakukan proses penilaian pada lembar jawab komputer seperti pada gambar 3.19.

## BAB IV

### IMPLEMENTASI DAN PEMBAHASAN

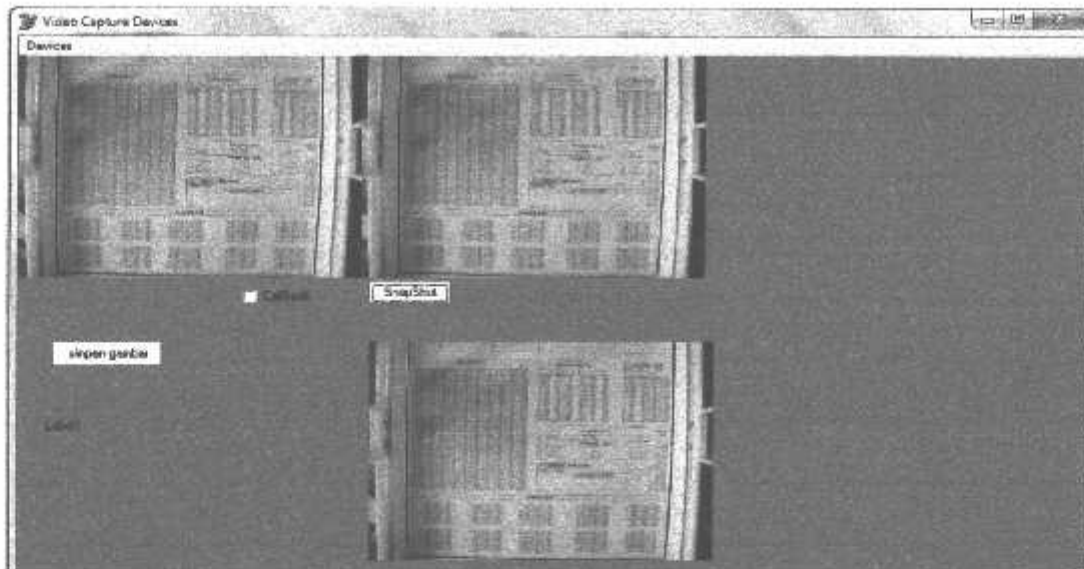
Sistem aplikasi penilaian lembar jawab komputer memanfaatkan *webcam* berbasis *dhelphi* diperlukan pengujian agar sesuai dengan perancangan. Pengujian ini terdiri dari pengujian perangkat keras (*hardware*) dan pengujian perangkat lunak (*software*). Adapun pengujian terhadap *hardware* meliputi beberapa komponen rangkaian, yaitu: pengujian *webcam* sebagai alat untuk *mengcapture* obyek, dan pengujian *software*.

#### 4.1 Pengujian Hardware

Pengujian ini dilakukan terhadap perangkat keras pada masing-masing komponen. Pengujian ini bertujuan untuk mengetahui apakah setiap komponen dapat bekerja dengan baik. Jika rangkaian setiap komponen dapat bekerja dengan baik, maka akan mendukung bekerjanya sistem secara keseluruhan.

##### 4.1.1 Pengujian Webcam sebagai alat *mengcapture* obyek

Pada proses pengujian alat, tahap awal yang dilakukan yaitu pengujian *webcam* sebagai sensor warna. Dilakukan pengujian ini untuk mengetahui apakah *webcam* dapat berfungsi, serta dapat terhubung dengan PC (*Personal Computer*). Pengujian *webcam* dilakukan dengan menghubungkan *webcam* dengan PC (*Personal Computer*) melalui port USB. Selanjutnya mengaktifkan USB *video device* melalui program explorer. Seperti pada gambar 4.1.



Gambar 4.1 Pengujian Menggunakan *Webcam A4TECH* Dengan Resolusi 1,3 Megapixel

#### Hasil Pengujian:

Pada pengujian ini *webcam* berfungsi dengan baik dalam menangkap atau *capture* obyek (lembar jawab komputer) sesuai yang dikehendaki.

#### 4.2 Pengujian *Software*

Pengujian ini dilakukan terhadap perangkat lunak (*software*) dalam hal ini menggunakan bahasa pemrograman *Delphi 7*. Pengujian ini bertujuan untuk mengetahui apakah *software* dapat bekerja sesuai dengan yang diinginkan atau tidak.

Proses pengambilan dan pengujian data dilakukan di dalam ruangan yang memiliki cahaya cukup terang. Sehingga pengaturan *threshold* pun dapat dilakukan dengan mudah. Langkah-langkah yang harus dilakukan antara lain:

1. *Listing code* di bawah ini untuk mengaktifkan *webcam* agar dapat menangkap lembar jawaban. *Listing program* untuk menampilkan gambar:

```
SampleGrabber.GetBitmap (IFOTO.Picture.Bitmap);
```

2. *Listing code* di bawah ini untuk proses penyaringan warna merah (warna form lembar jawab) agar terbaca sebagai warna putih sehingga tidak dianggap sebagai jawaban.

```
begin
  if ((ImWarna.Picture.Bitmap.Canvas.Pixels[xx,yy] mod
    $100)< th) then
  BEGIN
```

3. *Listing code* di bawah ini untuk menentukan nilai batas hitam

```
procedure TForm1.TrackBar2Change(Sender: TObject);
begin
  EdToleransi.Text:=inttostr(TrackBar2.Position);
  batas:=TrackBar2.Position;
```

4. *Listing code* di bawah ini untuk menentukan nilai *threshold*.

```
jmlpixel:=0;
for xx:=(X-3) to (X+3) do
  for yy:=(y-3) to (y+3) do
  begin
    if ((IFOTO.Picture.Bitmap.Canvas.Pixels[xx,yy] mod
    $100)< threshold) then
    BEGIN
      jmlpixel:=jmlpixel+1;
    END
  end;
Ejmlpixel.Text:=inttostr(jmlpixel);
Erd ;
```

5. *listing code* untuk memulai proses koreksi atau penilaian terhadap lembar jawab yang sudah di tangkap atau di capture oleh *webcam*.

```

procedure TForm1.btKoreksiClick(Sender: TObject);
var i, benar, salah, kosong, nilai: Integer;
begin
  bnama.Click;
  bjwb.Click;
  btTL.Click;
  bNOMOR.Click;
  bPAKET.Click;
  BJURUSAN.Click;

```

6. Listing code dibawah ini untuk menentukan nilai sesuai jumlah jawaban benar dan jawaban salah.

```

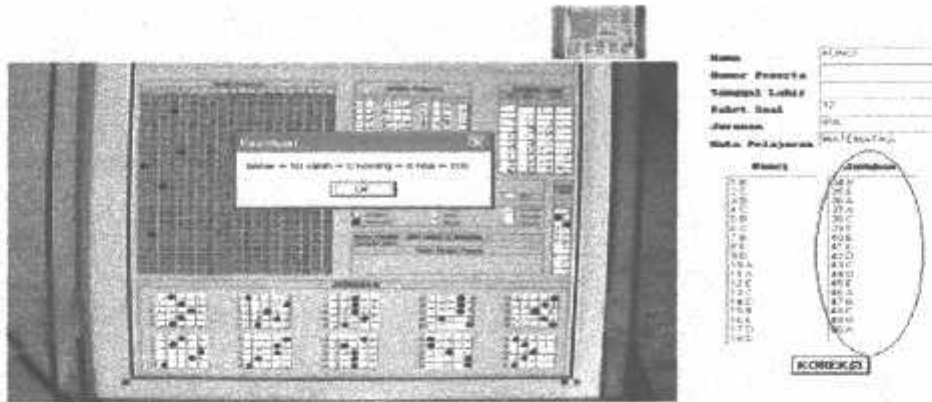
MKunci.Lines.LoadFromFile (EdPakso.Text+EdJur.Text+EdMapel.T
ext+'.csv');

  benar:=0;
  salah:=0;
  kosong:=0;
  for i:=0 to 49 do
    if
MKunci.Lines.Strings[i]=MJAWABAN.Lines.Strings[i]
    then benar:=benar+1
    else
    begin
      if
MJAWABAN.Lines.Strings[i]=(IntToStr(i+1)+'.') then
      kosong:=kosong+1
      else
      salah:=salah+1;
    end;
  nilai:=(benar*2);
  ShowMessage ('benar = '+IntToStr(benar)+' '+'salah
='+IntToStr(salah)+ ' '+'nilai = '+IntToStr(nilai));
  end

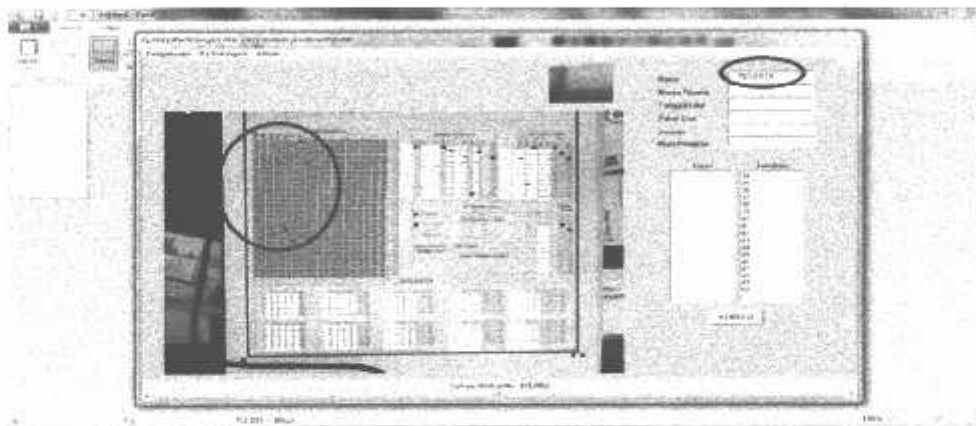
```



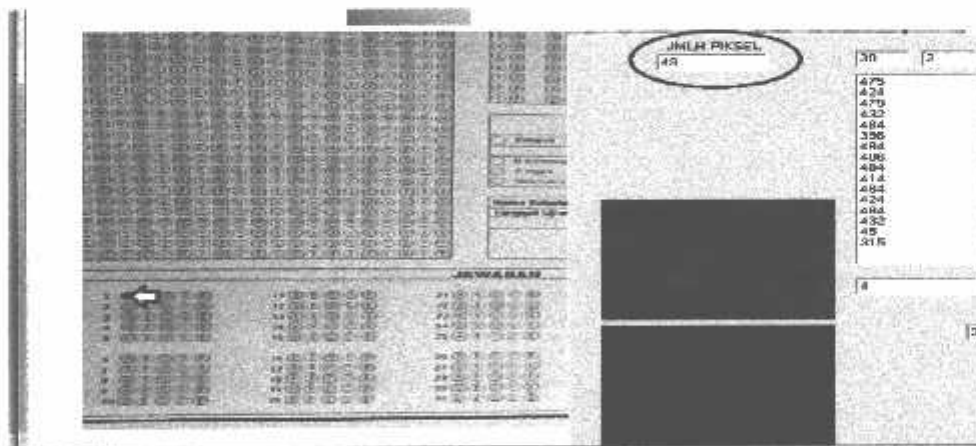
Pengujian *software* dalam mendeteksi jawaban jawaban:



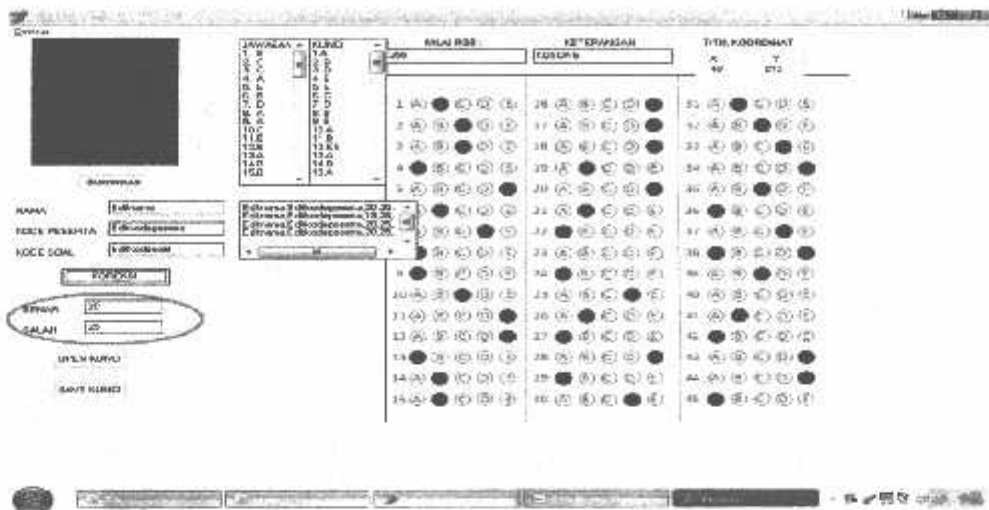
Gambar 4.2 Pengujian *Software* Mendeteksi Jawaban



Gambar 4.3 Pengujian *Software* Mendeteksi Nama









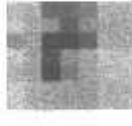




Gambar 4.4 Pengujian *Software* Mendeteksi Jumlah *Pixel* Pada Bulatan Jawaban ( jika di isi penuh )

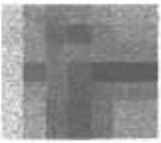





Gambar 4.5 Pengujian *Software* Mendeteksi Jumlah Benar Dan Salah Setelah Dicocokkan Dengan Kunci Jawaban

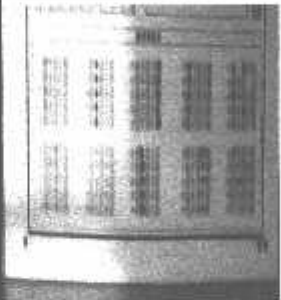
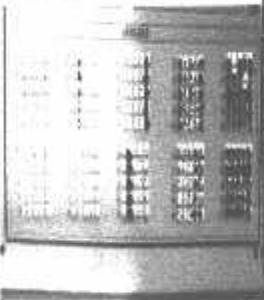
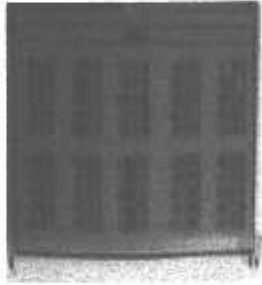

Berikut adalah tabel data yang didapat dari hasil pengujian dengan nilai *threshold* antara 0 – 255. Karena jika diuji satu persatu dari 0 hingga 255, akan menghasilkan data terlalu banyak. Maka nilai *threshold* yang diambil adalah 75 dengan batas hitam sebesar 15 *pixel*. Pengujian gambar *region* opsi pada lembar jawaban komputer dilakukan dengan perbesaran 7X7 *pixel*, jadi jumlah keseluruhan *pixel* untuk satu *region merging* berjumlah 49 *pixel*. Jarak / tinggi *webcam* dengan lembar jawab ditentukan sebesar 30 cm dengan waktu pengambilan siang hari dalam ruangan.

Tabel 4.1 Pengujian Mendeteksi Jawaban Berdasarkan Jumlah Nilai *Pixel* Pada Warna Jawaban Dalam Berbagai Bentuk Pengisian

| Bentuk Pengisian   | Warna Jawaban      | Gambar Asli   | Gambar setelah pengaruh threshold   | Jumlah Pikel    | Keterangan    |
|--|--------------------|---|---|-----------------|---------------|
| Tipis memakai pensil 2B                                  | Hitam ke abu-abuan |    |    | 23 <i>pixel</i> | Jawaban       |
| Tebal penuh memakai spidol                               | Hitam              |    |    | 40 <i>pixel</i> | Jawaban       |
| Disilang tebal memakai pensil 2B                         | Hitam              |  |  | 17 <i>pixel</i> | Jawaban       |
| Hanya di bagian tengah jawaban yang dihitamkan           | Hitam              |  |  | 9 <i>pixel</i>  | Bukan jawaban |
| Jawaban salah kemudian dihapus menggunakan <i>type-x</i> | Putih              |  |   | 0 <i>pixel</i>  | Bukan jawaban |
| Jawaban diisi dengan pena biru                           | Biru               |  |  | 17 <i>pixel</i> | Jawaban       |

|   |       |   |   |          |         |
|---|-------|---|---|----------|---------|
| Jawaban diisi dengan pena warna hijau   | Hijau |  |  | 15 pixel | Jawaban |
| Jawaban diisi dengan spidol warna hitam | Hitam |  |  | 39 pixel | Jawaban |

Tabel 4.2 Pengujian Jika Terdapat Bayangan Pada Lembar Jawaban

| Letak Bayangan                              | Gambar Sebelum Proses Penilaian   | Gambar Sesudah Proses Penilaian   | Keterangan  |
|---|---|---|---|
| Setengah Bagian Kanan LJK Tertutup Bayangan |   |   | Pada bagian yang tertutup bayangan terdeteksi jawaban "A B C D E"                     |
| Seluruh Bagian Kanan LJK Tertutup Bayangan  |  |  | Software menganggap semua jawaban di isi, terdeteksi jawaban "ABCDE" pada semua nomor |

Dari hasil pengujian faktor bayangan, maka pada proses penilaian lembar jawab komputer ini, faktor pencahayaan sangat berpengaruh terhadap keakuratan sistem dalam mendeteksi jawaban. Adapun hasil pengujian keseluruhan sistem seperti pada table 4.3 dan 4.4 :

Tabel 4.3 Pengujian Bentuk Pengisian Jawaban

| Percobaan   | Alat pengisian jawaban | Bentuk pengisian jawaban | Keterangan (Terdeteksi sebagai) |
|-------------|------------------------|--------------------------|---------------------------------|
| Percobaan 1 | PENSIL                 | Bulatan                  | JAWABAN                         |
|             |                        | Silang                   | JAWABAN                         |
|             |                        | Cawang                   | BUKAN JAWABAN                   |
| Percobaan 2 | PENSIL                 | Bulatan                  | JAWABAN                         |
|             |                        | Silang                   | JAWABAN                         |
|             |                        | Cawang                   | BUKAN JAWABAN                   |
| Percobaan 3 | PENSIL                 | Bulatan                  | JAWABAN                         |
|             |                        | Silang                   | BUKAN JAWABAN                   |
|             |                        | Cawang                   | BUKAN JAWABAN                   |
| Percobaan 4 | PENA HITAM             | Bulatan                  | JAWABAN                         |
|             |                        | Silang                   | JAWABAN                         |
|             |                        | Cawang                   | BUKAN JAWABAN                   |
| Percobaan 5 | PENA HITAM             | Bulatan                  | JAWABAN                         |
|             |                        | Silang                   | BUKAN JAWABAN                   |
|             |                        | Cawang                   | JAWABAN                         |
| Percobaan 6 | PENA HITAM             | Bulatan                  | JAWABAN                         |
|             |                        | Silang                   | JAWABAN                         |
|             |                        | Cawang                   | BUKAN JAWABAN                   |

|             |              |         |         |
|-------------|--------------|---------|---------|
| Percobaan 7 | SPIDOL HITAM | Bulatan | JAWABAN |
|             |              | Silang  | JAWABAN |
|             |              | Cawang  | JAWABAN |
| Percobaan 8 | SPIDOL HITAM | Bulatan | JAWABAN |
|             |              | Silang  | JAWABAN |
|             |              | Cawang  | JAWABAN |
| Percobaan 9 | SPIDOL HITAM | Bulatan | JAWABAN |
|             |              | Silang  | JAWABAN |
|             |              | Cawang  | JAWABAN |

---

Tabel 4.4 Pengujian Keseluruhan Sistem

| Percobaan       | Data Peserta       |                    |                    |                    |                    |                    |
|-----------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
|                 | Nama               | Nomor              | TTL                | Prodi              | Mata Ujian         | Jawaban            |
| Pertama<br>1    | Terdeteksi (salah) | Terdeteksi (salah) | Terdeteksi (salah) | Tidak Terdeteksi   | Terdeteksi (benar) | Terdeteksi (salah) |
| Kedua<br>2      | Terdeteksi (salah) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (salah) |
| Ketiga<br>3     | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) |
| Keempat<br>4    | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) |
| Kelima<br>5     | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) |
| Keenam<br>6     | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) |
| Ketujuh<br>7    | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) |
| Kedelapan<br>8  | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) |
| Kesembilan<br>9 | Terdeteksi (salah) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (benar) | Terdeteksi (salah) |

### 4.3 Pembahasan

Dari hasil pengujian yang telah dilakukan ada beberapa faktor yang dapat mempengaruhi proses penilaian lembar jawab komputer, faktor utama yang sangat berpengaruh dalam pendeteksian jawaban ini adalah pencahayaan, dimana setiap ruangan yang digunakan pada saat penerapan aplikasi ini memiliki intensitas cahaya berbeda-beda, sehingga dalam *software* terdapat menu pengaturan *threshold* yang digunakan untuk kalibrasi awal pengolahan citra sesuai dengan keadaan pencahayaan.

Dari hasil pengujian pencarian nilai *threshold* yang sesuai diperoleh nilai batas hitam sebesar 15 *pixel* dengan nilai *threshold* 75. Batas hitam ini didapat dari metode *region merging* yaitu menghitung banyaknya jumlah *pixel* yang berasal dari citra *grayscale* (filter warna merah), yang kurang dari nilai *threshold*. Sedangkan penentuan nilai *threshold* berdasarkan hasil pengujian lembar jawaban yang pengisian jawaban tersebut terdeteksi semua dengan benar dan sesuai. Kemudian di dapatkan juga hasil pengujian dari berbagai macam bentuk pengisian pada lembar jawab komputer.

Selain itu, faktor yang mempengaruhi proses penilaian lembar jawab komputer ini adalah terdapat adanya bayangan pada saat *webcam* menangkap gambar lembar jawab. Hal ini diperkuat oleh adanya pengujian bayangan yang terjadi baik terdapat bayangan di sebagian lembar jawab komputer maupun terdapat bayangan yang menutupi seluruh bagian dari lembar jawaban komputer. Karena jika terdapat bayangan sekecil apapun, maka *software* tidak akan tepat dalam mendeteksi opsi - opsi jawaban pada lembar jawab komputer. Dari hasil penelitian yang diperoleh ini, aplikasi penilaian pada lembar jawab komputer ini nantinya akan mempermudah pendeteksian dalam intensitas banyak.

---



## BAB V

### KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian dan pembahasan sistem kerja alat Penilaian Lembar Jawab Komputer Menggunakan *Webcam* Berbasis *Borland Delphi* maka dapat diperoleh kesimpulan dan saran sebagai berikut :

#### 5.1 Kesimpulan

Dari penelitian dan pengujian aplikasi ini, dapat disimpulkan

1. Keakuratan perangkat lunak dalam mendeteksi jawaban berjalan dengan baik sesuai yang diharapkan (7 data dapat dideteksi dengan benar dari 10 pengujian).
2. Faktor pencahayaan sangat berpengaruh pada keakuratan sistem.
3. Bentuk pengisian bulatan menggunakan spidol memiliki keakuratan yang paling baik dari beberapa bentuk pengisian sesuai dengan pengujian.
4. Aplikasi ini mampu mendeteksi pengisian lembar jawab menggunakan alat tulis pensil 2B, pena hitam, pena biru, dan spidol warna hitam.
5. Aplikasi ini dapat mencocokkan jawaban peserta dengan kunci jawaban sehingga menghasilkan output berupa nilai.

#### 5.2 Saran

Adapun beberapa saran yang dapat diberikan adalah :

1. Desain tempat lembar jawab komputer yang lebih baik, sehingga meminimalisir pergeseran lembar jawab komputer maupun pergeseran *webcam*.
2. Desain tempat lembar jawab komputer yang lebih baik, agar pencahayaan ruangan tidak menjadi faktor keakuratan system dalam membaca tanda hitam.
3. Menggunakan *webcam* yang mempunyai resolusi lebih besar agar gambar yang di tangkap bisa lebih jelas.



PT. BNI (PERSERO) MALANG  
BANK NIAGA MALANG

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG  
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI  
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN  
PROGRAM PASCASARJANA MAGISTER TEKNIK

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145  
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

**BERITA ACARA UJIAN SKRIPSI  
FAKULTAS TEKNOLOGI INDUSTRI**

NAMA : CAHYO WIDYANTO  
NIM : 10.12.912  
JURUSAN : Teknik Elektro S-1  
KONSENTRASI : Teknik Komputer  
MASA BIMBINGAN: SEMESTER GENAP 2011/2012  
JUDUL : **RANCANG BANGUN APLIKASI PENILAIAN PADA  
LEMBAR JAWAB KOMPUTER MENGGUNAKAN  
WEBCAM BERBASIS BORLAND DELPHI 7.0**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Sabtu  
Tanggal : 4 Agustus 2012  
Dengan Nilai : 84 (A) *0*

**PANITIA UJIAN SKRIPSI**

**Ketua Majelis Penguji**



**Ir. Yusuf Ismail Nakhoda, MT**  
NIP.Y.1018800189

**Sekretaris Majelis Penguji**



**Dr. Eng. Arquanto S, ST, MT**  
NIP.P.1030800417

**ANGGOTA PENGUJI**

**Dosen Penguji I**



**Bambang Prio Hartono, ST, MT**  
NIP.Y.1028400082

**Dosen Penguji II**



**Bima Aulia Firmandani, ST**  
1121



**FORMULIR PERBAIKAN SKRIPSI**

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : CAHYO WIDYANTO  
 NIM : 10.12.912  
 JURUSAN : Teknik Elektro S-1  
 KONSENTRASI : Teknik Komputer  
 MASA BIMBINGAN: SEMESTER GENAP 2011/2012  
 JUDUL : RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR JAWAB KOMPUTER MENGGUNAKAN WEBCAM BERBASIS BORLAND DELPHI 7.0

| Tanggal                      | Uraian  | Paraf |
|------------------------------|---|-------|
| Penguji I<br>04 - 08 - 2012  | - Abstrak<br>- Kesimpulan   |       |
| Penguji II<br>04 - 08 - 2012 | - Pengujian ditambahkan<br>- Pensil, Bolpoint, Spidol<br>- Silang, Bulatan, Cawang<br>- Kesimpulan menyesuaikan pengujian |       |

**Disetujui,**

**Dosen Penguji I**

**Bambang Prio Hartono, ST, MT**  
 NIP.Y.1028400082

**Dosen Penguji II**

**Bima Aulia Firmandani, ST**  
 1121

**Mengetahui,**

**Dosen Pembimbing I**

**Dr. Eng. Arwanto S, ST, MT**  
 NIP.P.1030800417

**Dosen Pembimbing II**

**Ir. Yusuf Ismail Nakhoda, MT**  
 NIP.Y.1018800189



## FORMULIR BIMBINGAN SKRIPSI

Nim : 10129212  
Nama : Cahyo Widyanto  
Masa Bimbingan : Semester Genap 2011-2012  
Judul : Rancang Bangun Aplikasi Penilaian Pada Lembar Jawab Komputer Menggunakan Webcam Berbasis Borland Delphi.

| Tanggal    | Keterangan   | Paraf |
|------------|--|-------|
| 04-06-2012 | Konsultasi BAB I dan BAB II                        |       |
| 08-06-2012 | Konsultasi Cara Kerja Aplikasi                     |       |
| 13-06-2012 | Konsultasi BAB III                                 |       |
| 16-06-2012 | Konsultasi BAB IV dan BAB V                        |       |
| 20-06-2012 | Perubahan jarak webcam dengan LJK                  |       |
| 25-06-2012 | Uji Coba Aplikasi Pada Software Secara Keseluruhan |       |
| 02-07-2012 | Perbaikan Format Laporan                           |       |
| 10-07-2012 | Penyempurnaan Aplikasi                             |       |

Malang, 19-09-2012  
Dosen Pembimbing I

**Dr. Eng. Aryuanto S, ST, MT**  
NIP.P.1030800417



**PROGRAM STUDI TEKNIK ELEKTRO S-1**

**FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG**

Kampus II : Jl. Raya Karanglo Km. 2 Telp. (0341) 417636 Malang

**FORMULIR BIMBINGAN SKRIPSI**

Nim : 10129212  
Nama : Cahyo Widyanto  
Masa Bimbingan : Semester Genap 2011-2012  
Judul : Rancang Bangun Aplikasi Penilaian Pada Lembar Jawab  
Komputer Menggunakan Webcam Berbasis Borland Delphi.

| Tanggal    | Keterangan   | Paraf |
|------------|--|-------|
| 04-06-2012 | Konsultasi BAB I dan BAB II                        |       |
| 08-06-2012 | Konsultasi Cara Kerja Aplikasi                     |       |
| 13-06-2012 | Konsultasi BAB III                                 |       |
| 16-06-2012 | Konsultasi BAB IV dan BAB V                        |       |
| 20-06-2012 | Perubahan jarak webcam dengan LJK                  |       |
| 25-06-2012 | Uji Coba Aplikasi Pada Software Secara Keseluruhan |       |
| 02-07-2012 | Perbaikan Format Laporan                           |       |
| 10-07-2012 | Penyempurnaan Aplikasi                             |       |

**Malang, 19-09-2012**

**Dosen Pembimbing II**

**Ir. Yusuf Ismail Nakhoda, MT**

**NIP.Y.1018800189**



## PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : CAHYO WIDYANTO  
Nim : 1012912  
Semester : VIII (Delapan)  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer

Dengan ini menyatakan bersedia/tidak bersedia\*) Membimbing skripsi dari mahasiswa tersebut, dengan judul :

**" RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR JAWAB KOMPUTER MENGGUNAKAN WEBCAM BERBASIS BORLAND DELPHI"**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Hormat Kami

Dr. Erg. Aryanto Soetedjo, ST, MT  
NIP.P. 1030300417

\*) Coret yang tidak perlu



PROGRAM STUDI TEKNIK ELEKTRO S-1  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
Kampus II (Jl. Raya Karanglo Km. 2 Tejo (0341) 417636 Malang

Jumlah Lembaran : 1 (satu) berkas  
Pembimbing Skripsi

Keperluan : Yth. Bapak/Ibu Dr. Eng. Aryunto Soetedjo, ST, MT  
Dosen Teknik Elektro S-1  
ITN Malang

Yang bertanda tangan dibawah

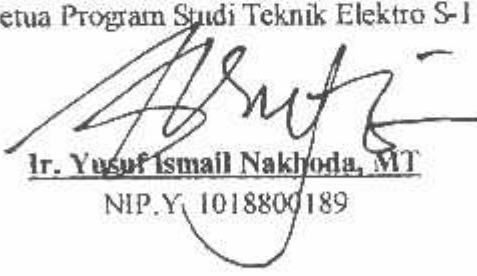
Nama : CAHYO WIDYANTO  
Nim : 1012912  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing untuk penyusunan Skripsi dengan judul :


**"RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR JAWAB KOMPUTER MENGGUNAKAN WEBCAM BERBASIS BORLAND DELPHI"**

Demikian permohonan kami buat dan atas kesediaan Bapak kami ucapkan terima kasih.

Mengetahui  
Ketua Program Studi Teknik Elektro S-1

  
Ir. Yusuf Ismail Nakhoda, MT  
NIP. Y. 1018800189

Hormat Kami,

  
CAHYO WIDYANTO  
NIM. 1012912



## PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

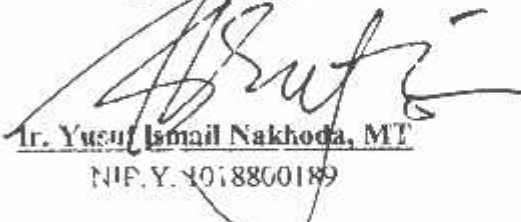
Nama : CAHYO WIDYANTO  
Nim : 1012912  
Semester : VIII (Delapan)  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer

Dengan ini menyatakan bersedia/tidak bersedia\*) Membimbing skripsi dari mahasiswa tersebut, dengan judul :

**" RANCANG BANGUN APLIKASI PENILAIAN PADA LEMBAR JAWAB KOMPUTER MENGGUNAKAN WEBCAM BERBASIS BORLAND DELPHI"**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Hormat Kami

  
Ir. Yusuf Ismail Nakhoda, MT  
NIP. Y. 1018800189

Catatan :

Setelah disetujui agar formulir ini diserahkan mahasiswa/i yang bersangkutan kepada jurusan untuk diproses lebih lanjut

\*) Coret yang tidak perlu



## LAMPIRAN

### *Listing Code Program :*

```
procedure TForm1.Keluar1Click(Sender: TObject);
begin
  Application.Terminate;
end;
procedure TForm1.BuatKunci1Click(Sender: TObject);
begin
  // Menampilkan form2 dan menutup akses form1
end;
procedure TForm1.FormCreate(Sender: TObject);
begin
  batas:=TrackBar2.Position;
  th:=TBthresh.Position;
  //menjadikan hasil image 10x10 pixel
  ImWarna.Picture.Bitmap.Width:=7;
  ImWarna.Picture.Bitmap.Height:=7;
  ImGray.Picture.Bitmap.Width:=7;
  ImGray.Picture.Bitmap.Height:=7;
  //agar webcam hidup
  SysDev:= TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
  FilterGraph.ClearGraph;
  FilterGraph.Active := false;
  Filter.BaseFilter.Moniker := SysDev.GetMoniker(0); //letak port webcam
  FilterGraph.Active := true;
  with FilterGraph as ICaptureGraphBuilder2 do
    RenderStream(@PIN_CATEGORY_PREVIEW, nil, Filter as IBaseFilter,
    SampleGrabber as IBaseFilter, VideoWindow as IbaseFilter);
  FilterGraph.Play;
  sleep(1000);
```

---

```

    SampleGrabber.GetBitmap(IFOTO.Picture.Bitmap);
end;
procedure TForm1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
begin
    SysDev.Free;
    FilterGraph.ClearGraph;
    FilterGraph.Active := false;
end;
procedure TForm1.btKoreksiClick(Sender: TObject);
var i, benar, salah, kosong, nilai: Integer;
begin
    //untuk ambil gambar
    SampleGrabber.GetBitmap(IFOTO.Picture.Bitmap);
bnama.Click;
bjwb.Click;
bTTL.Click;
bNOMOR.Click;
bPAKET.Click;
BJURUSAN.Click;
if pos('KUNCI', Enama.Text) > 0 then
    MJAWABAN.Lines.SaveToFile(EdPakso.Text+EdJur.Text+EdMapel.Text+'.csv');
if FileExists(EdPakso.Text+EdJur.Text+EdMapel.Text+'.csv')
then
begin
    MKunci.Lines.LoadFromFile(EdPakso.Text+EdJur.Text+EdMapel.Text+'.csv');
    benar:=0;
    salah:=0;
    kosong:=0;
    for i:=0 to 49 do
        if MKunci.Lines.Strings[i]=MJAWABAN.Lines.Strings[i]

```

---

```

        then benar:=benar+1
    else
        begin
            if Mjawab.Lines.Strings[i]-(IntToStr(i+1)+'.') then
                kosong:=kosong+1
            else
                salah:=salah+1;
            end;
            nilai:=(benar*2)-salah;
            ShowMessage('benar = '+IntToStr(benar)+' '+'salah = '+IntToStr(salah)+'
kosong = '+IntToStr(kosong)+' '+'nilai = '+IntToStr(nilai));
        end
    else
        ShowMessage(EdPakso.Text+EdJur.Text+EdMapel.Text+'.csv tidak ada');
end;

procedure TForm1.IFOTOMouseDown(Sender: TObject; Button: TMouseButton;
    Shift: TShiftState; X, Y: Integer);
//var ARect, BRECT: TRect;
    //xx,yy,jmlpixel: integer;
begin
    MKoord.Lines.Append(inttostr(x-STRTOINT(EKORX.Text)));
    MKoord.Lines.Append(inttostr(y-STRTOINT(EKORY.Text)));
    IFOTO.Picture.Bitmap.Canvas.FillRect(RECT(X-3,Y-3,X+3,Y+3));
end;

procedure TForm1.IFOTOMouseMove(Sender: TObject; Shift: TShiftState; X,
    Y: Integer);
var ARect, BRECT: TRect;
    xx,yy,jmlpixel,thku: integer;
begin
    Edit6.Text:=inttostr(x);
    edit7.Text:=inttostr(y);

```

---

```

ARect := Rect(0, 0, 7, 7);
BRect := Rect(x-3, y-3, x+3, y+3);

ImWarna.Picture.Bitmap.Canvas.CopyRect(arect, Ifoto.Picture.Bitmap.Canvas, brect);

ImBw.Picture:=ImWarna.Picture;
ARect := Rect(0, 0, 7, 7);
BRect := Rect(x-3, y-3, x+3, y+3);

ImGray.Picture.Bitmap.Canvas.CopyRect(arect, Ifoto.Picture.Bitmap.Canvas, brect);

ImBw.Picture:=ImWarna.Picture;
thku:=0;
for xx:=0 to 6 do
  for yy:=0 to 6 do
    begin
      thku:=thku+(ImWarna.Picture.Bitmap.Canvas.Pixels[xx,yy] mod
$100);
    end;
  Ed:=1.Text:=-inttostr(round(thku/(7*7)));
  jm_pixel:=0;
  for xx:=0 to 6 do
    for yy:=0 to 6 do
      begin
        then
          if ((ImWarna.Picture.Bitmap.Canvas.Pixels[xx,yy] mod $100)< th)
            BEGIN
              ImWarna.Picture.Bitmap.Canvas.Pixels[xx,yy]:=-0; // menjadi
warna hitam
            END
          ELSE
            BEGIN
              ImWarna.Picture.Bitmap.Canvas.Pixels[xx,yy]:=$fffff; //
menjadi warna putih
            END
          END
        end;
      end;
    end;
  end;
end;

```

---

```

        END;
    end;
    for xx:=0 to 6 do
        for yy:-0 to 6 do
            begin
lmGray.Picture.Bitmap.Canvas.Pixels[xx,yy]:=((ImBw.Picture.Bitmap.Canvas.Pixe
ls[xx,yy] mod $100)*$10101);
            end;
        for xx:=(X-3) to (X+3) do
            for yy:=(y-3) to (y+3) do
                begin
                    if ((IFOTO.Picture.Bitmap.Canvas.Pixels[xx,yy] mod $100)< th) then
                        BEGIN
                            jmlpixel:=jmlpixel+1;
                        END
                    end;
                Ejmlpixel.Text:=inttostr(jmlpixel);
            end;
        end;
    procedure TForm1.Button1Click(Sender: TObject);
    VAR YP,xp,XPOS,YPOS,x1,x2,y1,y2:INTEGER;
        XAWAL,YAWAL:REAL;
    begin
    MKoord.Text:='';
    x1:=StrToInt(Edit2.Text);
    y1:=StrToInt(Edit3.Text);
    x2:=StrToInt(Edit4.Text);
    y2:=StrToInt(Edit5.Text);
    yAWAL:=y1;
    FOR xp:=1 TO 5 DO
    BEGIN
        xAWAL:=-x1;

```

---

```

FOR yP:=1 TO 5 DO
BEGIN
    XPOS:=ROUND(XAWAL-0.5);
    yPOS:=ROUND(YAWAL-0.5);
    MKoord.Lines.Append(INTTOSTR(xPOS));
    MKoord.Lines.Append(INTTOSTR(yPOS));
    IFOTO.Picture.Bitmap.Canvas.FillRect(RECT(XPOS-3, YPOS-3, XPOS+3, YPOS+3));
    xAWAL:=-xAWAL+(x2-x1)/4;
END;
yAWAL:=yAWAL+(y2-y1)/4;
END;
end;
procedure TForm1.Button2Click(Sender: TObject);
VAR YP, xp, XPOS, YPOS, x1, x2, y1, y2: INTEGER;
    XAWAL, YAWAL: REAL;
begin
MKoord.Text:='-';
x1:=StrToInt(Edit2.Text);
y1:=StrToInt(Edit3.Text);
x2:=StrToInt(Edit4.Text);
y2:=StrToInt(Edit5.Text);
xAWAL:=-x1;
FOR xP:=1 TO 20 DO
BEGIN
    yAWAL:=y1;
    FOR yP:-1 TO 26 DO
    BEGIN
        XPOS:=ROUND(XAWAL+0.5);
        yPOS:=ROUND(YAWAL+0.5);
        MKoord.Lines.Append(INTTOSTR(xPOS));
    
```

---

```

MKoord.Lines.Append(INTOSTR(yPOS));
IFOTO.Picture.Bitmap.Canvas.FillRect(RECT(XPOS-3, YPOS-3, XPOS+3, YPOS+3));
yAWAL:=yAWAL+(y2-y1)/25;
END;
xAWAL:=xAWAL+(x2-x1)/19;
END;
end;
procedure TForm1.Button3Click(Sender: TObject);
VAR XP, YP, XPOS, YPOS: INTEGER;
    BAWAL: INTEGER;
begin
BAWAL:=0;
FOR yP:=1 TO 50 DO
BEGIN
    FOR xP:=1 TO 5 DO
    BEGIN
        XPOS:=STRTOINT('0'+MKoord.Lines.Strings[BAWAL]);
        BAWAL:=BAWAL+1;
        yPOS:=STRTOINT('0'+MKoord.Lines.Strings[BAWAL]);
        BAWAL:=BAWAL+1;
        IFOTO.Picture.Bitmap.Canvas.FillRect(RECT(XPOS-3, YPOS-3, XPOS+3, YPOS+3));
    END;
END;
END;
procedure TForm1.bnamaClick(Sender: TObject);
VAR XP, YP, XPCS, YPCS, xpj, ypj, jmpixel: INTEGER;
    nobaris, thku: INTEGER;
    huruf, NAMA: string;
begin
NAMA:='          ';
MKoord.Lines.LoadFromFile('koord nama.txt');

```

---

```

huruf:='ABCDEFGHIJKLMNOPQRSTUVWXYZ' ;
nobaris:-0;
FOR xP:=1 TO 20 DO
BEGIN
  FOR yP:=1 TO 26 DO
  BEGIN
    XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
    nobaris:=nobaris+1;
    YPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
    nobaris:=nobaris+1;
    thku:=0;
    for xpj:=(XPOS-3) to (XPOS+3) do
      for ypj:=(YPOS-3) to (YPOS+3) do
        begin
          thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
          end;
          thku:=round(thku/(7*7));
          for xpj:=(XPOS-3) to (XPOS+3) do
            for ypj:=(YPOS-3) to (YPOS+3) do
              begin
                if thku< th then
                  IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=$000000
                else
                  IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=$ffffff;
              end;
            if thku< th then
              NAMA[XP]:=HURUF[YP];
          END;
        END;
      ENAMA.Text:=NAMA;

```

---



```

procedure TForm1.bjwbClick(Sender: TObject);
VAR XP,YP,XPOS,YPOS,xpj,ypj,jmpixel:INTEGER;
    nobaris,thku:INTEGER;
    huruf,jwbn: string;
begin
MKoord.Lines.LoadFromFile('kordinat jawaban.txt');
MJAWABAN.Text:='';
huruf:='ABCDE' ;
nobaris:=0;
FOR yP:=1 TO 50 DO
BEGIN
    jwbn:='';
    FOR xP:=1 TO 5 DO
    BEGIN
        XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
        nobaris:=nobaris+1;
        yPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
        nobaris:=nobaris+1;
        thku:=0;
        for xpj:=(XPOS-3) to (XPOS+3) do
            for ypj:=(yPOS-3) to (yPOS+3) do
                begin
                    thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
                end;
            thku:=round(thku/(7*7));
            for xpj:=(XPOS-3) to (XPOS+3) do
                for ypj:=(yPOS-3) to (yPOS+3) do
                    begin
                        IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=thku*$10101;
                    end;
                end;
            end;
        end;
    end;
end;

```

---

```

    if thku<th then
        begin
            jwbn:=jwbn+HURUF[xP];
        end
    if jmlpixel> BATAS then
        begin
            jwbn:=jwbn+HURUF[xP];
        end
    else
        END;
    MJAWABAN.Lines.Append(INTTOSTR(YP)+'.'+jwbn);
    END;
end;
procedure TForm1.bNOMORClick(Sender: TObject);
VAR XP,YP,XPOS,YPOS,xpj,ypj,jmlpixel:INTEGER;
    nobaris,thku:INTEGER;
    huruf,NAMA: string;
begin
    NAMA:='          ';
    MKoord.Lines.LoadFromFile('koord nomor peserta.txt');
    huruf:='0123456789' ;
    nobaris:=0;
    FOR xP:=1 TO 9 DO
    BEGIN
        FOR yP:=1 TO 10 DO
        BEGIN
            XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
            nobaris:=nobaris+1;
            YPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
            nobaris:=nobaris+1;

```

---

```

jmlpixel:=0;
thku:=0;
for xpj:=(XPCS-3) to (XPOS+3) do
  for ypj:=(yPOS-3) to (yPOS+3) do
    begin
      thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
    end;
  thku:=round(thku/(7*7));
  for xpj:=(XPOS-3) to (XPOS+3) do
    for ypj:=(yPOS-3) to (yPOS+3) do
      begin
        IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=thku*$10101;
      end;
    if thku< th then
      NAMA[XP]:=HURUF[YP];
    END;
  END;
END;
EDITS.Text:=-NAMA;
end;
procedure TForm1.btTLClick(Sender: TObject);
VAR XP,YP,XPOS,YPOS,xpj,ypj,jmlpixel:INTEGER;
    nobaris,thku:INTEGER;
    huruf,ttg: string;
begin
ttg:='      ';
MKoord.Lines.LoadFromFile('koord ttl.txt');
huruf:='0123456789' ;
nobaris:=0;
FOR xP:-1 TO 9 DO
BEGIN

```

---

```

FOR yP:=1 TO 10 DO
BEGIN
  XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
  nobaris:=nobaris+1;
  yPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
  nobaris:=nobaris+1;
  jmpixel:=0;
  thku:=0;
  for xpj:=(XPOS-3) to (XPOS+3) do
    for ypj:=(yPOS-3) to (yPOS+3) do
      begin
        thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
        end;
        thku:=round(thku/(7*7));
        for xpj:=(XPOS-3) to (XPOS+3) do
          for ypj:=(yPOS-3) to (yPOS+3) do
            begin
              IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=thku*$10101;
            end;
            if thku< th then
              ttg[XP]:=HURUF[YP];
            END;
          END;
        END;
      END;
    END;
  END;
  EDIT9.Text:=ttg;
  END;
procedure TForm1.bPAKETClick(Sender: TObject);
VAR XP,YP,XPOS,YPOS,xpj,ypj,jmpixel:INTEGER;
    nobaris,thku:INTEGER;
    huruf,NAMA: string;
begin

```

---

```

NAMA:='          ';
MKoord.Lines.LoadFromFile('koord paket soal.txt');
huruf:='0123456789' ;
nobaris:=0;
FOR xP:=1 TO 2 DO
BEGIN
  FOR yP:=1 TO 10 DO
  BEGIN
    XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
    nobaris:=nobaris+1;
    YPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
    nobaris:=nobaris+1;
    jmlpixel:=0;
    thku:=0;
    for xpj:=(XPOS-3) to (XPOS+3) do
      for ypj:=(YPOS-3) to (YPOS+3) do
        begin
          thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
        end;
      thku:=round(thku/(7*7));
      for xpj:=(XPOS-3) to (XPOS+3) do
        for ypj:=(YPOS-3) to (YPOS+3) do
          begin
            IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=thku*$10101;
          end;
        if thku< th then
          NAMA[xP]:=HURUF[yP];
        END;
      END;
    EdPakso.Text:=NAMA;
  
```

---

```

END;
procedure TForm1.BJURUSANClick(Sender: TObject);
//KOORD JURUSAN.txt
VAR XP,XPOS,YPOS,xpj,ypj,jmlpixel:INTEGER;
    nobaris,thku:INTEGER;
    huruf,NAMA: string;
begin
MKoord.Lines.LoadFromFile('KCCRD JURUSAN.txt');
huruf:='0123456789' ;
nobaris:=0;
FOR xP:=1 TO 3 DO
BEGIN
    XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
    nobaris:=nobaris+1;
    yPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
    nobaris:=nobaris+1;
    jmlpixel:=0;
    thku:=0;
    for xpj:=(XPOS-3) to (XPOS+3) do
        for ypj:=(yPOS-3) to (yPOS+3) do
            begin
                thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
            end;
            thku:=round(thku/(7*7));
        for xpj:=(XPOS-3) to (XPOS+3) do
            for ypj:=(yPOS-3) to (yPOS+3) do
                begin
                    IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=thku*$10101;
                end;
            if thku< th then

```

---

```

        NAMA:=MJURUSAN.Lines.Strings[XP];
END;
EdJur.Text:=NAMA;
FOR xP:=1 TO 9 DO
BEGIN
    XPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORX.Text);
    nobaris:=nobaris+1;
    yPOS:=STRTOINT('0'+MKoord.Lines.Strings[nobaris])+STRTOINT(EKORY.Text);
    nobaris:=nobaris+1;
    jmpixel:=0;

    thku:=0;
    for xpj:=(XPOS-3) to (XPOS+3) do
        for ypj:=(yPOS-3) to (yPOS+3) do
            begin
                thku:=(IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] mod
$100)+thku;
            end;
            thku:=round(thku/(7*7));
            for xpj:=(XPOS-3) to (XPOS+3) do
                for ypj:=(yPOS-3) to (yPOS+3) do
                    begin
                        IFOTO.Picture.Bitmap.Canvas.Pixels[xpj,ypj] :=thku*$10101;
                    end;
                if thku< th then
                    NAMA:=mmapel.Lines.Strings[XP];
            end;
END;
EdMapel.Text:=NAMA;
end;

```

---