

**IDENTIFIKASI UANG KERTAS BERDASARKAN WARNA DENGAN
METODE TEMPLATE MATCHING**

Tugas Akhir disusun untuk memenuhi syarat
Mencapai gelar Kesarjanaan Komputer pada
Program Studi Teknologi Informasi
Jenjang Program Strata-1



Oleh :

YEHUDA YOHANES RAPSUDIA

06.01.53.0036

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS STIKUBANK (UNISBANK)**

SEMARANG

2013

PERNYATAAN KESIAPAN UJIAN TUGAS AKHIR

Saya Yehuda Yohanes Rapsudia, dengan ini menyatakan bahwa Laporan Tugas

Akhir yang berjudul:

“IDENTIFIKASI UANG KERTAS BERDASARKAN WARNA DENGAN METODE TEMPLATE MATCHING”

Adalah benar hasil karya saya dan belum pernah diajukan sebagai karya ilmiah, sebagian atau seluruhnya, atas nama saya atau pihak lain.

(Yehuda Yohanes Rapsudia)

NIM :06.01.53.0036

Disetujui oleh pembimbing

Kami setuju Laporan tersebut diajukan untuk Ujian Tugas Akhir

Pembimbing I

Semarang, Februari2013

(Dwi Agus Diartono, S.Kom,M.Kom)

Pembimbing II

Semarang, Februari2013

(Felix Andreas Sutanto, S.Kom, M.Cs)

HALAMAN PENGESAHAN

Telah dipertahankan di depan tim dosen penguji Tugas Akhir Fakultas Teknologi Informasi UNIVERSITAS STIKUBANK (UNISBANK) Semarang dan diterima sebagai salah satu syarat guna menyelesaikan Jenjang Program Strata 1, Program Studi : Teknik Informatika

Semarang :Februari2013

Ketua

(Dwi Agus Diartono, S.Kom,M.Kom)

Sekretaris

(Felix Andreas Sutanto, S.Kom, M.Cs)

Anggota

(Siti Munawaroh S. Kom M. Cs)

MENGETAHUI :

UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG

Fakultas Teknologi Informasi

Dekan

(Dwi Agus Diartono, S.Kom, M.Kom)

MOTTO DAN PERSEMBAHAN

MOTTO

- Takut akan TUHAN adalah permulaan pengetahuan.
- Ketika seseorang berpikir tidak bisa, maka ia sesungguhnya telah membuang kesempatan untuk bisa.
- Semua impian kita dapat menjadi nyata, jika kita memiliki keberanian untuk mengejanya.
- Jangan repot-repot untuk menjadi lebih baik dari generasi terdahulu, cobalah untuk menjadi lebih baik dari diri sendiri.
- Sahabat adalah seseorang yang datang menghampirimu ketika semua orang di dunia meninggalkanmu.

PERSEMBAHAN

Skripsi ini saya persembahkan kepada :

MY LORD JESUS CHRIST

Ayah, ibu, kakak dan adik yang selalu mendukungku dalam doanya dan yang selalu mencintaiku.

Buat herlin mety, panatus, tunjung yang selalu menguatkan dan tetap memberiku semangat.

Bapak dwi Agus dan Bapak Felix yang membimbing saya dalam menyelesaikan tugas akhir ini.

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG**

**Program Studi: Teknik Informatika
Tugas Akhir Sarjana Komputer**

**IDENTIFIKASI UANG KERTAS BERDASARKAN WARNA DENGAN
METODE TEMPLATE MATCHING**

**IDENTIFICATION OF INDONESIAN MONEY BASED ON COLOR WITH
TEMPLATE MATCHING METHOD**

Yehuda Yohanes Rapsudia
NIM :06.01.53.0036

Abstrak

Dengan berkembangnya teknologi, media penyampaian informasi pun semakin beragam. Salah satunya adalah yang berkaitan dengan pengenalan objek. Aplikasi object recognition yang telah ada salah satunya adalah proses identifikasi citra pada uang kertas berdasarkan RGB. Dalam proyek akhir ini akan merancang bangun untuk mengidentifikasi mata uang kertas berdasarkan RGB.

Dalam pembagian warna umumnya harus dilakukan inisialisasi jumlah yang diinginkan terlebih dahulu. Untuk itu dalam proyek akhir ini menggunakan aplikasi delphi untuk menyelesaikan masalah tersebut.

Kata kunci

Image, rgb, uang

Semarang, Februari 2013

Pembimbing I

Pembimbing II

(Dwi Agus Diartono, S.Kom, M.Kom)

(Felix Andreas Sutanto, S.Kom, M.Cs)

KATA PENGANTAR

Terima kasih kepada Tuhan Yesus, atas berkat dan limpahan kasihNYa sehingga penyusun dapat menyelesaikan Laporan Tugas Akhir ini dengan baik sehingga terbentuklah suatu Tugas Akhir yang berjudul **“IDENTIFIKASI UANG KERTAS BERDASARKAN WARNA DENGAN METODE TEMPLATE MATCHING”**.Skripsi ini disusun untuk memenuhi salah satu syarat kelulusan program studi strata satu jurusan Teknik Informatika di UNISBANK Semarang.

Tugas Akhir ini dapat penyusun selesaikan berkat kerja sama dari berbagai pihak, baik moril maupun materil. Oleh karena itu, penyusun ingin menyampaikan rasa hormat dan terima kasih yang sebesar–besarnya kepada :

1. Bapak Dr.Bambang Suko Priyono, MM Selaku Rektor Universitas Stikubank Semarang.
2. Bapak Dwi Agus Diantoro, S.Kom, M.Kom Selaku Dekan Fakultas Teknologi Informasi sekaligus Pembimbing I yang telah memberikan bimbingan dan pengarahan dengan sabar kepada penulis.
3. Ibu Dewi Handayani UN, S.Kom M.Kom Selaku Kaprogdi Jurusan Teknik Informatika.
4. Bapak Felix Andreas Sutanto, S.Kom, M.Cs Selaku Pembimbing II yang telah memberikan bimbingan dan pengarahan dengan sabar kepada penulis.
5. Bapak,Ibu dan keluarga tercinta yang tidak pernah henti-hentinya mendo’akan dan memberi dukungan baik moral maupun material untuk menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwasanya dalam penyusunan Tugas Akhir ini masih jauh dari kesempurnaan.Untuk itu, dengan kerendahan hati penyusun mohon maaf dan penyusun sangat mengharapkan segala saran dan kritikan yang

bersifat membangun dari semua pihak. Akhir kata, semoga Laporan Tugas Akhir ini dapat menambah pengetahuan bagi pembaca.

Semarang, Februari 2013

Penyusun,

(Yehuda Yohanes Rapsudia)

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERNYATAAN KESIAPAN UJIAN TUGAS AKHIR	ii
HALAMAN PENGESAHAN	iii
MOTTO DAN PERSEMBAHAN	iv
ABSTRAKSI	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
 BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	1
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	2
1.5. Manfaat Penelitian	2
1.6. Metodologi Penelitian	3

1.6.1.	Metode Pengumpulan Data	3
1.6.2.	Metode Pengembangan Sistem	4
1.7.	Sistematika Penulisan	5
BAB II	LANDASAN TEORI	
2.1.	Citra Digital	7
2.2.	Template Matching	11
2.3.	Ciri Gambar	14
2.4.	Histogram	16
2.5.	Data Flow Diagram (DFD)	19
2.5.1.	Data Flow Diagram Level Nol	20
2.5.2.	Data Flow Diagram Level Rinei	21
2.6.	Entity –Relationship Diagram (ERD)	23
2.7.	Flowchart	25
2.8.	Bahasa Pemrograman Delphi	29
2.8.1.	Program Aplikasi Windows dan Form	30
2.8.2.	Paket Komponen	31

2.8.3. IDE (Integrated Development Environment)	32
2.8.4. Menu Bar.....	33
BAB III ANALISA DAN PERANCANGAN SISTEM	
3.1. Perancangan Sistem	35
3.2. Extraksi Ciri	37
3.3. Macthing	38
3.4. Implementasi	39
3.5. DFD	41
3.6. Perancangan Form	43
3.6.1. Menu Utama	43
3.6.2. Sub Menu Isi Template	44
3.6.3. Sub Menu Analisa	45
3.7. Flowchart	45
BAB IV IMPLEMENTASI SISTEM	
4.1. Perangkat Keras (Hardware)	47
4.2. Perangkat Lunak (Software)	48

4.3. Pemilihan Brainware	48
4.4. Tampilan Program	49
4.4.1. Tampilan Form Menu Utama	49
4.4.2. Tampilan Form Isi Template	54
4.4.3. Tampilan Open File	58
4.4.4. Tampilan Form Analisa	59
 BAB V PENUTUP	
5.1. Kesimpulan	65
5.2. Saran	66
 DAFTAR PUSTAKA	67
 LAMPIRAN	
Listing Program	

DAFTAR GAMBAR

Gambar 2.1.	Jenis Format Citra	11
Gambar 2.2.	Template Matching	13
Gambar 2.3.	Konsep Warna	15
Gambar 2.4.	RGB 24-bit color cube	16
Gambar 2.5.	Pemetaan RGB cube dengan sumbu x,y,z	16
Gambar 2.6.	Pencampuran warna dasar RGB	17
Gambar 2.7.	Warna pada tiap pixel	18
Gambar 2.8.	Simbol-simbol program DFD	19
Gambar 2.9.	Contoh diagram konteks	20
Gambar 2.10.	Contoh diagram level 1	21
Gambar 2.11.	Contoh diagram level 2	22
Gambar 2.12.	Contoh hubungan satu ke satu	24
Gambar 2.13.	Contoh hubungan satu ke banyak	25
Gambar 2.14.	Contoh hubungan banyak ke banyak	25

Gambar 2.15.	Simbol-simbol flowchart	29
Gambar 2.16.	Tampilan program Delphi 7.0	32
Gambar 3.1.	Contect Diagram	41
Gambar 3.2.	DFD Level 1	42
Gambar 3.3.	Rancangan Menu Utama	43
Gambar 3.4.	Rancangan Isi Template	44
Gambar 3.5.	Rancangan Analisa	45
Gambar 3.6.	Flowchart Isi Template	46
Gambar 3.7.	Flowchart Analisa	46
Gambar 4.1.	Form Menu Utama	50
Gambar 4.2.	Tampilan Form Input Template	54
Gambar 4.3.	Tampilan Open File	58
Gambar 4.4.	Tampilan Form Analisa	59

BAB I

PENDAHULUAN

1.1. LATAR BELAKANG

Dengan berkembangnya teknologi, media penyampaian informasi pun semakin beragam. Salah satunya adalah yang berkaitan dengan pengenalan objek. Aplikasi object recognition yang telah ada salah satunya adalah proses identifikasi citra pada uang kertas berdasarkan RGB. Dalam proyek akhir ini akan merancang bangun untuk mengidentifikasi mata uang kertas berdasarkan RGB. Dalam pembagian warna umumnya harus dilakukan inisialisasi jumlah yang diinginkan terlebih dahulu. Untuk itu dalam proyek akhir ini menggunakan aplikasi java untuk menyelesaikan masalah tersebut.

Hal inilah yang melatar belakangi penulis untuk mengetahui dari pengenalan objek dengan membuat suatu aplikasi citra dengan mengusung judul **“IDENTIFIKASI UANG KERTAS BERDASARKAN WARNA DENGAN METODE TEMPLATE MATCHING”**.

1.2. RUMUSAN MASALAH

Berdasarkan latar belakang permasalahan yang telah diuraikan di atas maka dapat di ambil rumusan yang akan menjadi pembahasan penelitian ini yaitu bagaimana mengidentifikasi uang kertas apa saja yang ada pada gambar input.

1.3. BATASAN MASALAH

Mengingat luasnya permasalahan maka untuk menjaga agar tidak terjadi penyimpangan terhadap permasalahan, penulis hanya membahas tentang :

- a. Objek yang digunakan adalah mata uang kertas dimulai dari yang terendah sampai yang nominal uang kertas besar.
- b. Objek yang digunakan sudah tersimpan dalam format JPG, dengan ukuran 720 x 360.
- c. Software dirancang menggunakan Borland Delphi.
- d. Ciri yang dipakai adalah ciri warna dengan fitur R, G dan B.

1.4. TUJUAN PENELITIAN

Penelitian ini bertujuan untuk merancang dan membangun suatu aplikasi yang mampu mengenali atau mengidentifikasi uang kertas berdasarkan RGB. Diharapkan penelitian ini dapat memberikan manfaat baik bagi penulis, pembaca/pengguna, Universitas STIKUBANK Semarang.

1.5. MANFAAT PENELITIAN

Manfaat dari penelitian ini :

- a. Bagi Penulis

- Sebagai syarat untuk menyelesaikan pendidikan strata 1 (S1) di Universitas STIKUBANK (UNISBANK) Semarang.
 - Dapat menambah pengetahuan dan pengalaman.
- b. Bagi Pembaca/Pengguna.
- Pengguna mendapat pengetahuan tentang metode pengolahan citra digital, serta bagaimana pula menjalankan aplikasinya.
- c. Bagi Universitas STIKUBANK Semarang
- Menambah ragam hasil penelitian yang dapat digunakan sebagai bahan acuan bagi peneliti lainnya.

1.6. METODOLOGI PENELITIAN

Metodologi penelitian tergantung pada masalah yang akan dipecahkan, tujuan yang akan dicapai dan berbagai alternatif yang mungkin digunakan.

1.6.1. METODE PENGUMPULAN DATA

1. Observasi

Yaitu metode pengumpulan data dengan dengan cara melihat secara langsung pada objek yang digunakan untuk penelitian.

2. Studi Pustaka

Yaitu pengumpulan data dengan membaca literature – literature atau buku- buku yang berhubungan dengan penelitian ini.

Dari hasil pengumpulan data tersebut diharapkan dapat diperoleh data sebagai berikut :

1. Data Primer.

Data yang diperoleh secara langsung dari sumber atau objek penelitian.

2. Data Sekunder.

Data yang diperoleh dari sumber lain, diantaranya buku-buku, literature- literature maupun sumber dari internet.

1.6.2. METODE PENGEMBANGAN SISTEM

Pada penelitian untuk mengidentifikasi nominal uang kertas indonesia berdasarkan RGB dilakukan beberapa tahapan sebagai berikut:

- Uang kertas yang akan digunakan sebagai sampel untuk obyek penelitian.
- Dalam program aplikasi dilakukan beberapa langkah pengolahan citra digital sebagai berikut:
 - a. Image yang dimasukkan berupa gambar.
 - b. Jenis citra yang digunakan adalah citra berwarna, dengan format JPEG.

1.7. SISTEMATIKA PENULISAN

Dalam penyusunan tugas akhir ini, dengan judul “**IDENTIFIKASI UANG KERTAS BERDASARKAN WARNA DENGAN METODE TEMPLATE MATCHING**”, penulis membagi dalam berbagai bab, sebagai berikut :

BAB I PENDAHULUAN

Didalamnya menguraikan mengenai latar belakang, tujuan dan manfaat serta sistematika laporan.

BAB II LANDASAN TEORI

Bab ini berisi tinjauan penelitian terdahulu dan uraian beberapa landasan teori yang ada hubungannya dengan pokok permasalahan yang akan dipilih, yang akan dijadikan landasan penulisan tugas akhir

BAB III ANALISA DAN PERANCANGAN

Bab ini berisi tentang data yang dibutuhkan dan algoritma yang dibutuhkan.

BAB IV IMPLEMENTASI SISTEM

Didalamnya menguraikan tentang pembahasan dan implementasi program.

BAB V PENUTUP

Didalamnya menguraikan tentang kesimpulan dan saran.

BAB II

LANDASAN TEORI

2.1 Citra Digital

Citra didefinisikan sebagai intensitas cahaya dua dimensi $f(x,y)$, dimana x dan y menyatakan koordinat dan nilai f pada setiap titik (x,y) menyatakan intensitas atau kecerahan dari citra pada titik tersebut. Citra digital adalah citra $f(x,y)$ yang diubah koordinatnya dan kecerahannya kedalam bentuk diskrit. Citra digital dapat dianggap sebagai matrik dimana index baris dan kolomnya menyatakan kecerahan pada titik tersebut (Gonzalez, 2001). Titik pada citra digital disebut pixel, jangkauan nilai f disebut jangkauan aras kelabu (kecerahan). Jika f hanya mempunyai dua nilai f maka disebut citra biner. Citra digital 1 bit merupakan citra ditra digital yang mempunyai jangkauan aras kelabu 1 bit atau mempunyai dua nilai intensitas (21). Nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Citra aras kelabu memperbolehkan mempunyai lebih dari dua nilai intensitas. Jangkauan aras kelabu dinyatakan bit dimana 1 bit menyatakan citra mempunyai 21 atau dua intensitas, 2 bit menyatakan citra mempunyai 22 atau empat nilai intensitas, dan seterusnya (Hong,2002,h118).

Pada saat pendigitalan citra terjadi dua macam penurunan kualitas citra yaitu penurunan kualitas secara deterministic (*Deterministic degradations*),

seperti kabur (*blur*) yang disebabkan penyimpangan sistem optis, gerakan, kekacauan atmosfer, serta ketidak linearan film dan penurunan kualitas secara statistic (*statistical degradations*), seperti derau (*noise*) yang disebabkan sensor-sensor pencitraan elektronis, granularitas film, dan fluktuasi cahaya atmosfer. Kedua penurunan kualitas citra tersebut dapat terjadi secara bersamaan pada saat pendigitalan.

Citra (*Image*) merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Secara matematis fungsi intensitas cahaya pada bidang dua dimensi dinotasikan sebagai $f(x,y)$ dimana $f(x,y)$ merupakan koordinat pada bidang dua dimensi dan $f(x,y)$ merupakan intensitas cahaya pada bidang dua dimensi.

Agar dapat diolah dengan komputer digital, maka suatu citra harus direpresentasikan secara numerik dengan nilai-nilai diskrit. Representasi citra dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi dan citra yang dihasilkan merupakan citra digital.

Sebuah citra dapat diubah ke bentuk digital agar dapat disimpan dalam memori komputer atau media lain. Proses mengubah citra ke bentuk digital bisa dilakukan dengan beberapa perangkat, misalnya *scanner*, kamera digital dan handycam. Ketika sebuah citra sudah diubah ke dalam bentuk digital (selanjutnya

disebut dengan citra digital), bermacam-macam proses pengolahan citra dapat diperlakukan terhadap citra tersebut.

Berikut ini adalah contoh format citra baik yang lossless maupun lossy :

Ekstensi	Nama	Keterangan
Jpg/jpeg	Joint Photographic Experts Group	JPEG biasanya digunakan untuk foto atau citra di website. JPEG menggunakan kompresi tipe <i>lossy</i> . Kualitas JPEG 2000 bisa bervariasi tergantung setting kompresi yang digunakan. Kompresi JPEG berbasis CT(<i>Discrete Cosine Transform</i>)
jp2/jpg2/j2k	Joint Photographic Experts Group 2000	Merupakan pengembangan dari JPEG yang berbasis transformasi wavelet. Format ini mendukung kompresi tipe <i>lossless</i> dan <i>lossy</i> . Namun,

		support JPEG 2000 dalam berbagai aplikasi masih kurang, disebabkan kebutuhan hardware yang tangguh dan paten.
Pbm	Portable Bitmap Format	Merupakan format citra hitam putih yang sederhana. PBM memerlukan 1 bit tiap pixel. Tidak seperti format citra lainnya, format PBM merupakan <i>plain text</i> yang bisa diolah dengan menggunakan pengolah text. Format PBM merupakan bagian dari PNM (Portable Pixmap File Format).
Pgm	Portable Graymap Format	Merupakan format citra abu-abu yang sederhana. Format PGM memerlukan 8 bit tiap pixel. PGM merupakan citra mentah dengan kompresi tipe

		<i>lossless</i> . Format PGM merupakan bagian dari PNM (Portable Pixmap File Format).
Ppm	Portable Pixmap Format	Merupakan format citra berwarna yang sederhana. PPM memerlukan 24 bit tiap pixel. PPM merupakan citra mentah dengan kompresi tipe <i>lossless</i> . Format PPM merupakan bagian dari PNM (Portable Pixmap File Format).
Png	Tagged Image File Format	Merupakan format citra yang sudah digunakan sejak dulu. Mendukung kompresi tipe <i>lossless</i> dan <i>lossy</i> .

Gambar 2.1 Jenis Format Citra

Format yang berbeda-beda ini mengakibatkan adanya perbedaan pula dalam proses pembacaannya. Dengan menggunakan toolox image processing di matlab, proses pembacaan gambar dapat dilakukan dengan mudah.

2.2 Template Matching

Metode template matching merupakan salah satu metode yang cukup populer digunakan dalam permasalahan pencocokan pola pada pengolahan citra digital. Banyak peneliti yang telah menerapkan implementasi template matching ini untuk berbagai aplikasi di masyarakat. Salah satu penelitian yang pernah dilakukan adalah penerapan metode template matching untuk pendeteksian kendaraan yang dilakukan oleh Thiang dkk. Sementara Mulyadi dkk telah menerapkan pula penggunaan template matching untuk melakukan identitas lampu lalu lintas. Dalam kedua penelitian tersebut disimpulkan bahwa pengguna metode template matching telah memberikan hasil yang memuaskan untuk mengenali jenis-jenis kendaraan dan lampu lalu lintas.

Sementara itu Acmad Hidayanto dkk, telah melakukan penelitian terkait penggunaan template matching ini untuk penentuan wilayah wajah manusia pada citra berwarna berdasarkan warna kulit. Pada penelitian ini disimpulkan bahwa penggunaan informasi warna kulit dan metode template matching dapat digunakan untuk mendeteksi wajah manusia dalam citra berwarna.

Dari sejumlah penelitian diatas maka pada prinsipnya metode template matching memiliki karakteristik antara lain :

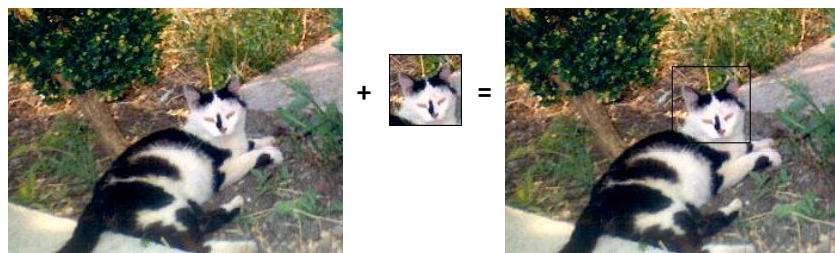
1. Relatif mudah untuk diaplikasikan dalam teknik pengolahan citra digital.

2. Hasilnya relative sangat akurat karena mendeteksi kesalahan hingga ukuran piksel.
3. Walaupun demikian, metode ini cukup rentan terhadap orientasi antara citra acuan (template) dengan citra yang akan diidentifikasi, yang meliputi : ukuran, posisi dan kualitas citra.
4. Untuk mendapatkan hasil yang maksimal maka metode ini sangat tergantung pada teknik pengolahan citra digital yang lain seperti enhancement, filtering, dll.

Template matcing adalah salah satu teknik dalam pengolahan citra digital yang berfungsi untuk mencocokkan tiap-tiap bagian dari suatu citra dengan citra yang menjadi template (acuan).

Metode template matching adalah salah satu metode terapan dari teknik konvolusi. Metode ini sering digunakan untuk mengidentifikasi citra karakter huruf, angka, sidik jari dan aplikasi-aplikasi pencocokkan citra lainnya. Secara umum teknik konvolusi didefinisikan sebagai suatu cara untuk mengkombinasikan dua buah deret angka yang menghasilkan deret angka ke tiga.

Contoh :



Gambar 2.2 Template Matching

2.3 Ciri Gambar

Ciri merupakan suatu tanda yang khas, yang membedakan antara satu dengan yang lain. Tidak berbeda dengan sebuah gambar yang lain. Masing-masing ciri gambar didapatkan dari proses ekstraksi ciri.

Ciri-ciri dasar dari gambar :

1. Warna

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

2. Bentuk

Ciri bentuk suatu gambar dapat ditentukan oleh tepi (sketsa) atau besaran moment dari suatu gambar. Pemakaian besaran moment pada ciri bentuk ini banyak digunakan orang dengan memanfaatkan nilai-nilai transformasi fourier dari gambar.









Proses yang dapat digunakan untuk menentukan ciri bentuk adalah deteksi tepi, *threshold*, segmentasi dan perhitungan moment seperti mean, median dan standart deviasi dari setiap lokal gambar.

3. Tekstur

Ciri tekstur dari suatu gambar dapat ditentukan dengan menggunakan filter gabor.

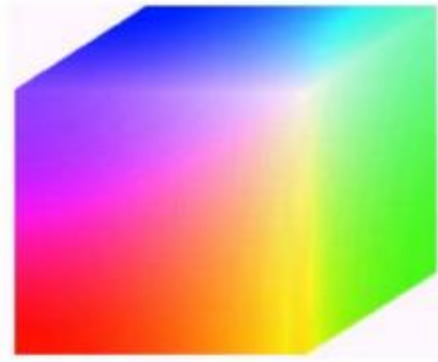
Ciri tekstur ini sangat handal dalam menentukan informasi suatu gambar bila digabungkan dengan ciri warna gambar. Dari ketiga ciri diatas, dalam hal ini menggunakan ciri warna dan ciri bentuk.

Warna pokok dalam pengelolaan gambar terdiri dari 3 (tiga) unsur, yaitu merah (R), hijau (H) dan biru (B). Jika warna-warna pokok tersebut digabungkan, maka akan menghasilkan warna lain. Penggabungan warna tersebut bergantung pada warna pokok dimana tiap-tiap warna memiliki nilai 256 (8bit).

	(255, 0, 0)
	(255, 255, 0)
	(255, 255, 255)
	(0, 255, 0)
	(0, 0, 255)
	(128, 128, 128)
	(255, 0, 255)
	(0, 0, 0)

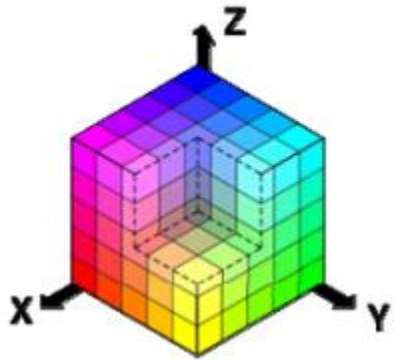
Gambar 2.3 Konsep warna

Konsep ruang warna adalah setiap pixel mempunyai warna yang dinyatakan dalam RGB, sehingga merupakan gabungan nilai R, nilai G dan nilai B yang tidak bisa dipisahkan satu dengan yang lainnya. Hal ini dapat dituliskan dengan $P(r,g,b)$.



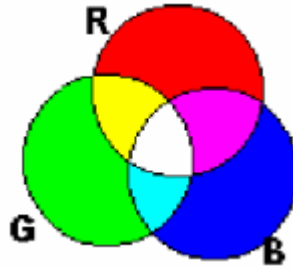
Gambar 2.4 RGB 24-bit color cube

Warna yang dideskripsikan dengan RGB adalah pemetaan yang mengacu pada panjang gelombang dari RGB. Pemetaan menghasilkan nuansa warna untuk masing-masing R,G dan B. Masing-masing R, G dan B di diskritkan dalam skala 256 sehingga RGB akan memiliki indeks antara 0 sampai 255. Jika dilihat dari pemetaan model warna RGB yang berbentuk cube (kubus) seperti gambar dibawah.



Gambar 2.5 Pemetaan RGB cube dengan sumbu x,y,z

Dengan pemetaan RGB 24-bit color cube maka 3 warna dasar dapat dicampurkan sehingga mendapatkan warna yang baru.



Gambar 2.6 pencampuran warna dasar RGB

2.4 Histogram

Color histogram merupakan gabungan dari intensitas tiga macam warna. Dimana setiap gambar mempunyai distribusi warna tertentu. Distribusi warna ini dimodelkan dengan color histogram. Color histogram tersebut didefinisikan sebagai berikut :

$$H_{R,G,B}[r,g,b] = N.Prob \{R=r, G=g, B=b\} \quad (2.4.1)$$

Dimana R,G,B merupakan tiga macam warna dan N adalah jumlah pixel pada gambar.

Color histogram dihitung dengan cara mendiskritkan warna dalam gambar dan menghitung jumlah dari tiap-tiap pixel pada gambar. Karena jumlah dari tiap-tiap warna terbatas, maka untuk lebih tepatnya dengan cara mentransform 3 histogram

kedalam single variable histogram. Misalkan pada gambar RGB salah satu transformnya didefinisikan sebagai berikut :

$$M = r + Nrg + NrNg b \quad (2.4.2)$$

Dimana Nr, Ng dan Nb merupakan jumlah biner dari warna merah, hijau dan biru secara berturut-turut. Untuk mendapatkan color histogram menggunakan persamaan sebagai berikut :

$$H_{r,g,b} = \{(r, g, b) \mid r \in I, g \in I, b \in I\} \quad (2.4.3)$$

Keterangan :

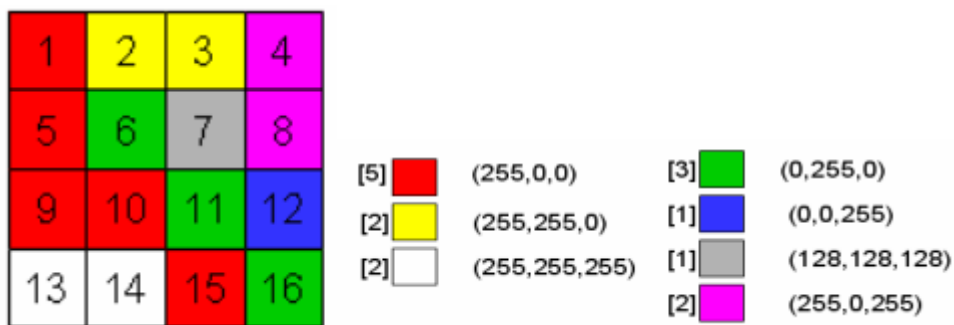
R = warna merah

G = warna hijau

B = warna biru

Hr,g,b = data untuk menampung nilai probabilitas warna RGB

Contoh histogram warna :







Gambar 2.7 Warna pada tiap pixel

2.5 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan diagram yang digunakan untuk menggambarkan proses-proses yang terjadi pada sistem yang akan dikembangkan. Dengan model ini, data-data yang terlibat pada masing-masing proses dapat diidentifikasi. (Hanif Al Fatta, 2007).

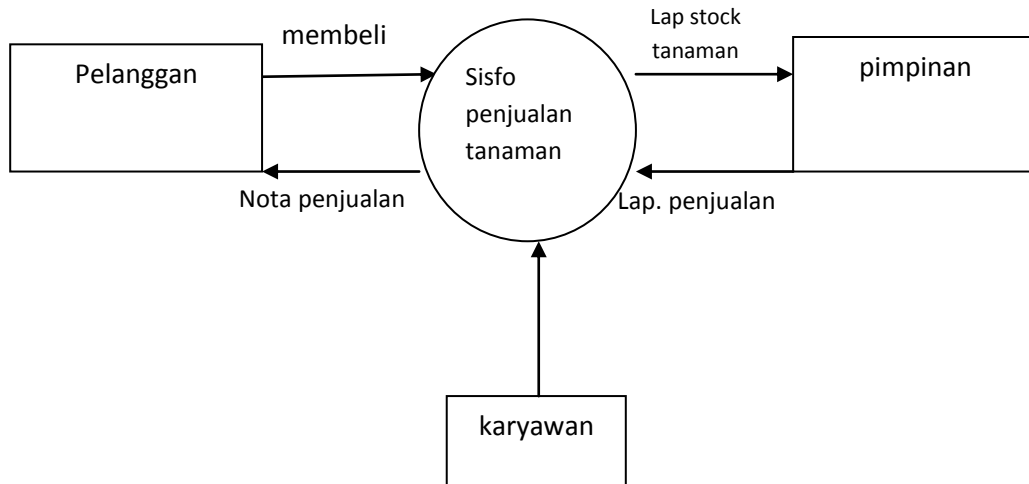
Pegembangan DFD biasanya menggunakan cara berjenjang, dimulai dari konteks diagram DFD level 1, level 2 dan seterusnya sesuai dengan kompleksitas dari sistem yang akan dikembangkan.

Untuk membaca suatu DFD harus memahami dulu elemen-elemen yang menyusun DFD. Ada 4 elemen yang menyusun suatu DFD, yaitu :

Simbol	Keterangan
	External entity
	Data Store
	Proses
	Data Flow

Gambar 2.8 Simbol-simbol Program DFD

2.5.1. Data Flow Diagram Level Nol



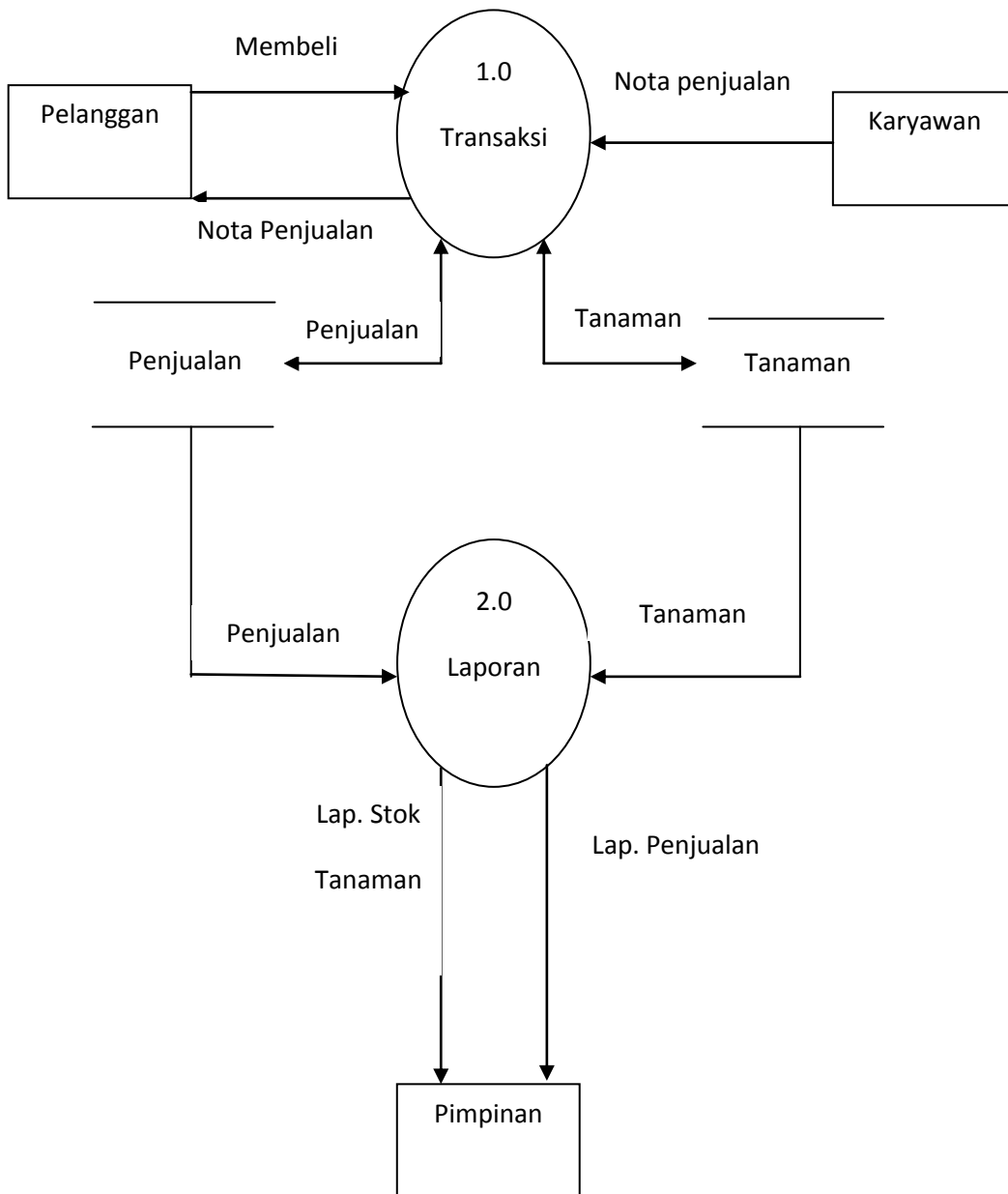
Gambar 2.9 Contoh digram konteks

Keterangan :

Dari diagram level nol diatas terlihat pelanggan membeli tanaman, kemudian karyawan memberikan nota penjualan sebagai bukti pembelian atas tanaman tersebut. Nota penjualan tersebut di simpan kedalam data base sistem. Setiap bulannya sistem akan memberikan laporan ke pemimpin toko yang diambil dari transaksi penjualan dalam satu bulan.

2.5.2. Data Flow Diagram Level Rinci

1. Level Rinci Transaksi

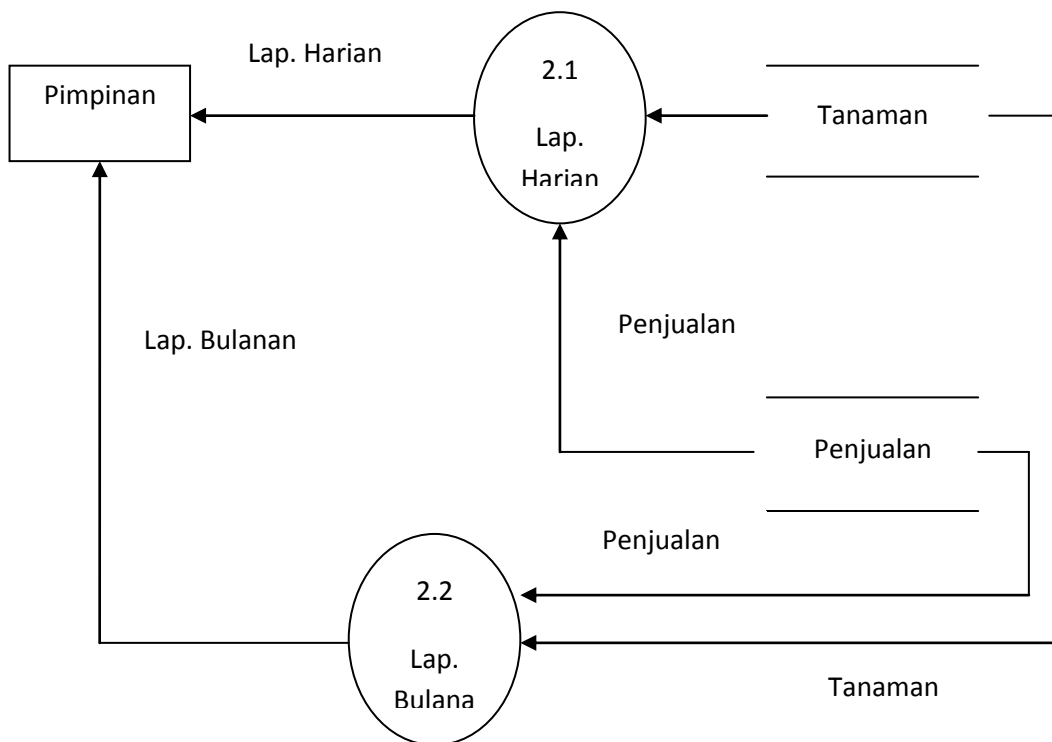


Gambar 2.10. Contoh diagram level 1

Keterangan :

DFD level rinci pertama yaitu DFD level 1 untuk sistem transaksi. Sistem transaksi dimulai dari pelanggan yang membeli tanaman tersebut, kemudian karyawan memberikan nota penjualan sebagai bukti pembelian tanaman tersebut kepada pelanggan. Lalu hasil dari transaksi tersebut di simpan dalam data base sistem.

2. Level Rinci Laporan



Gambar 2.11. Contoh diagram level 2

Keterangan :

DFD level 2 dari diagram level nol adalah sistem laporan. Sistem laporan didapat dari data yang telah disimpan dalam data base sistem akan memberikan laporan penjualan setiap bulan kepada pemimpin toko.

2.6 Entity – Relationship Diagram (ERD)

Entity relationship diagram (E-R) berisi komponen-komponen entitas dari himpunan relasi yang masing-masing dilengkapi atribut-atribut yang dapat digambarkan dengan lebih sistematis dengan menggunakan entity relationship (ERD). Entity relationship (E-R) didasarkan atas objek riil dunia nyata dan hubungan antar objek-objek dapat berupa entitas, atribut, domain dan relationship.

1. Entitas

Entitas adalah objek riil yang dapat dibedakan dengan lainnya dan tidak saling bergantung. Ada dua macam entitas yaitu : entitas yang bersifat fisik dan entitas yang bersifat konsep.

2. Atribut

Atribut menerangkan sebuah entitas.

3. Domain

Domain adalah kumpulan harga atau nilai yang dapat dimiliki oleh atribut dari suatu entitas.

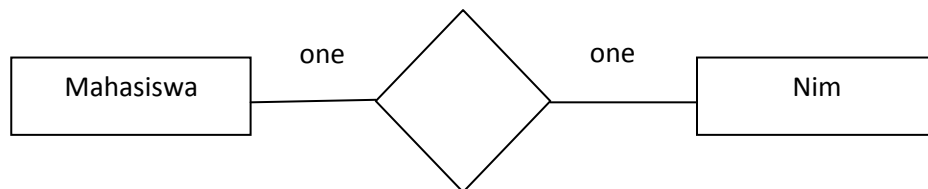
4. Tipe hubungan (relationship)

Tipe hubungan R di antara n buah entitas E_1, E_2, \dots, E_n adalah kumpulan dari relationship di antara entitas-entitas tersebut.

Batasan –batasan pada relationship :

a. Satu ke satu

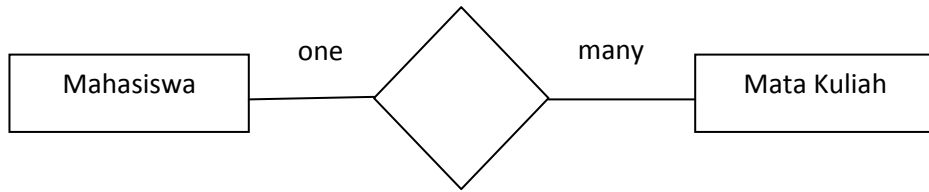
Yang berarti setiap entitas pada himpunan A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu sebaliknya setiap pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.



Gambar 2.12 Contoh hubungan satu ke satu

b. Satu ke banyak

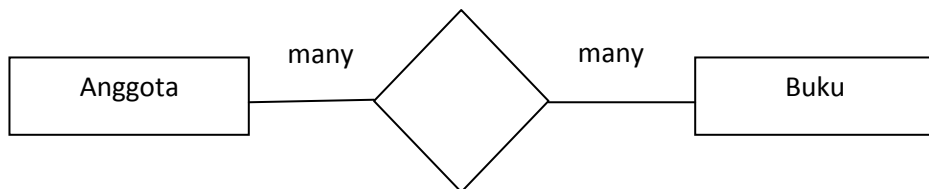
Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak satu entitas pada himpunan entitas A.



Gambar 2.13 Contoh hubungan satu ke banyak

c. Banyak ke banyak

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian sebaliknya dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.



Gambar 2.14 Contoh hubungan banyak ke banyak

2.7 Flowchart



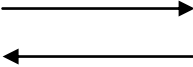
Bagan alir atau flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya.


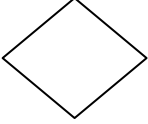

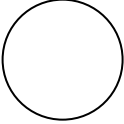
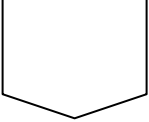

Tujuan utama dari penggunaan flowchart adalah menggambarkan suatu tahap penyelesaian masalah secara sederhana, teratur, rapi dan jelas dengan menggunakan

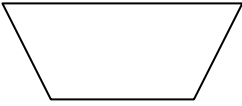
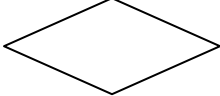


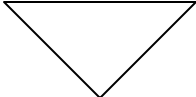
simbol-simbol standart. Tahap penyelesaian masalah yang disajikan jelas, sederhana, efektif dan tepat. Dalam penulisan flowchart dikenakan dua model yaitu sistem flowchart dan program flowchart.



2.7.1 Program Flowchart

Program flowchart merupakan diagram alir yang menggambarkan urutan logika dari suatu pemecahan masalah. Dalam menggambarkan program flowchart, telah tersedia simbol-simbol standart, tetapi seperti pada sistem flowchart, pemrograman dapat menambahkan simbol sendiri jika ternyata simbol-simbol standart yang tersedia kurang mencukupi. Berikut ini adalah simbol-simbol standart yang telah banyak digunakan pada penggambaran program flowchart :

Simbol	Keterangan
	Simbol inpu atau output ini digunakan untuk memasukan dan mengeluarkan.
	Simbol proses ini digunakan untuk operasi pengolahan data
	Simbol aliran data ini digunakan untuk penghubung atau aliran data.

	<p>Simbol terminal ini digunakan untuk awalan dan akhiran program.</p>
	<p>Simbol decision ini digunakan untuk kondisi percabangan.</p>
	<p>Simbol inisialisasi ini digunakan untuk mendefinisikan nama dan format-format yang digunakan dalam input dan output.</p>
	<p>Simbol conector ini digunakan untuk menghubungkan flowchart ke bagian yang diinginkan.</p>
	<p>Simbol off-page connector ini digunakan untuk menghubungkan flowchart dari halaman satu ke halaman yang lainnya.</p>
	<p>Menunjukkan proses pengolahan dan perubahan harga</p>

	<p>Manual operation, suatu simbol yang menunjukkan setiap pengolahan yang tidak dilakukan oleh komputer</p>
	<p>Decision, suatu kondisi yang akan menghasilkan kemungkinan jawaban atau aksi</p>
	<p>Predefined process, suatu simbol untuk menyediakan tempat pengolahan dalam storage</p>
	<p>Terminal, untuk memulai atau mengakhiri suatu program / juga sebagai interupsi dalam program</p>
	<p>Of line storage, simbol data yang berada dalam simbol ini akan disimpan</p>

	<p>Input-input setiap peralatan (1/0) dapat menggunakan symbol ini tanpa mengetahui jenis peralatan baik input atau output</p>
	<p>Dokumen simbol, untuk data yang berbentuk kertas atau informasi</p>

Gambar 2.15 Simbol-simbol flowchart

2.8 Bahasa Pemrograman Delphi

Delphi merupakan perangkat pengembangan aplikasi yang sangat terkenal di lingkungan Windows. Dengan menggunakan perangkat lunak ini, kita dapat membangun berbagai aplikasi Windows (permainan, multimedia, database, dan lain-lain) dengan cepat dan mudah. Dengan pendekatan visual kita dapat menciptakan aplikasi yang canggih dan pasti akan disukai oleh pemakai tanpa banyak menuliskan kode. Delphi menggunakan bahasa object pascal sebagai bahasa dasar, dan jika kita menguasai pascal, kita akan lebih mudah memahami program Delphi.

2.8.1 Program, Aplikasi Windows dan Form

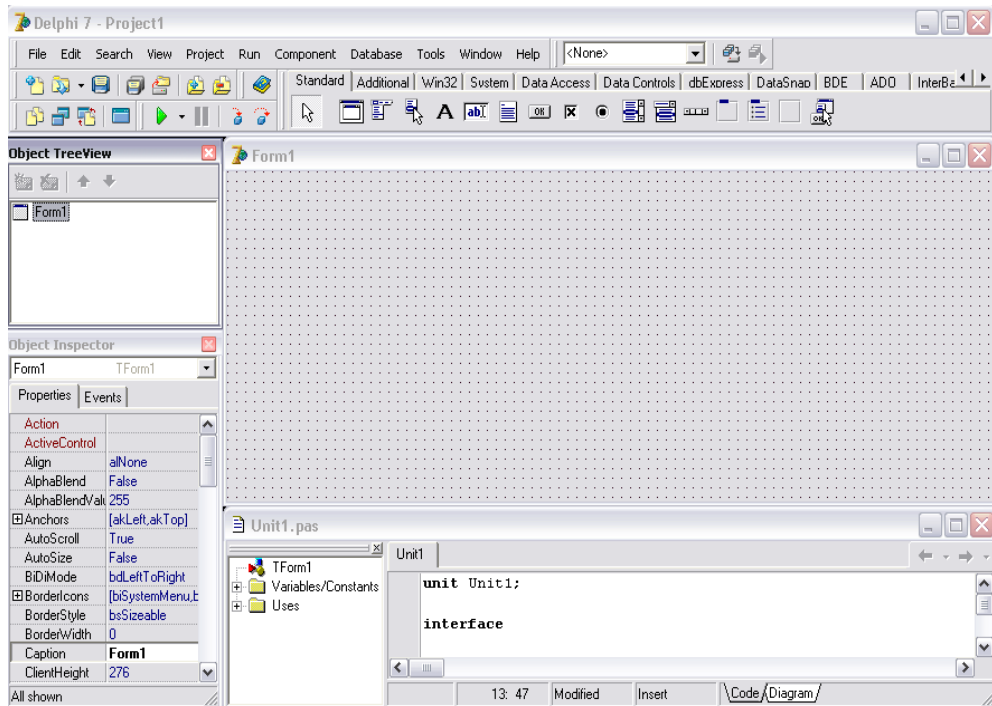
Program adalah sederetan kode yang digunakan untuk mengatur komputer agar melakukan sesuatu sesuai dengan keinginan pembuatnya. Program Aplikasi dapat dibedakan menjadi aplikasi windows dan aplikasi konsol.

Aplikasi windows adalah aplikasi yang berjalan dengan windows. Sedangkan aplikasi konsol yaitu aplikasi yang berjalan diluar windows, seperti halnya DOS. Secara umum, sebuah aplikasi paling tidak melibatkan sebuah form. Namun tentu saja aplikasi juga bisa melibatkan banyak form. Ketika dijalankan form akan berbentuk jendela. Oleh karena itu istilah form dan jendela seringkali dipertukarkan.

Sebuah form umumnya akan melibatkan banyak komponen lain, mengingat form itu sendiri tergolong sebagai komponen. Kotak kombo dan tombol radio merupakan contoh komponen. Namun perlu diketahui tidak semua komponen terlihat secara visual. Kotak kombo dan tombol radio adalah salah satu komponen yang terlihat secara visual, komponen yang terlihat secara visual disebut juga dengan kontrol. Pada Delphi, sebuah aplikasi akan diletakkan pada sebuah proyek. Sebuah proyek dapat membawahi sejumlah form.

2.8.2 Paket Komponen

Paket adalah jenis file DLL (Dynamic Link Libraries) yang digunakan oleh IDE (Integrated Development Environment) atau aplikasi atau berguna untuk pengaturan kode bersama antar aplikasi saat beberapa program dijalankan secara bersamaan. Adanya kode bersama ini akan mengurangi jumlah file eksekusi, sehingga mengurangi beban memori komputer. Hasil akhirnya adalah unjuk kerja program yang lebih baik. Librari komponen pada Delphi juga dikembangkan sehingga dapat melakukan kompilasi terhadap beberapa paket khusus tanpa harus mengkompilasi keseluruhan library. Paket khusus ini biasa dipakai pada tiap proyek, sehingga menghemat waktu dan tenaga. Semua paket Delphi diinstal pada IDE sebagai paket, yaitu file berekstensi DPL. Kelengkapan paket seperti Package Editor berikut kotak dialognya.



Gambar 2.16. Tampilan Program Delphi 7.0.

2.8.3 IDE (Integrated Development Environment)

Tampilan bidang kerja di Delphi lazim disebut dengan IDE (Integrated Development Environment). Pada Delphi 7.0 bisa kita lihat pada Gambar 3.1. di atas. Secara garis besar terdiri atas empat bagian yaitu Window utama, Object TreeView, Object Inspector, dan Editor. Jendela utama terdiri atas Menu bar, Tool bar dan Component palette. Object inspector menyediakan dua kelompok pengaturan komponen, yaitu properties dan event. Editor yang disediakan ada dua buah yaitu Form editor dan Code Editor. Form Editor sering disebut dengan editor saja.

2.8.4 Menu Bar

Menu Bar di Delphi 7.0 menyediakan kelompok perintah yang digolongkan dalam sebelas menu. Menu tersebut antara lain File, Edit, Search, View, Project, Run, Component, Database, Tools, Window, dan Help. Diantara menu-menu Bar tersebut mempunyai fungsi sebagai berikut :

1. File adalah kelompok perintah yang berfungsi untuk pengaturan suatu file. Misalnya New, Save, Open dan sebagainya.
2. Edit adalah menu yang menyimpan perintah-perintah untuk pengeditan. Mulai dari pengeditan objek, pengeditan komponen maupun pengeditan kode pada kode editor. Misalannya: Cut, Paste, Align.
3. Search adalah menu yang berfungsi untuk mencari file, kata pada listing program, dan sebagainya.
4. View yaitu menu yang menampung perintah-perintah untuk mengaktifkan bagian pendukung IDE (Integrated Development Environment).
5. Project adalah kelompok perintah yang berfungsi untuk manajemen proyek berikut bagian-bagian pendukungnya.

6. Run yaitu menu yang merupakan kelompok perintah yang menangani proses kompilasi program, seperti Run, Build, Step Over, Debug dan seterusnya.
7. Component, jika kita ingin mengatur suatu komponen, gunakanlah menu komponen ini sebab mengandung perintah-perintah yang berfungsi untuk pengaturan komponen, mulai dari pemasangan, palette, pembuatan komponen, penghapusan komponen dan sebagainya.
8. Database, seluruh pekerjaan pemrograman yang berhubungan database harus menggunakan menu ini, sebab mengandung perintah-perintah untuk pengaturan aplikasi database.
9. Tools yaitu kelompok perintah yang berfungsi sebagai penyedia perlengkapan tambahan yang diperlukan dalam penyusunan program seperti Image editor, Database Engine, dan lain-lain.
10. Windows yaitu kelompok perintah yang berfungsi sebagai bantuan untuk berpindah dari satu jendela ke jendela yang ada di Delphi 7.0
11. Help yaitu menu yang berfungsi memberikan informasi yang bersifat menolong pemakai dalam menggunakan Delphi.

BAB III

ANALISA DAN PERANCANGAN SISTEM

3.1 Perancangan Sistem

Dari apa yang akan dikerjakan pada perancangan sistem ini, warna pada apa yang akan dibuat berdasarkan citra warna dengan pencarian hasil berdasarkan warna RGB dengan mengimplemantasikan sebuah warna dengan uang kertas yang berlaku baik itu uang yang masih baru maupun yang sdh lama. Dengan menggunakan metode kesamaan warna yang ada dalam uang kertas tersebut. Dengan adanya perbedaan warna dalam uang maka akan mendeteksi kemungkinan bahwa apa yang dimaksud adalah uang dengan nilai nominal yang sama dengan apa yang ada di gambar. Dengan mengambil gambar di salah satu sisi uang, yang diambil adalah sisi dimana gambar dan pola warna yang sudah jelas. Dengan mendeteksi warnanya berdasarkan RGB pada uang yang sudah tersimpan maka dapat menyimpulkan apakah data yang keluar benar apa yang dimaksudkan dengan gambar mata uang tersebut.

Ciri merupakan suatu tanda yang khas, yang akan membedakan antara satu dengan yang lain. Tidak berbeda dengan sebuah gambar, gambar juga memiliki ciri yang dapat membedakannya dengan gambar yang lain. Masing- masing ciri gambar didapatkan dari proses penjumlahan total dari hasil pencarian R(red),

G(green) dan B(blue). warna merah (R), hijau(G) dan biru(B) merupakan warna pokok dalam pengelolaan gambar. Jika warna-warna tersebut digabungkan maka akan menghasilkan warna lain.

Color histogram merupakan hubungan dari intensitas tiga macam warna di mana setiap gambar mempunyai distribusi warna ini. Distribusi warna di modelkan dengan color histogram. Color histogram di hitung dengan cara mendiskretkan warna dalam gambar dan menghitung jumlah dari tiap-tiap pixel pada gambar.

Untuk mengidentifikasi gambar uang digunakan metode sebagai berikut:

1. Menghitung hasil penjumlahan dari warna merah, hijau dan biru pada bagian gambar kemudian akan di bandingkan antara gambar pola dengan gambar query.
2. Proses matching untuk mendapatkan gambar pola yang memiliki jarak terdekat dengan gambar query. Gambar yang memiliki jarak terdekat adalah gambar yang semua atau yang paling mirip.

Hal pertama yang dilakukan adalah mengambil gambar uang yang akan dijadikan pola dan query. Pengambilan gambar dilakkukan dengan ketentuan sebagai berikut:

1. Gambar diambil dengan scanner canon

2. Tipe gambar jpg dengan ukuran 720 x 360
3. Diambil pada posisi yang sama background gambar putih.
4. Gambar uang yang dijadikan objek penelitian adalah uang 1000, 2000, 5000, 10000, 20000, 50000 dan 100000.
5. Gambar uang yang diambil hanya satu sisi saja , sehingga analisa gambar nantinya hanya dilakukan pada satu sisi saja. Sisi gambar yang dipilih adalah sisi yang tidak ada nomor serinya. Karena pada sisi tersebut terdapat perbedaan warna yang mencolok pada bagian pojok kiri atas. Bagian tersebut nantinya digunakan untuk segmentasi gambar.

3.2 Extraksi Ciri

Proses ekstraksi ciri warna dengan menggunakan histogram warna dengan menghitung rata-rata warna merah, hijau dan biru pada keseluruhan pixel gambar.

$$\text{Ratr} = (\sum \text{Totr})$$

$$\text{Ratg} = (\sum \text{Totg})$$

$$\text{Ratb} = (\sum \text{Totb})$$

Keterangan:

Ratr : Nilai rata-rata warna merah.

Totr : Total warna merah.

Ratg : Nilai rata-rata warna hijau.

Totg : Total warna hijau.

Ratb : Nilai rata-rata warna biru.

Totb : Total warna biru

Hasil ekstrasi ciri disimpan dalam database dengan file sebagai berikut:

Field	Type	Keterangan
Uang	Text	Untuk menyimpan jenis uang
R	Number	Untuk menyimpan rata-rata warna merah
G	Number	Untuk menyimpan rata-rata warna hijau
B	Number	Untuk menyimpan rata-rata warna biru

3.3 Matching

Matching adalah proses mencocokkan data antara gambar query dengan data pola yang telah tersimpan dalam database. Sebelumnya, gambar query juga dilakukan ekstrasi ciri yang sama seperti pada gambar pola. Sehingga gambar query akan memberikan nilai rata-rata warna merah, hijau dan biru untuk dibandingkan dengan database.

Untuk proses matching, diberikan toleransi x_1 . Sehingga data gambar yang memiliki nilai $x - x_1 \geq x \leq x + x_1$ akan dianggap gambar yang paling mirip. Jika tidak ditemukan data yang sesuai, maka akan dianggap tidak ada yang mirip.

3.4 Implementasi

Implementasi program menggunakan Microsoft Access 2003. Adapun struktur program terdiri dari dua bagian :

1. Input Data pola : digunakan untuk mengekstraksi ciri dan menyimpan data gambar yang digunakan sebagai pola.
2. Matching : digunakan untuk mengekstraksi ciri gambar query dan pencarian data dari database yang sesuai dengan hasil ekstraksi ciri gambar query.

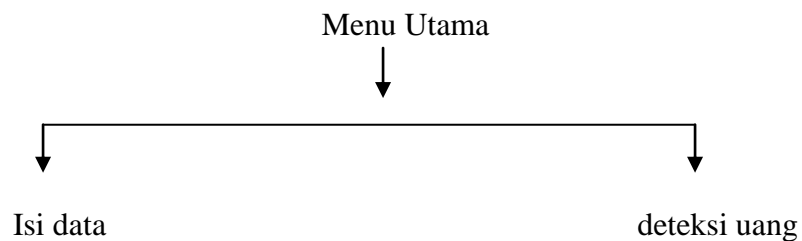
Percobaan yang dilakukan pada penelitian ini menggunakan 15 gambar, enam gambar digunakan sebagai gambar pola, sisinya digunakan untuk gambar query yang digunakan untuk proses matching.

Setelah tahap analisa, tahap berikutnya adalah tahap perancangan. Pada tahap ini akan memberikan sedikit gambaran tentang pencocokan gambar uang yang akan dibuat nantinya akan di kembangkan pada bab selanjutnya. Pada tahap ini akan di jabarkan tampilan menu, DFD, dari program informasi tentang pendektasian mata uang.

Adapun perancangan dari program ini adalah:

1. State Transaction Diagram

Di dalam perancangan sebuah aplikasi diperlukan adanya state transaction diagram yaitu suatu aliran info yang dapat memperjelas bagaimana urutan program itu beroperasi. Berikut tampilan state transaction diagram.



Keterangan :

1. Menu utama

Pada menu utama akan ditampilkan tombol-tombol (botton). Menu utama merupakan suatu form yang menghubungkan dengan menu lainnya sesuai dengan pilihan yang kita pilih.

2. Isi data

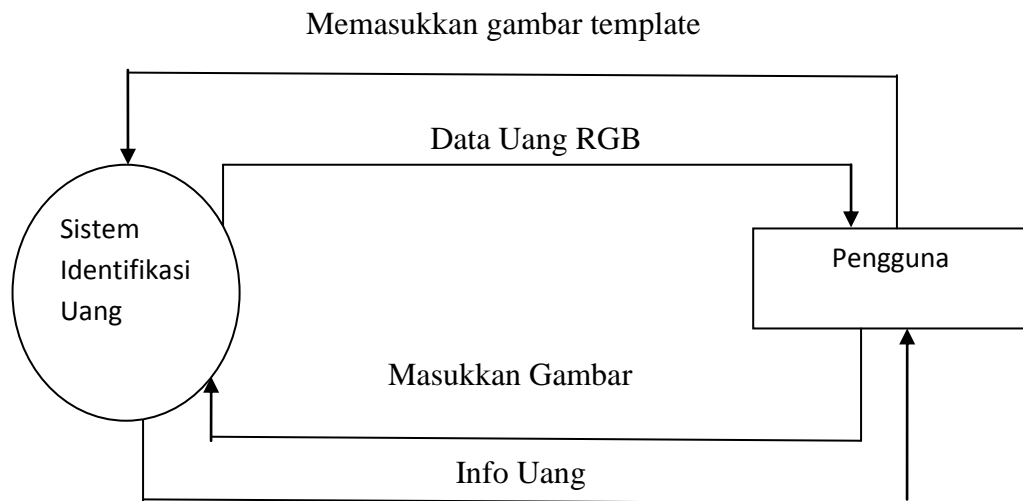
Berisikan data-data yang telah di isikan sebelumnya yg selanjut akan di identifikasikan.

3. Deteksi uang

Berisikan data yang telah tersimpan dan akan menampilkan hasil dari data tersebut.

3.5 DFD

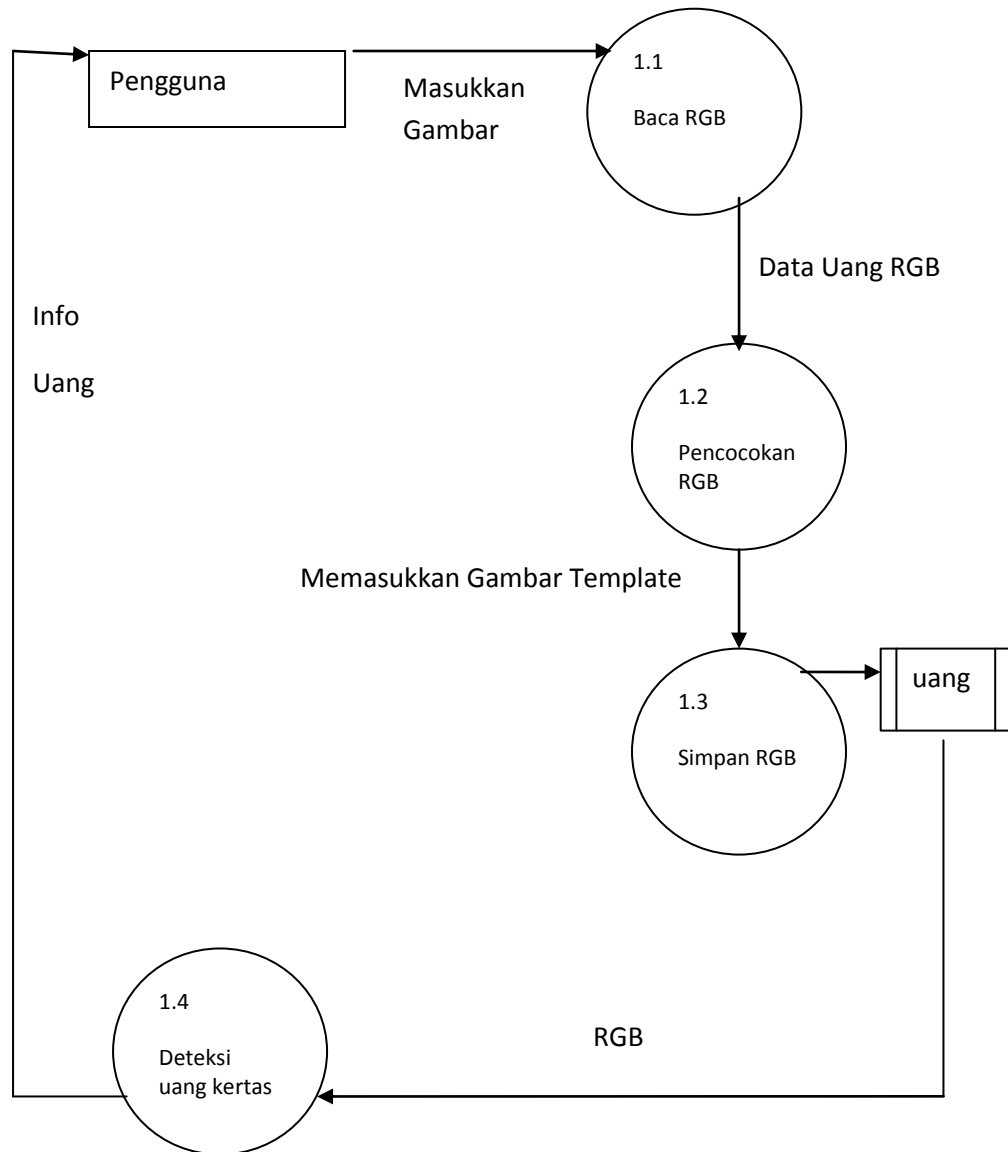
Dalam perancangan suatu program juga memerlukan adanya DFD yang berguna untuk mengetahui jalannya program tersebut.



Gambar 3.1 Context Diagram

Keterangan :

User memasukkan gambar kedalam sistem indentifikasi uang, yang kemudian akan diolah dan memberikan data uang berupa data RGB kepada user. Kemudian user memasukkan gambar yang berupa template ke sistem, memberikan info kepada user berupa hasilnya.



Gambar 3.2. DFD Level 1

Keterangan :

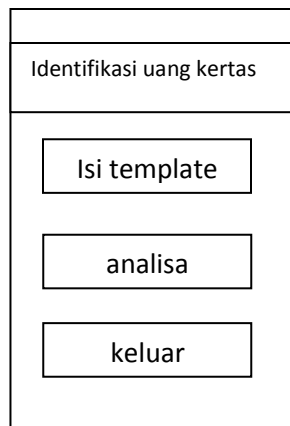
User memasukkan gambar untuk dibaca nilai RGBnya kemudian akan menjadi template dan dicocokkan RGB, kemudian hasil dari RGB tersebut akan disimpan

yang untuk dicocokkan dengan deteksi uang kertas, kemudian hasilnya akan mengirimkan nilai uang ke user.

3.6 Perancangan Form

3.6.1 Menu utama

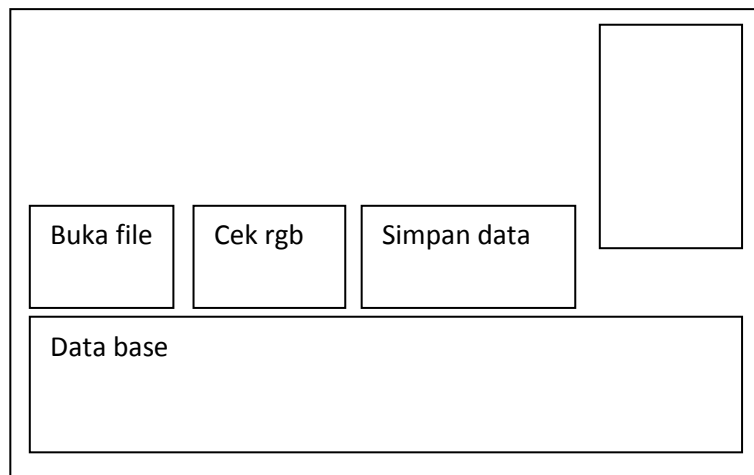
Form menu utama adalah form dimana langkah awal untuk masuk kedalam program identifikasi uang. Didalam form ini terdapat isi template untuk masuk kedalam sistem penyimpanan uang template. Dan label analisa untuk masuk kedalam sistem menganalisa akan uang yang akan di analisis.



Gambar 3.3 Rancangan Menu Utama

3.6.2 Sub menu isi template

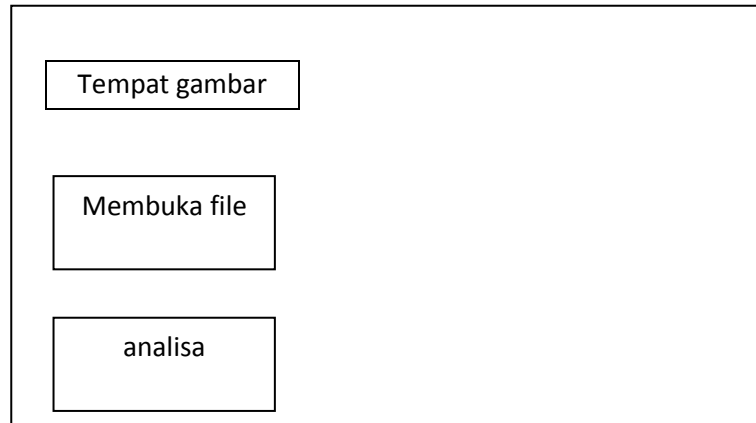
Form menu isi template merupakan desain yang isinya untuk membuka file yang dimana file tersebut berisi uang kertas yang akan dipilih untuk dijadikan template, kemudian di cek akan nilai RGBnya berdasarkan warna yang ada di uang kertas tersebut. Setelah itu di simpan dalam database.



Gambar 3.4 Rancangan Isi Template

3.6.3 Sub menu analisa

Form menu analisa berisi akan data yang akan di cek akan kebenaran dari uang tersebut.



Gambar 3.5 Rancangan analisa

3.7 Flowchart

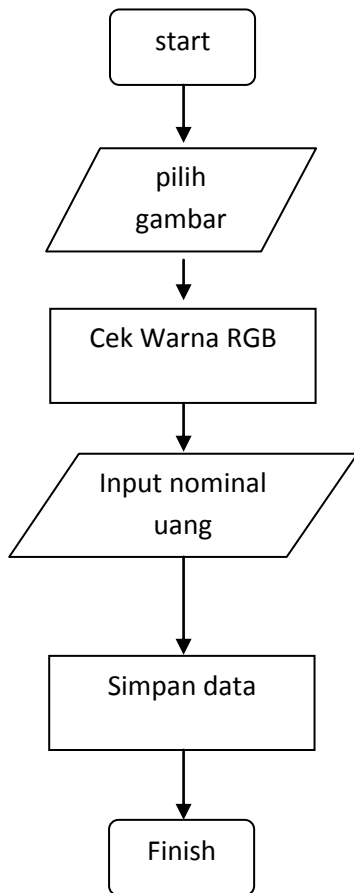
Dalam perancangan suatu program juga diperlukan adanya flowchart yang berguna untuk mengetahui jalannya program tersebut. Dengan adanya flowchart ini maka programmer akan dengan mudah dalam pembuatan suatu program karena sudah mengetahui alur program tersebut. Adapun beberapa gambaran flowchart pada tiap form yaitu :

1. Isi Template

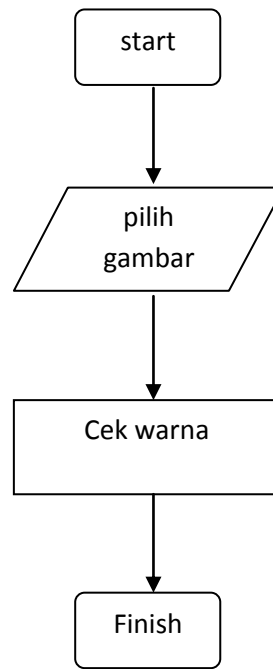
Bagian ini berisikan cek RGB serta menyimpan data yang sudah dimasukkan. Dan sebagai data template.

2. Analisa

Pada analisa ini akan menampilkan hasil dari menganalisa gambar.



Gambar 3.6 Flowchart Isi Template



Gambar 3.7 Flowchart Analisa

BAB IV

IMPLEMENTASI SISTEM

Rencana implementasi merupakan rencana meletakkan sistem supaya siap untuk dioperasikan dan supaya kegiatan implementasi nantinya dapat beroperasi sesuai dengan yang diharapkan, maka suatu rencana implementasi perlu dibuat terlebih dahulu.

Pada tahapan ini diperlukan beberapa peralatan atau fasilitas pendukung pekerjaan sistem. Karena sistem baru ini berbasis komputer, maka semua kegiatan pengolahan data harus ada keterkaitan antara 3 unsur pokok yaitu:

4.1 Perangkat Keras (Hardware)

Adapun spesifikasi standar minimal kebutuhan perangkat keras yang akan dibutuhkan untuk menjalankan aplikasi adalah sebagai berikut:

1. Processor intel pentium III 800 Mhz.
2. Menggunakan RAM minimal 512 MB
3. Harddisk yang digunakan untuk menjalankan dan menyimpan data program
4. Monitor
5. Printer scanner

4.2 Perangkat Lunak (Software)

Perangkat lunak atau software yang digunakan dalam pembuatan identifikasi warna ini yaitu:

a. Perangkat lunak aplikasi (Application software)

Program yang ditulis dengan diterjemahkan oleh language software untuk aplikasi tertentu dan untuk membantu pemakai komputer untuk menyelesaikan pekerjaannya dengan menggunakan delphi 7.0

b. Sistem operasi (Operating System)

Sutau sistem operasi yang terdiri dari kumpulan program-program sistem yang terorganisasi dan dibuat untuk membantu menghubungkan antara perangkat keras komputer dengan perangkat lunak yang dibuat oleh pemakai. Operating system yang digunakan windows XP Proffesional.

4.3 Pemilihin Brainware

Brainware sering juga disebut pemakai komputer atau orang yang menjalankan peralatan komputer. Ada beberapa istilah yang berhubungan dengan brainware, diantaranya adalah operator brainware harus memiliki keahlian dalm menjalankan instruksi-instruksi program, karena dengan tenaga brainware yang handal kemungkinan sistem baru yang diterapkan akan sesuai untuk

meningkatkan kinerja pada suatu perusahaan. Agar sistem berjalan dengan baik, brainware diharapkan memenuhi syarat sebagai berikut:

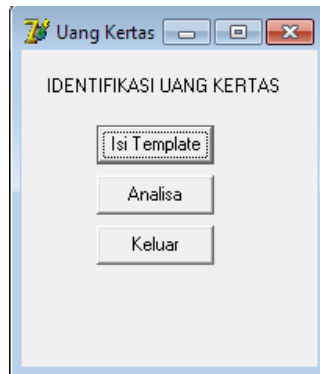
1. Orang yang memiliki pengetahuan dan penggunaan komputer yang memadai sehingga dapat menggunakan sistem yang baru dengan baik dan benar.
2. Mempunyai tanggung jawab terhadap sistem yang ada.
3. Mampu bertindak profesional dalam pengambilan keputusan.

4.4 Tampilan Program

Dalam tampilan program akan dijelaskan tentang rancang bangun identifikasi nominal citra uang kertas indonesia berdasarkan RGB.

4.4.1 Tampilan Form Menu Utama

Pada menu form utama terdapat 3 menu utama yaitu isi template, analisa dan keluar. Sedangkan pada menu isi template terdapat sub menu buka file, cek RGB dan simpan analisa. Sedangkan pada menu analisa terdapat sub menu buka file dan analisa. Tampilan form menu utama adalah sebagai berikut:



Gambar 4.1 Form Menu Utama

object Form2: TForm2

Left = 163

Top = 163

Width = 185

Height = 195

Caption = 'Uang Kertas'

Color = clBtnFace

Font.Charset = DEFAULT_CHARSET

Font.Color = clWindowText

Font.Height = -11

Font.Name = 'MS Sans Serif'

Font.Style = []

OldCreateOrder = False

PixelsPerInch = 96

TextHeight = 13

object Label1: TLabel

Left = 16

Top = 16

Width = 149

Height = 13

Caption = 'IDENTIFIKASI UANG KERTAS'

end

object Button1: TButton

Left = 48

Top = 48

Width = 75

Height = 25

```
Caption = 'Isi Template'  
  
TabOrder = 0  
  
OnClick = Button1Click
```

```
end
```

```
object Button2: TButton
```

```
Left = 48
```

```
Top = 80
```

```
Width = 75
```

```
Height = 25
```

```
Caption = 'Analisa'
```

```
TabOrder = 1
```

```
OnClick = Button2Click
```

```
end
```

```
object Button3: TButton
```

```
Left = 48
```

```
Top = 112
```


Width = 75

Height = 25

Caption = 'Keluar'

TabOrder = 2

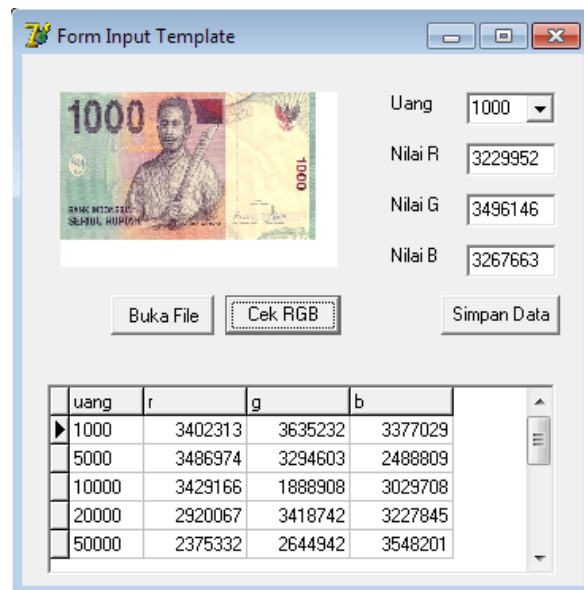
OnClick = Button3Click

end

end

4.4.2 Tampilan Form Isi Template

Merupakan form untuk memasukkan data uang kertas yang akan di cek warna berdasarkan RGB nya. Dan hasilnya akan bisa disimpan dalam data. Tampilan menu isi template dapat dilihat pada gambar berikut:



Gambar 4.2 Tampilan Form Input Template

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
ColNumb : TColor;
```

```
Color: $0..$FFFFFFF;
```

```
R, G, B,i,j: Byte;
```

```

TR, TG, TB : integer;

begin

TR:=0;TG:=0;TB:=0;

for i:= 1 to Image1.Height do

begin

for j:= 1 to Image1.Width do

begin

ColNumb := Image1.Canvas.Pixels[i, j];

Color := ColorToRGB(ColNumb);

R := ($000000FF and Color);

G := ($0000FF00 and Color) Shr 8;

B := ($00FF0000 and Color) Shr 16;

TR:=TR + R;

TG:=TG + G;

TB:=TB + B;

end;

end;

end;

```

```
end;

edit1.Text := IntToStr(TR);

edit2.Text := IntToStr(TG);

edit3.Text := IntToStr(TB);

end;

procedure TForm1.SpeedButton1Click(Sender: TObject);

begin

Image1.Width:=175;

Image1.Height:=110;

if (OpenPictureDialog1.Execute)then

Image1.Picture.LoadFromFile(OpenPictureDialog1.FileName);

end;

procedure TForm1.Button2Click(Sender: TObject);

begin

adotable1.Insert;
```

```
adotable1.FieldValues['uang']:= combobox1.Text;
```

```
adotable1.FieldValues['r']:= edit1.Text;
```

```
adotable1.FieldValues['g']:= edit2.Text;
```

```
adotable1.FieldValues['b']:= edit3.Text;
```

```
adotable1.Post;
```

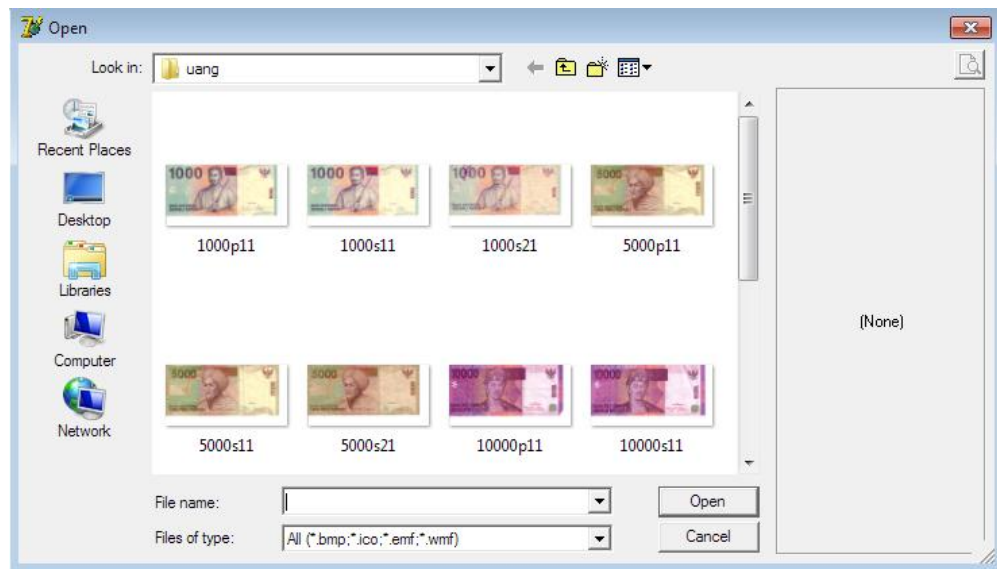
```
end;
```

```
end.
```

4.4.3 Tampilan open file

Pada saat masuk ke menu input template diminta untuk membuka file yang berisi deretan pilihan gambar mata uang yang sebagai inputan.

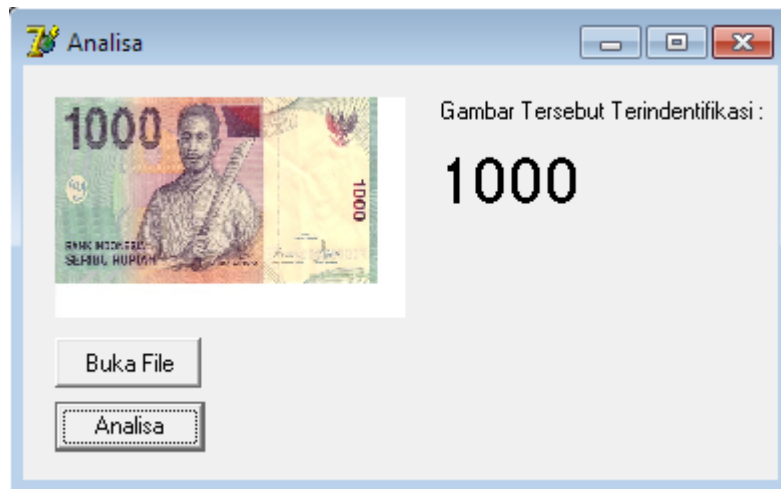
Tampilannya seperti berikut:



Gambar 4.3 tampilan open file

4.4.4 Tampilan Form Analisa

Pada form analisa akan menampilkan hasil dari gambar yang akan dianalisa dan gambar yang teridentifikasi. Tampilan form analisa dapat dilihat pada gambar berikut:



Gambar 4.4 Tampilan Form Analisa

```
unit Unit3;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms,
```

```
Dialogs, StdCtrls, ExtCtrls, ExtDlgs, DB, ADODB, DBCtrls, Buttons;
```

type

TForm3 = class(TForm)

Image1: TImage;

Button2: TButton;

Label1: TLabel;

OpenPictureDialog1: TOpenPictureDialog;

ADOConnection1: TADOConnection;

ADOQuery1: TADOQuery;

Label2: TLabel;

SpeedButton1: TSpeedButton;

procedure Button2Click(Sender: TObject);

procedure SpeedButton1Click(Sender: TObject);

private

{ Private declarations }

public


```

    { Public declarations }

end;

var

    Form3: TForm3;

implementation

    {$R *.dfm}

    procedure TForm3.Button2Click(Sender: TObject);

    var

        ColNumb : TColor;

        Color: $0..$FFFFFFF;

        R, G, B,i,j: Byte;

        TR, TG, TB : integer;

        hasilanalisa:string;

    begin

        TR:=0;TG:=0;TB:=0;

        for i:= 1 to Image1.Height do

```

```

begin

for j:= 1 to Image1.Width do

begin

ColNumb := Image1.Canvas.Pixels[i, j];

Color := ColorToRGB(ColNumb);

R := ($000000FF and Color);

G := ($0000FF00 and Color) Shr 8;

B := ($00FF0000 and Color) Shr 16;

TR:=TR + R;

TG:=TG + G;

TB:=TB + B;

end;

end;

with ADOQuery1 do begin

Close;

SQL.Clear;

```

```

SQL.Add('SELECT * FROM uang ');

SQL.Add('where r>=:ptr and r<=:ptr2 and g>=:ptg and g<=:ptg2 and
        b>=:ptb and b<=:ptb2');

Parameters.ParamByName('ptr').Value:=TR-200000;

Parameters.ParamByName('ptr2').Value:=TR+200000;

Parameters.ParamByName('ptg').Value:=TG-200000;

Parameters.ParamByName('ptg2').Value:=TG+200000;

Parameters.ParamByName('ptb').Value:=TB-200000;

Parameters.ParamByName('ptb2').Value:=TB+200000;

Open;

//label2.Caption:=adoquery1.FieldValues['uang'];

//hasilanalisa :=adoquery1.FieldValues['uang'];

if (ADOQuery1.IsEmpty) then

label2.caption := 'Uang Tidak Dikenal'

else

label2.caption := adoquery1.FieldValues['uang'];

```

end;

end;

procedure TForm3.SpeedButton1Click(Sender: TObject);

begin

Image1.Width:=175;

Image1.Height:=110;

if (OpenPictureDialog1.Execute)then

Image1.Picture.LoadFromFile(OpenPictureDialog1.FileName);

end;

end.

BAB V

KESIMPULAN DAN SARAN

Dari penjelasan yang telah dikemukakan pada Bab I sampai Bab IV mengenai identifikasi nominal citra uang kertas Indonesia ini penulis mengambil kesimpulan serta saran sebagaimana diuraikan di bawah ini :

5.1 Kesimpulan

Dari pembahasan yang telah dijelaskan pada bab-bab sebelumnya maka dalam pembuatan identifikasi nominal uang kertas Indonesia dapat diambil kesimpulan sebagai berikut :

1. Aplikasi rancang bangun identifikasi nominal citra uang kertas Indonesia berdasarkan RGB. Adalah software yang dapat digunakan untuk mengidentifikasi uang kertas yang berdasarkan warna.
2. Ciri warna dapat dijadikan acuan dalam menentukan nominal uang kertas.
3. Agar aplikasi dapat mengenali uang kertas dengan benar dapat menggunakan template yang akan membandingkan dengan uang yang akan diidentifikasi.

5.2 Saran

Saran yang bisa diberikan untuk pengembangan sistem ini adalah sebagai berikut:

1. Sebagai aplikasi alternatif, semoga pada tahap berikutnya bentuk tampilan dapat dibuat lebih menarik.
3. Program aplikasi identifikasi uang kertas indonesia ini masih sederhana dan jauh dari sempurna, jadi diharapkan aplikasi ini nantinya dapat dikembangkan agar menjadi lebih baik lagi.
4. Program identifikasi nominal citra uang kertas ini dapat dikembangkan lagi ke tingkat stadium lanjut.

DAFTAR PUSTAKA

Fadlisyah, Taufiq, Zulfikar, Fauzan. (2008), “Pengolahan Citra Menggunakan Delphi”, Graha Ilmu Yogyakarta.

Munir, Rinaldi. (2004), “Pengolahan Citra Digital dengan Pendekatan Algoritmik”, Informatika Bandung.

Ahmad, Usman. (2005), “Pengolahan Citra Digital & Teknik Pemrogramannya”, Graha Ilmu Yogyakarta.

Zukhri, Zainudin. (2003), “Dasar-dasar Pemrograman Visual dengan Delphi 6.0”, Graha Ilmu Yogyakarta.

LAMPIRAN

Form Utama

object Form2: TForm2

Left = 163

Top = 163

Width = 185

Height = 195

Caption = 'Uang Kertas'

Color = clBtnFace

Font.Charset = DEFAULT_CHARSET

Font.Color = clWindowText

Font.Height = -11

Font.Name = 'MS Sans Serif'

Font.Style = []

OldCreateOrder = False

PixelsPerInch = 96

TextHeight = 13

object Label1: TLabel

Left = 16

Top = 16

Width = 149

Height = 13

Caption = 'IDENTIFIKASI UANG KERTAS'

End

object Button1: TButton

Left = 48

Top = 48

Width = 75

Height = 25

Caption = 'Isi Template'

TabOrder = 0

OnClick = Button1Click

End

object Button2: TButton

Left = 48

Top = 80

Width = 75

Height = 25

Caption = 'Analisa'

```
    TabOrder = 1  
    OnClick = Button2Click  
End
```

```
object Button3: TButton  
    Left = 48  
    Top = 112  
    Width = 75  
    Height = 25  
    Caption = 'Keluar'  
    TabOrder = 2  
    OnClick = Button3Click  
End
```

```
End
```

Form Template

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
```

```
Dialogs, StdCtrls, ExtCtrls, Buttons, ExtDlgs, DB, ADODB, Grids, DBGrids,
```

```
DBCtrls;
```

```
type
```

```
TForm1 = class(TForm)
```

```
Image1: TImage;
```

```
Button1: TButton;
```

```
Edit1: TEdit;
```

Edit2: TEdit;

Edit3: TEdit;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

SpeedButton1: TSpeedButton;

OpenPictureDialog1: TOpenPictureDialog;

Button2: TButton;

Label4: TLabel;

ComboBox1: TComboBox;

ADOTable1: TADOTable;

DBGrid1: TDBGrid;

DataSource1: TDataSource;

ADOConnection1: TADOConnection;

```
DBNavigator1: TDBNavigator;
```

```
procedure Button1Click(Sender: TObject);
```

```
procedure SpeedButton1Click(Sender: TObject);
```

```
procedure Button2Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form1: TForm1;
```

```
implementation
```

```
{ $R *.dfm }
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
ColNumb : TColor;
```

```
Color: $0..$FFFFFFFF;
```

```
R, G, B,i,j: Byte;
```

```
TR, TG, TB : integer;
```

```
begin
```

```
//hitung rgb
```

```
TR:=0;TG:=0;TB:=0;
```

```
for i:= 1 to Image1.Height do
```

```
begin

for j:= 1 to Image1.Width do

begin

ColNumb := Image1.Canvas.Pixels[i, j];

Color := ColorToRGB(ColNumb);

R := ($000000FF and Color);

G := ($0000FF00 and Color) Shr 8;

B := ($00FF0000 and Color) Shr 16;

TR:=TR + R;

TG:=TG + G;

TB:=TB + B;

end;

end;

edit1.Text := IntToStr(TR);
```

```
edit2.Text := IntToStr(TG);
```

```
edit3.Text := IntToStr(TB);
```

```
end;
```

```
procedure TForm1.SpeedButton1Click(Sender: TObject);
```

```
begin
```

```
Image1.Width:=175;
```

```
Image1.Height:=110;
```

```
//open file
```

```
if (OpenPictureDialog1.Execute)then
```

```
Image1.Picture.LoadFromFile(OpenPictureDialog1.FileName);
```

```
end;
```



```
procedure TForm1.Button2Click(Sender: TObject);

begin

//isi database

adotable1.Insert;

adotable1.FieldValues['uang']:= combobox1.Text;

adotable1.FieldValues['r']:= edit1.Text;

adotable1.FieldValues['g']:= edit2.Text;

adotable1.FieldValues['b']:= edit3.Text;

adotable1.Post;

end;

end.
```

Form Analisa

```
unit Unit3;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
```

```
Dialogs, StdCtrls, ExtCtrls, ExtDlgs, DB, ADODB, DBCtrls, Buttons;
```

```
type
```

```
TForm3 = class(TForm)
```

```
Image1: TImage;
```

```
Button2: TButton;
```

```
Label1: TLabel;
```

OpenPictureDialog1: TOpenPictureDialog;

ADOConnection1: TADOConnection;

ADOQuery1: TADOQuery;

Label2: TLabel;

SpeedButton1: TSpeedButton;

procedure Button2Click(Sender: TObject);

procedure SpeedButton1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

```
Form3: TForm3;
```

```
implementation
```

```
{ $R *.dfm }
```

```
procedure TForm3.Button2Click(Sender: TObject);
```

```
var
```

```
ColNumb : TColor;
```

```
Color: $0..$FFFFFFF;
```

```
R, G, B,i,j: Byte;
```

```
TR, TG, TB : integer;
```

```
hasilanalisa:string;
```

```
begin
```

```
TR:=0;TG:=0;TB:=0;
```

```

for i:= 1 to Image1.Height do

begin

for j:= 1 to Image1.Width do

begin

ColNumb := Image1.Canvas.Pixels[i, j];

Color := ColorToRGB(ColNumb);

R := ($000000FF and Color);

G := ($0000FF00 and Color) Shr 8;

B := ($00FF0000 and Color) Shr 16;

TR:=TR + R;

TG:=TG + G;

TB:=TB + B;

end;

end;

//bandingkan dengan database

with ADOQuery1 do begin

```

Close;

SQL.Clear;

SQL.Add('SELECT * FROM uang');

SQL.Add('where r>=:ptr and r<=:ptr2 and g>=:ptg and g<=:ptg2 and b>=:ptb
and b<=:ptb2');

Parameters.ParamByName('ptr').Value:=TR-200000;

Parameters.ParamByName('ptr2').Value:=TR+200000;

Parameters.ParamByName('ptg').Value:=TG-200000;

Parameters.ParamByName('ptg2').Value:=TG+200000;

Parameters.ParamByName('ptb').Value:=TB-190000;

Parameters.ParamByName('ptb2').Value:=TB+190000;

Open;

//label2.Caption:=adoquery1.FieldValues['uang'];

```
//hasilanalisa :=adoquery1.FieldValues['uang'];

if (ADOQuery1.IsEmpty) then

label2.caption := 'Uang Tidak Dikenal'

else

label2.caption := adoquery1.FieldValues['uang'];

end;

end;

procedure TForm3.SpeedButton1Click(Sender: TObject);

begin

Image1.Width:=175;

Image1.Height:=110;

if (OpenPictureDialog1.Execute)then

Image1.Picture.LoadFromFile(OpenPictureDialog1.FileName);

end;

end.
```