

**MONITORING GERAKAN PADA RUANGAN
MENGUNAKAN WEBCAM DAN MOTOR STEPPER**

SKRIPSI

Oleh:

IBNU CATUR MUSTOFA

NIM: 04550051



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MALANG
2008**

**MONITORING GERAKAN PADA RUANGAN
MENGUNAKAN WEBCAM DAN MOTOR STEPPER**

SKRIPSI

**Diajukan Kepada:
Dekan Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

Oleh:

**IBNU CATUR MUSTOFA
NIM: 04550051**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MALANG
2008**

MONITORING GERAKAN PADA RUANGAN MENGUNAKAN WEBCAM DAN MOTOR STEPPER

Oleh:

IBNU CATUR MUSTOFA

NIM: 04550051

Telah Disetujui untuk Diuji

Malang, 1 Oktober 2008

Dosen Pembimbing I,

Fatchurrochman, M.Kom.

NIP 150 368 774

Dosen Pembimbing II,

Ahmad Barizi, M.A.

NIP 150 283 991

Mengetahui,

Ketua Jurusan Teknik Informatika

Suhartono, S.Si, M.Kom.

NIP 150 327 241

LEMBAR PENGESAHAN

**MONITORING GERAKAN PADA RUANGAN
MENGUNAKAN WEBCAM DAN MOTOR STEPPER**

SKRIPSI

OLEH
IBNU CATUR MUSTOFA
NIM 04550051

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 18 Oktober 2008

Susunan Dewan Penguji:

Tanda Tangan

- | | | | |
|------------------|--------------------------|---|---|
| 1. Penguji Utama | : Suhartono, S.Si, M.Kom | (|) |
| 2. Ketua | : Syahiduz Zaman, M.Kom | (|) |
| 3. Sekretaris | : Fatchurrochman, M.Kom | (|) |
| 4. Anggota | : Ahmad Barizi, M.A | (|) |

**Mengetahui dan Mengesahkan,
Ketua Jurusan Teknik Informatika**

Suhartono, S.Si, M.Kom.
NIP 150 327 241

LEMBAR PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Ibnu Catur Mustofa
NIM : 04550051
Jurusan : Teknik Informatika
Judul Tugas Akhir : **MONITORING GERAKAN PADA RUANGAN
MENGUNAKAN WEBCAM DAN MOTOR
STEPPER**

Dengan ini menyatakan bahwa :

1. Isi dari tugas Akhir yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama termaktub di isi dan tertulis di daftar pustaka dalam Tugas Akhir ini.
2. Apabila dikemudian hari ternyata Tugas Akhir saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 8 Juni 2008

Yang menyatakan,

Ibnu Catur Mustofa

NIM. 04550051

Motto

" لكل يوم زيادة من العلم واسبح في بحور الفوائد "

"Tiap hari bertambah ilmu dan bergelimang dalam lautan berfaedah"

"Hidup adalah proses belajar dan berjuang tanpa batas! Jatuh, berdiri lagi! Kalah, bangun lagi! Gagal, bangkit lagi! Pastikan SUKSES dalam genggam tangan Kita!!!"

*Hidup adalah tanggung jawab,
Tanggung jawab kepada Allah, orang tua, keluarga,
Tanggung jawab terhadap orang lain, negara, diri sendiri dan alam sekitarnya.*

Kalau kita tidak bisa berbakti, mengasahi dan melayani orang tua kita sendiri dengan baik, jangan berharap anak-anak kita kelak mau melakukan hal yang serupa kepada kita.

Sesungguhnya, sejelek apapun rupa maupun sifat orang tua kita, mereka tetap layak dan harus dihormati, jika kita mampu menjankan itu semua niscaya kita tergolong orang-orang yang sukses.

KATA PENGANTAR

Segala puji hanyalah bagi Allah SWT, Tuhan semesta alam atas segala rahmat dan karuniaNya. Shalawat serta salam semoga tetap tercurah bagi Rasulullah Muhammad SAW. Alhamdulillahirobbil'alamin, penulis dapat menyelesaikan Tugas Akhir ini dengan judul : **“Monitoring Gerakan pada Ruang Menggunakan Webcam dan Motor Stepper”**.

Selama mengerjakan penelitian sampai dengan tersusunnya Tugas Akhir ini, banyak bantuan moril maupun materiil yang telah penulis peroleh dari berbagai pihak secara langsung maupun tidak langsung. Untuk itu dengan segala kerendahan hati dan penghargaan setulus – tulusnya penulis menyampaikan ucapan terima kasih kepada:

1. Allah SWT yang telah memberikan Berkah, Rahmad dan Hidayah-Nya hingga terselesaikannya tugas akhir ini.
2. Prof. Dr. H. Imam Suprayogo, selaku Rektor Universitas Islam Negeri (UIN) Malang.
3. Prof. Drs. Sutiman Bambang Sumitro, SU, Dsc. selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Malang.
4. Suhartono, M.Kom Selaku Ketua Jurusan Teknik Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri (UIN) Malang.
5. Fatchurrahman, M.Kom. selaku dosen pembimbing dalam pengerjaan tugas akhir ini, serta kesabaran, ketelatenan dalam memberikan bimbingan dan arahan dalam penulisan tugas akhir ini.

6. Ahmad Barizi, M.A yang meluangkan waktunya untuk memberikan bimbingan dan pengarahan selama penulisan skripsi dibidang integrasi Sains dan Al-Qur'an.
7. Ainul Yaqin, M.Kom selaku dosen konsultasi dalam pengerjaan tugas akhir ini dan segenap staf dan dosen pengajar Teknik Informatika.
8. Bapak Ibuku tercinta yang selalu memberikan do'a, materi dan semangat yang sangat saya butuhkan serta Kakak dan Adikku.
9. M. Arief Budiman, S.Kom, Son Wicaksana, S.Tol, M. Hattan Sururi, S.Boy, dan saudara Rizal S.T yang telah ikut membantu dalam penyelesaian tugas akhir ini.
10. Alumni Kontrakan Winning Eleven with Badhogger Futsal Club thank's for you'r time, bocah-bocah TI terutama angkatan 2004.
11. Dan semua pihak yang tidak dapat saya sebutkan satu-persatu terimakasih banyak.

Dengan bekal dan kemampuan terbatas, penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna, oleh karena itu kritik dan saran yang sifatnya membangun sangat diperlukan untuk memperbaiki mutu penulisan selanjutnya. Akhirnya tiada kata selain harapan semoga tugas akhir ini bermanfaat sesuai dengan maksud dan tujuannya. Amiin Ya Robbal Alamiin.

Malang, 27 Juli 2008

Penulis

DAFTAR ISI

Halaman

HALAMAN JUDUL	i
HALAMAN PENGANTAR	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
LEMBAR PERNYATAAN	v
MOTTO	vi
KATA PENGANTAR.....	vii
DAFTAR ISI	ix
DAFTAR TABEL DAN GAMBAR	xiii
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian	5
1.7 Metode Penelitian	6
1.7 Sistematika Pembahasan.....	7
BAB II TINJAUAN PUSTAKA	9
2.1 Citra (Gambar Diam).....	9
2.1.1 Definisi Citra	9
2.1.2 Komponen Citra Digital.....	11
2.1.3 Representasi Citra Digital	12
2.1.4 Pengolahan Citra.....	12
2.1.5 Operasi Pengolahan Citra.....	14
2.1.5.1 Deteksi Gerakan	15
2.1.6 Bitmap.....	16
2.2 Perangkat Keras (<i>Hardware</i>).....	21
2.2.1 Definisi Perangkat Keras.....	21

2.2.2	Unit Komputer Pribadi (<i>Personal Computer</i>)	21
2.2.3	Unit Webcam	22
2.2.4	Motor Stepper	24
2.3	Perangkat Lunak (Software).....	31
2.3.1	Definisi Perangkat Lunak.....	31
2.3.2	Sistem Operasi (<i>Operating System</i>)	31
2.3.3	Bahasa Pemrograman.....	33
2.4	Komunikasi Parallel.....	34
2.4.1	Antar Muka Port Parallel.....	34

BAB III DESAIN DAN PERANCANGAN SISTEM

3.1	Perancangan Perangkat Keras	36
3.1.1	Pengendali Motor Stepper	36
3.1.2	Kamera	38
3.1.3	Unit Komputer	40
3.2	Perancangan Algoritma Deteksi Gerak.....	41
3.1.1	Algoritma Deteksi Gerak (Secara Umum)	41
3.1.2	Algoritma <i>Capture</i> Citra	45
3.3	Perancangan Sistem Deteksi Gerak	46
3.2.1	Tujuan Sistem Deteksi Gerak	46
3.2.2	Arsitektur Sistem Deteksi Gerak	47
3.4	Perancangan Dan Pembuatan Perangkat Lunak	48
3.4.1	Perancangan Dan Pembuatan GUI.....	48

BAB IV HASIL DAN PEMBAHASAN

4.1	Perangkat Keras	55
4.2	Perangkat Lunak	55
4.3	Penjelasan Program.....	56
4.4	Sistem Kerja Software	56
4.5	Mendeteksi nilai RGB pada titik tertentu.....	61

4.6	Deteksi Gerakan yang disebabkan oleh obyek yang masuk	63
4.7	Pembahasan	68

BAB V KESIMPULAN DAN SARAN

5.1	Kesimpulan	71
5.2	Saran	71

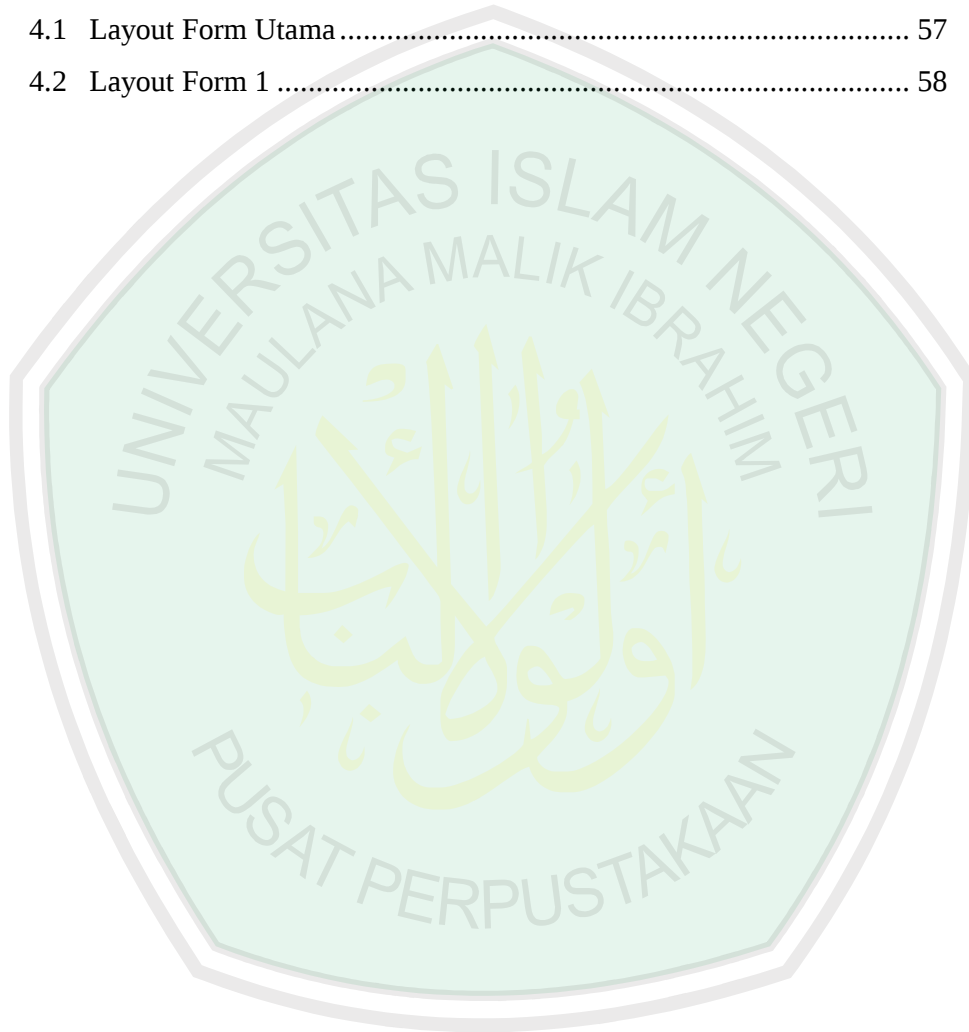
DAFTAR PUSTAKA



DAFTAR TABEL DAN GAMBAR

Tabel	Halaman
2.1 Mode operasi motor langkah 4 fase	21
2.2 Pin-pin konektor port LPT.....	35
3.1 Spesifikasi unit komputer	40
4.1 Koordinat piksel yang digunakan untuk mengidentifikasi letak obyek	63
4.2 Koordinat piksel dengan nilai RGB lebih besar dari 230 pada obyek yang bergerak dari arah kiri.....	64
4.3 Koordinat piksel dengan nilai RGB lebih besar dari 230 pada obyek yang bergerak dari arah kanan.....	66
 Gambar	
2.1 Pengelompokan jenis-jenis citra	9
2.2 Urutan Pengolahan Citra Digital.....	13
2.3 Bitmap dengan nilai matriknya.....	16
2.4 Koordinat RGB	17
2.5 Bitmap hitam putih dan representasi biner.....	19
2.6 Motor langkah jenis VR 4 fase	27
2.7 Dasar-dasar motor langkah PM	30
3.1 Diagram blok pengendali motor stepper	37
3.2 Rangkaian Driver	38
3.3 Logitech QuickCam	39
3.4 <i>Flowchart</i> algoritma deteksi gerak secara umum	42
3.5 <i>Flowchart</i> cara kerja sistem operasi deteksi gerak	44
3.6 Diagram alir proses pengambilan gambar.....	46
3.7 Blok diagram sistem deteksi gerak	47
3.8 Gambar peta form sistem deteksi gerak	49
3.9 Tampilan form utama.....	50
3.10 Tampilan form pilihan kamera	51

3.11 Tampilan form pengaturan kamera.....	52
3.12 Tampilan form pengaturan <i>streaming</i> video	53
3.13 Tampilan form proses deteksi gerakan.....	53
3.14 Tampilan form citra.....	54
4.1 Layout Form Utama	57
4.2 Layout Form 1	58



ABSTRAK

Mustofa, Ibnu Catur. **Monitoring Gerakan pada Ruangan Menggunakan Webcam dan Motor Stepper**. Skripsi. Jurusan Teknik Informatika. Fakultas Sains dan Teknologi. Universitas Islam Negeri (UIN) Malang.
Pembimbing: (1) Fatchurrochman, M.Kom. (2) Ahmad Barizi, M.A.

Kata kunci: *Image processing*, deteksi gerakan, citra, *pixel*, RGB(*red, green, blue*), motor stepper, *port paralel*, PC(*personal computer*)

Sistem pengamanan pada suatu gedung instansi atau perusahaan saat ini banyak yang menggunakan sistem kamera. Dengan sistem ini akan lebih membantu para petugas keamanan, kerja menjadi lebih mudah dan efisien terhadap jarak. Jika pengamanan masih dilakukan secara manual maka petugas akan lebih banyak dan pengawasan menjadi kurang efisien terhadap jarak dan kepekaan bila terjadi suatu kejadian pada salah satu area di instansi atau perusahaan tersebut. Dalam perspektif islam sebagian ulama memperbolehkan penggunaan kamera keamanan selama digunakan pada hal-hal yang positif.

Banyak metode dan teknik yang digunakan dalam sistem pengamanan dengan menggunakan sistem kamera ini, salah satunya adalah dengan menggunakan teknik pengolahan citra atau sering di sebut *image processing*. Metode yang digunakan adalah deteksi gerakan. Metode ini mampu mendeteksi perbedaan nilai RGB (*red, green, blue*) di setiap titik *pixel* pada suatu citra atau gambar digital. Keadaan inilah yang dimanfaatkan untuk di aplikasikan menjadi suatu sistem keamanan. Untuk mendapatkan nilai RGB yang lebih besar maka obyek yang ditangkap oleh kamera harus bercahaya. Dengan nilai RGB yang semakin besar maka pendeteksian gerakan akan lebih mudah. Pendeteksian gerakan dinyatakan aktif jika alarm telah berbunyi.

Bunyi alarm disertai dengan putaran motor stepper yang menggerakkan kamera untuk mengikuti letak obyek yang bergerak. Putaran motor stepper dikontrol oleh PC (*personal computer*) melalui port paralel. Proses pengikutan letak obyek akan berhenti jika obyek telah tidak terdeteksi oleh kamera atau aplikasi dimatikan. Sistem dikatakan stabil apabila masukan acuan dan keluaran yang diinginkan terjadi keseimbangan sebagaimana telah digambarkan dalam Al-Qur'an surat Ar-Ra'du ayat 2 bahwa *field force* (Allah SWT.) yang berlaku sebagai pengontrol bekerja sesuai dengan hukum tertentu untuk menjaga keseimbangan alam dan menahan benda langit pada porosnya serta mencegahnya dari ketergelinciran di angkasa luar atau jatuh menimpa yang lain.

BAB I

PENDAHULUAN

1.1 Latar Belakang

هُوَ اللَّهُ الَّذِي لَا إِلَهَ إِلَّا هُوَ الْمَلِكُ الْقُدُّوسُ السَّلَامُ الْمُؤْمِنُ الْمُهَيْمِنُ الْعَزِيزُ الْجَبَّارُ
الْمُتَكَبِّرُ سُبْحَانَ اللَّهِ عَمَّا يُشْرِكُونَ ﴿٢٣﴾

Dialah Allah yang tiada Tuhan selain Dia, raja, yang Maha suci, yang Maha Sejahtera, yang Mengaruniakan keamanan, yang Maha Memelihara, yang Maha Perkasa, yang Maha Kuasa, yang memiliki segala Keagungan, Maha Suci Allah dari apa yang mereka persekutukan (QS. Al Hasyr : 23).

Dari ayat Al-Qur'an di atas disebutkan bahwa Allah adalah Tuhan yang wajib untuk kita sembah. Allah yang maha memelihara, yang maha perkasa, yang maha kuasa, yang memiliki segala keagungan, juga yang mengaruniakan keamanan bagi manusia.

Keamanan merupakan karunia Allah yang diberikan kepada manusia yang wajib untuk kita syukuri. Allah juga yang telah mengaruniakan manusia akal untuk berpikir. Masalah keamanan merupakan hal terpenting dalam kehidupan di dunia ini. Oleh karena itu, manusia wajib menggunakan akalnya untuk mempelajari dan menciptakan keamanan itu.

Perkembangan teknologi kamera yang semakin pesat sekarang ini ditandai dengan munculnya berbagai macam kamera digital, selain untuk mengambil

gambar diam yang sering disebut dengan foto, juga dapat merekam atau menangkap gambar yang bergerak dalam bentuk video. Webcam adalah salah satu jenis kamera digital yang mampu melakukan hal-hal tersebut. Dari sinilah muncul pemikiran untuk memanfaatkannya sebagai kamera yang berfungsi untuk mengawasi suatu ruangan perkantoran atau pertokoan. Teknologi seperti ini telah banyak terjual di pasaran dengan metode yang berbeda.

Pembuatan sistem keamanan ini dibuat dengan memanfaatkan kamera digital, yakni webcam sebagai pengambil gambar, di mana webcam dapat dengan mudah berkomunikasi data dengan PC (*Personal Computer*) melalui port USB dan mudah dicari di pasaran. Salah satu kelemahan dari webcam adalah kurang mampu mengambil obyek pada ruang gelap yang menyebabkan obyek yang ditangkap menjadi kurang jelas.

Adapun dalam rancang bangun ini juga memanfaatkan metode deteksi gerak pada teknik pengolahan citra digital (*image processing*) diolah dalam software Delphi 5. Metode deteksi gerak adalah proses perbandingan antara gambar (*capture*) pertama dengan kedua. Obyek gambar yang akan diambil harus dalam keadaan diam. Untuk mempermudah proses pengolahan gambarnya maka kapasitas atau ukuran piksel pada citra hanya menggunakan satu jenis ukuran piksel, di samping itu juga memandang kekuatan dari webcam itu sendiri mengenai jarak maksimal pengambilan gambarnya di mana semakin jauh jarak kamera terhadap obyek, obyek akan terlihat semakin kecil dan menjadi kurang jelas.

Delphi 5 dimanfaatkan dalam proses antarmuka (interfacing) untuk menggerakkan motor stepper. Motor stepper berfungsi untuk menggerakkan webcam berputar ke arah kanan dan kiri untuk mengikuti gerakan obyek, begitu pula pada proses menyalakan alarm yang memanfaatkan soundcard pada PC (*Personal Computer*). Alarm ini bertujuan untuk memberikan indikasi bahwa ada suatu perubahan pada gambar yang artinya ada suatu gerakan.

Untuk menghubungkan motor stepper dengan PC agar dapat diperintah bergerak melalui PC dibutuhkan suatu driver sebagai penghubungnya. Driver ini berupa rangkaian elektronik yang dapat dibuat sendiri, dan mendapat masukan dari port parallel PC yang kemudian keluarannya digunakan untuk mengontrol gerakan motor stepper sesuai dengan perintah dari softwarena.

Hal ini juga terjadi pada benda-benda angkasa yaitu misalnya gerak matahari juga mempunyai pengatur atau pengendali agar tetap bergerak atau beredar pada tempat peredarannya dan berputar dengan kecepatan stabil dalam mengitari pusat galaksinya. Yang dimaksud pengendali disini adalah Allah SWT Yang Maha Kuasa atas segala. Fakta ilmiah ini telah dinyatakan oleh Al-Qur'an sebagai berikut:

وَالشَّمْسُ تَجْرِي لِمُسْتَقَرٍّ لَّهَا ۚ ذَٰلِكَ تَقْدِيرُ الْعَزِيزِ الْعَلِيمِ ﴿٣٨﴾

Dan matahari berjalan di tempat peredarannya. Demikianlah ketetapan yang Maha Perkasa lagi Maha Mengetahui. (QS Yâsin/36:38).

Dari ayat diatas dijelaskan bahwa matahari beredar pada poros tertentu (لمستقر لها) yang telah diatur dan ditetapkan oleh Allah SWT. Apabila gerak matahari itu tidak diatur atau dikendalikan oleh Allah maka akan terjadi ketidakteraturan di angkasa ini yang tentunya akan merusak kehidupan, seperti halnya putaran kecepatan motor stepper jika tidak diatur atau dikontrol dengan sebuah pengontrol maka akan terjadi ketidakstabilan sistem yang menyebabkan kerusakan pada hasil *capture* dan pada alat tersebut. Ada dua hal yang perlu digaris bawahi dari inti ayat diatas adalah tentang pemantauan gerakan sesungguhnya Allah mengetahui apa yang selalu kita kerjakan dan motor stepper untuk pengatur gerakan stepper tersebut.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah tersebut diatas, dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana memanfaatkan Webcam sebagai sistem keamanan?
2. Bagaimana membuat software menggunakan teknik pengolahan citra digital untuk sistem keamanan?
3. Bagaimana menghubungkan motor langkah dengan PC, agar motor langkah dapat diperintah melalui PC?

1.3 Batasan Masalah

Beberapa batasan yang digunakan sebagai acuan dalam perancangan sistem keamanan ini adalah:

1. Sistem keamanan yang dirancang dipergunakan untuk ruangan tertutup dengan daerah yang tidak terlalu luas dan mempunyai penerangan yang tetap.
2. *Capture device* yang digunakan adalah webcam produksi Logitech QC Pro dengan system operasi windows.
3. Jarak obyek terhadap kamera diperkirakan ± 5 meter.
4. Motor stepper hanya dapat melakukan gerak 1 dimensi, yaitu ke kiri dan ke kanan.

1.4 Tujuan Penelitian

Tujuan pembuatan alat sistem keamanan ini adalah:

1. Menambah nilai guna dari webcam sebagai bagian dari sistem keamanan dengan menggunakan teknik pengolahan citra (*image processing*), di mana webcam berfungsi sebagai pengambil gambar.
2. Menerapkan proses pengolahan citra digital (*image processing*) yang diolah dalam software Delphi 5 untuk sistem pengaman.
3. Membuat rangkaian driver yang digunakan sebagai antarmuka (interface) motor langkah terhadap PC.

1.5 Manfaat Penelitian

Dalam pembuatan alat ini diharapkan dapat memberi banyak manfaat, di antaranya:

1. Dapat membantu security dalam mengawasi sistem keamanan dalam suatu ruangan.
2. Menerapkan metode pengolahan citra digital yakni metode penyamplingan gambar yaitu membedakan gambar 1 dengan gambar 2 dan seterusnya.

1.6 Metode Penelitian

Metode yang digunakan pada penelitian ini adalah :

- a. Studi pustaka

Mempelajari teori-teori yang menunjang dan mencari informasi yang berhubungan dengan penelitian ini.

- b. Perancangan sistem

Dalam sistem ini dibuat perancangan sistem monitoring gerakan menggunakan webcam dan motor stepper. Mulai perancangan perangkat lunak yang meliputi software dengan menggunakan bahasa pemrograman delphi dan juga perangkat keras yang meliputi hardware dalam pembuatan driver motor stepper.

c. Pengujian dan analisis sistem

Menguji sistem/perangkat lunak yang telah dibuat apakah sesuai dengan yang diharapkan.

d. Penyusunan laporan

Ini adalah langkah terakhir dalam penelitian ini.

1.7 Sistematika Penulisan

BAB I Pendahuluan

Bab ini berisi latar belakang, perumusan masalah, tujuan, batasan masalah dan metodologi penelitian tugas akhir ini.

BAB II Tinjauan Pustaka

Bab ini menjelaskan konsep dan teori dasar yang mendukung penulisan tugas akhir ini seperti pengertian citra, perangkat keras, perangkat lunak, dan komunikasi paralel.

BAB III Analisis dan Perancangan Sistem Monitoring Gerakan pada Ruangan Menggunakan Webcam dan Motor Stepper

Bab ini menjelaskan mengenai analisis dan perancangan Perancangan Sistem Monitoring Gerakan pada Ruangan Menggunakan Webcam dan Motor Stepper dan sistem secara keseluruhan.

BAB IV Hasil dan Pembahasan

Bab ini berisi pengujian dan analisis terhadap hasil pengujian dari aplikasi yang telah dibangun.

BAB V Kesimpulan dan Saran

Bab ini berisi kesimpulan dan saran terhadap seluruh kegiatan tugas akhir yang telah dilakukan.



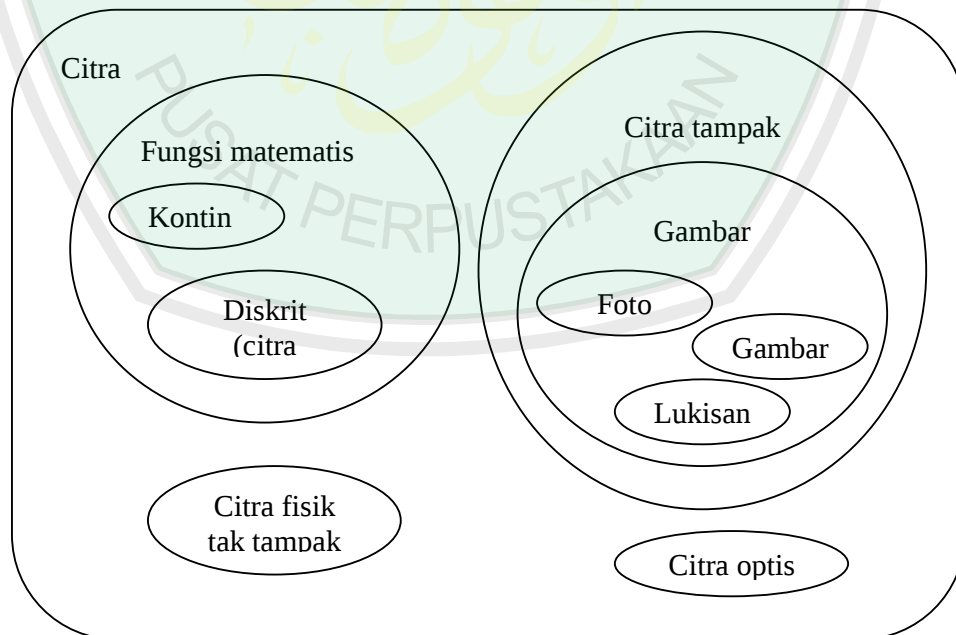
BAB II

TINJAUAN PUSTAKA

2.1 Citra (Gambar Diam)

2.1.1 Definisi Citra

Definisi Citra menurut Webster adalah suatu representasi, kemiripan, atau imitasi dari suatu obyek atau benda. Selain itu juga didalam sebuah citra juga terdapat Kompresi Citra adalah aplikasi kompresi data yang dilakukan terhadap citra digital dengan tujuan untuk mengurangi redundansi dari data-data yang terdapat dalam citra sehingga dapat disimpan atau ditransmisikan secara efisien. Citra dapat dikelompokkan menjadi citra tampak dan citra tak tampak, sebagaimana disajikan pada gambar 1 di bawah ini :



Gambar 2.1 Pengelompokan jenis-jenis citra (Castleman, 1996)

Contoh citra tampak dalam kehidupan sehari-hari adalah foto keluarga, gambar yang nampak pada layar monitor dan televisi, serta hologram (citra optis). Sedangkan contoh citra tak tampak adalah data gambar dalam file (citra digital) dan citra yang merepresentasikan menjadi fungsi matematis. Di samping itu ada juga citra fisik tak tampak, misalnya citra distribusi panas di kulit manusia serta peta densitas dalam suatu material. Untuk dapat dilihat mata manusia, citra tak tampak ini harus diubah menjadi citra tampak, misalnya dengan menampilkannya di monitor, dicetak di atas kertas, dan sebagainya.

Pengertian citra dapat diimplementasikan dengan ayat al-qur'an (QS. Al-Tiin ayat 4), penjelasan yang lebih dalam sebagai justifikasi dan afirmasi terhadap penggunaan dalil tersebut, khususnya kata *khalaqna* dan *taqwim*. Allah berfirman dalam ayat tersebut sebagaimana ayat berikut:

لَقَدْ خَلَقْنَا الْإِنْسَانَ فِي أَحْسَنِ تَقْوِيمٍ

Artinya: *Sesungguhnya Kami telah menciptakan manusia dalam bentuk yang sebaik-baiknya.* (QS. At-Tin: 4).

Imam Muslim meriwayatkan hadits yang artinya “*Sesungguhnya Allah Maha Indah dan mencintai keindahan*”. Disebutkan dalam ayat al-Qur'an bahwa manusia diciptakan dari citra Yang Maha Mencipta, yaitu Allah swt. Jika dirujuk lebih dalam pada hadits seperti disebutkan di atas, Allah Maha indah, tentunya ciptaan-Nya pun adalah kreasi yang sangat indah. Atas keindahan-Nya itulah, Allah juga menyukai keindahan,

dengan pengertian lain adalah keindahan di atas bentuk yang sebaik-baiknya.

2.1.2 Komponen Citra Digital

Setiap citra *digital* memiliki beberapa karakteristik, antara lain ukuran citra, resolusi, dan format nilainya. Karakteristik citra digital dalam al-Qur'an disebutkan dengan taqwim (تَقْوِيم) yang mempunyai makna bentuk yakni bentuk dari sebuah citra. Umumnya citra *digital* berbentuk persegi panjang yang memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik atau pixel, sehingga ukuran citra selalu bernilai bulat.

Format citra digital ada bermacam-macam. Karena sebenarnya citra mempresentasikan informasi tertentu, sedangkan informasi tersebut dapat dinyatakan secara bervariasi, maka citra yang mewakilinya dapat muncul dalam berbagai format. Citra yang mempresentasikan informasi yang hanya bersifat biner untuk membedakan 2 keadaan tentu tidak sama citra dengan informasi yang lebih kompleks sehingga memerlukan lebih banyak keadaan yang diwakilinya. Pada citra digital semua informasi disimpan dalam angka, sedangkan penampilan angka tersebut biasanya dikaitkan dengan warna.

Citra digital tersusun atas titik-titik yang biasanya berbentuk persegi panjang atau bujur sangkar (pada beberapa sistem pencitraan, piksel-piksel penyusun citra ada pula yang berbentuk segi enam) yang

secara beraturan membentuk barisan-barisan dan kolom-kolom. Setiap titik memiliki koordinat sesuai dengan posisinya dalam citra. Koordinat ini biasanya dinyatakan dalam bilangan bulat positif, yang dapat dimulai dari 0 atau 1 bergantung pada system yang digunakan.

Setiap titik juga memiliki nilai berupa angka digital yang mempresentasikan informasi yang diwakili titik tersebut. Format nilai piksel sama dengan format citra keseluruhan. Pada kebanyakan system pencitraan, nilai ini biasanya berupa bilangan bulat positif juga.

2.1.3 Representasi Citra

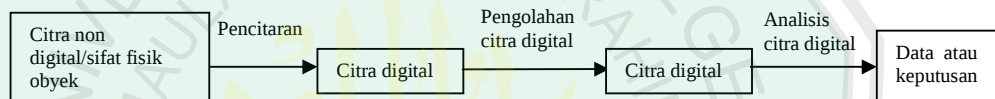
Komputer dapat mengolah isyarat-isyarat elektronik digital yang merupakan kumpulan sinyal biner (bernilai dua: 0 dan 1). Untuk itu, citra digital harus mempunyai format tertentu yang sesuai sehingga dapat mempresentasikan obyek pencitraan dalam bentuk kombinasi data biner.

Pada kebanyakan kasus, terutama untuk keperluan penampilan secara visual, nilai data digital tersebut mempresentasikan warna dari citra yang diolah, dengan demikian format data citra digital berhubungan erat dengan warna. Format citra digital yang banyak dipakai adalah citra biner, skala keabuan, warna, dan warna berindeks.

2.1.4 Pengolahan Citra

Kegiatan untuk mengubah informasi citra fisik non digital menjadi digital disebut sebagai pencitraan (*imaging*). Citra digital dapat diolah

dengan komputer karena berbentuk data numeris. Suatu citra digital melalui pengolahan citra digital (*digital image processing*) menghasilkan citra digital yang baru; termasuk di dalamnya adalah perbaikan citra (*image restoration*) dan peningkatan kualitas citra (*image enhancement*). Sedangkan analisis citra digital (*digital image analysis*) menghasilkan suatu keputusan atau suatu data; termasuk di dalamnya adalah pengenalan pola (*pattern recognition*), sebagaimana disajikan pada Gambar 2.2



Gambar 2.2 Urutan Pengolahan Citra Digital

Setiap tampilan citra digital memiliki beberapa karakteristik, antara lain ukuran citra, resolusi, dan format nilainya. Umumnya citra digital berbentuk persegi panjang yang memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik atau piksel (berasal dari kata *picture element*) yang memiliki koordinat sesuai dengan posisinya dalam citra. Koordinat ini biasanya dinyatakan dalam bilangan bulat positif, yang dapat dimulai dari 0 atau bergantung pada sistem yang digunakan. Setiap titik juga memiliki nilai berupa angka digital yang merepresentasikan informasi yang diwakili titik tersebut. Format nilai piksel sama dengan format citra keseluruhan. Pada citra digital semua informasi disimpan dalam bentuk angka, sedangkan penampilan angka tersebut biasanya dikaitkan dengan warna.

2.1.5 Operasi Pengolahan Citra

Pengolahan citra pada dasarnya dilakukan dengan cara memodifikasi setiap titik dalam citra tersebut sesuai dengan keperluan.

Secara garis besar, modifikasi tersebut dikelompokkan menjadi :

1. Operasi titik, dimana setiap titik diolah secara tidak gayut terhadap titik-titik yang lain.
2. Operasi global, dimana karakteristik global (biasanya berupa sifat statistik) dari citra digunakan untuk memodifikasi nilai setiap titik.
3. Operasi temporal/bebasis bingkai, dimana sebuah citra diolah dengan cara dikombinasikan dengan citra lain.
4. Operasi geometri, dimana bentuk, ukuran, atau orientasi citra dimodifikasi secara geometris.
5. Operasi banyak titik bertetangga , dimana data dari titik yang bersebelahan (bertetangga) dengan titik yang ditinjau ikut berperan dalam mengubah nilai.
6. Operasi morfologi, yaitu operasi yang berdasarkan segmen atau bagian dalam citra yang menjadi perhatian.

Batasan yang dipakai dalam pengolahan citra yang difokuskan pada format citra skala keabuan 8 bit dengan warna hitam pekat untuk nilai minimum (0) dan warna putih cemerlang untuk nilai maksimal (255), serta citra warna *true color*.

2.1.5.1 Deteksi gerakan

Deteksi gerakan secara sederhana dapat dilakukan dengan mencari beda antara 2 buah citra yang berurutan pada hasil pencitraan menggunakan kamera video digital. Untuk mengetahui beda antara 2 buah citra tersebut maka operasi yang digunakan adalah pengurangan. Operasi pengurangan pada bagian yang tidak bergerak dalam citra akan menghasilkan nilai RGB (red,green,blue) per piksel sama dengan nol, sedangkan bagian yang bergerak dalam citra memberikan nilai RGB (red,green,blue) per piksel tidak sama dengan nol. Dengan mengevaluasi selisih nilai RGB per piksel, dapat diketahui apakah pada citra terdapat obyek yang bergerak.

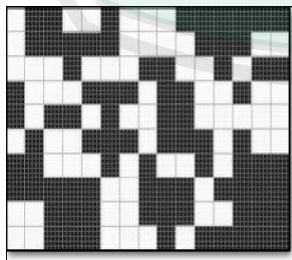
Nilai RGB per piksel pada koordinat x,y citra pertama dikurangkan dengan nilai piksel koordinat x,y citra kedua. Seperti dijelaskan di atas jika hasil pengurangan bernilai nol maka dinyatakan tidak ada gerakan, begitu pula sebaliknya. Untuk mengantisipasi nilai kurang dari nol maka hasil dari pengurangan ditambahkan suatu konstanta.

Obyek yang ditangkap dibuat bercahaya untuk mempermudah pendeteksian gerakan. Dengan obyek yang bercahaya perbedaan citra dari sisi nilai RGB per piksel akan lebih jelas dan pendeteksian gerakan jadi lebih mudah.

2.1.6 Bitmap

Citra *bitmap* adalah susunan bit-bit warna untuk tiap *pixel* yang membentuk pola tertentu. Pola-pola warna ini menyajikan informasi yang dapat dipahami sesuai dengan persepsi indera penglihatan manusia. Format file ini merupakan format grafis yang fleksibel untuk *platform Windows* sehingga dapat dibaca oleh program grafis manapun. Format ini mampu menyimpan informasi dengan kualitas tingkat 1 bit samapi 24 bit. (<http://slametriyanto.web.id>). Sedangkan pengertian Bitmap yang lain adalah sekumpulan titik-titik yang disebut *pixel*. *Pixel* menyimpan informasi tentang warna. Titik - titik ini saling berderet berhimpit membentuk baris dan kolom sehingga membentuk gambar yang utuh. Sebuah bitmap dapat mempunyai resolusi dan kedalaman bit yang berbeda. (Cybulski,2004).

Citra bitmap didefinisikan sebagai fungsi $f(x,y)$ dengan x dan y adalah koordinat bidang. Besaran f untuk tiap koordinat (x,y) disebut intensitas atau derajat keabuan citra pada titik tersebut.



(a) *bitmap* 15×10 *pixel*

1	0	0	1	1	0	1	1	1	0	0	0	0	0	0
1	0	0	0	0	0	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	0	0	1	1	0	1	0	0
0	1	1	1	0	0	0	1	0	0	1	1	0	1	1
0	1	0	0	1	0	1	1	0	0	1	1	1	1	1
1	0	1	1	0	0	0	1	0	0	0	1	0	1	1
0	0	1	1	1	0	1	0	1	1	0	1	0	0	0
0	0	0	0	0	1	1	0	0	0	1	0	0	0	0
1	1	0	0	0	1	1	0	0	0	1	1	0	0	0
1	1	0	0	0	1	1	1	0	1	0	0	0	0	0

(b) Matriks *bitmap*

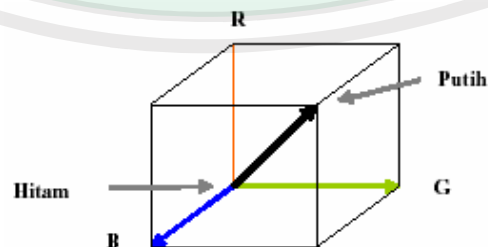
Gambar 2.2. *Bitmap* dengan nilai matriksnya

Dari definisi di atas yang diperjelas oleh gambar 1.1, *bitmap* dimodelkan dalam bentuk matriks. Nilai *pixel* atau entri-entri dari matriks ini mewakili warna yang ditampilkan dimana ordo matriks merupakan dimensi panjang dan lebar dari *bitmap*.

Nilai-nilai warna ditentukan berdasarkan intensitas cahaya yang masuk. Dalam komputer, derajat intensitas cahaya diwakili oleh bilangan cacah. Nilai 0 menerangkan tidak adanya cahaya sedangkan nilai yang lain menerangkan adanya cahaya dengan intensitas tertentu. Nilai-nilai ini bisa didapatkan melalui fungsi-fungsi yang disediakan oleh bahasa pemrograman berdasarkan input berupa lokasi entri-entri matriks yang hendak dicari.

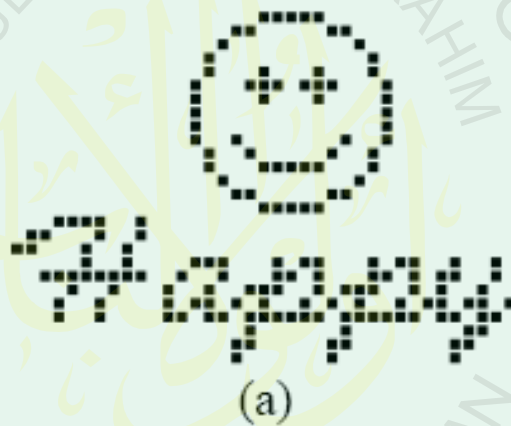
2.1.6.1 Pixel

Pixel adalah bagian terkecil dari sebuah gambar. *Pixel* kependekan dari *Picture Element*. *Pixel* menggambarkan posisi koordinat dan mempunyai intensitas yang dapat dinyatakan dengan bilangan. Intensitas ini menunjukkan warna gambar, dalam bentuk RGB (*Red, Green, Blue*).



Gambar 2.3 Koordinat RGB

Dalam gambar hitam putih, setiap pixel dapat direpresentasikan dalam satu bit, berisi 1 bila warnanya hitam dan berisi 0 bila warnanya putih. Ketika komputer akan menggambar mulailah komputer mencari angka-angka yang mengartikan informasi dari gambar. Setiap komputer menemukan angka 0 komputer akan menggambar warna putih, ketika menemukan angka 1 komputer akan menggambar sebuah *pixel* dengan warna hitam.

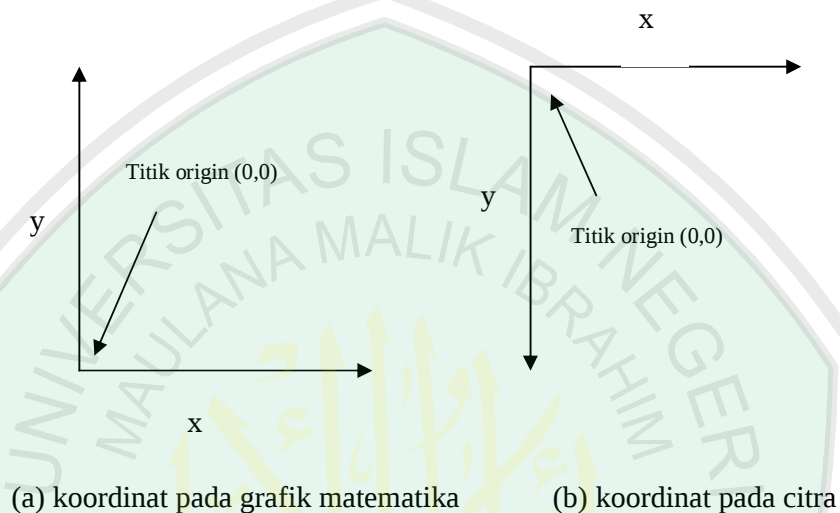


(b)

(a) Bitmap hitam putih
(b) Representasi biner

Menurut Usman Ahmad (2005:14) sebuah *pixel* adalah sampel dari pemandangan yang mengandung intensitas citra yang dinyatakan dalam bilangan bulat. Sebuah citra adalah kumpulan *pixel-pixel* yang disusun dalam larik dua dimensi. Indeks baris dan kolom (x,y) dari sebuah *pixel* dinyatakan dalam bilangan bulat. *Pixel* (0,0) terletak pada sudut kiri atas pada citra, indeks x bergerak ke kanan dan indeks y bergerak ke bawah. Konvensi ini dipakai merujuk pada cara penulisan larik yang digunakan dalam pemrograman komputer. Letak titik origin pada koordinat grafik citra dan koordinat pada grafik matematika terdapat perbedaan. Hal yang berlawanan untuk arah vertikal berlaku pada kenyataan dan juga pada

sistem grafik dalam matematika yang sudah lebih dulu dikenal. Gambar berikut memperlihatkan perbedaan kedua sistem ini.



Gambar 2.4 Perbedaan letak titik origin pada koordinat grafik dan pada citra

2.1.6.2 Kedalaman Bit Dalam Gambar

Kedalaman bit dalam gambar adalah jumlah bit yang digunakan untuk menyimpan informasi tentang sebuah *pixel* dalam gambar. Semakin tinggi kedalamannya, semakin banyak warna yang dapat dihasilkan dalam sebuah gambar. Contohnya pada kedalaman bit terendah ,1 bit sebuah gambar hanya dapat menghasilkan 2 warna, hitam dan putih. Ini karena hanya ada dua kombinasi angka dalam satu buah bit, yaitu 0 dan 1. Empat bit warna dapat menghasilkan 16 Warna karena mereka ada 16 kombinasi yang berbeda dalam empat bit, yaitu : 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111. Pada warna

dengan kedalaman 8 bit, kita dapatkan 256 warna yang dapat dihasilkan.

Pada kedalaman 24 bit, tiap *pixel* dapat dinyatakan dengan :

- bit 0 sampai dengan 7 untuk warna merah.
- bit 7 sampai dengan 15 untuk warna hijau
- bit 16 sampai dengan 24 untuk warna biru

Kemungkinan kombinasi warna yang ada adalah $= 256^3 + 256^2 + 256^1 = 16.843.008$, dimana nilai 0 menyatakan warna hitam sedangkan nilai 16 843 008 menyatakan warna putih.

2.2 Perangkat Keras (Hardware)

2.2.1 Definisi Perangkat Keras

Pengertian dari perangkat keras adalah peralatan yang secara fisik terlihat dan bisa dijamah, perangkat keras terdiri dari seperangkat computer pribadi, I/O *device*, dan memori.

2.2.2 Unit Komputer Pribadi (Personal Computer)

Komputer jenis ini (*Personal Computer*) adalah komputer yang ukuran relatif kecil dan dapat diletakkan di meja. Komputer ini ditujukan buat kenyamanan dan lebih responsif bagi pengguna komputer. Berbagai sistem operasi dapat berjalan dalam komputer jenis ini. Menurut Gramacomp Team, sering terjadi kesalahan dalam membagi jenis-jenis komputer bahkan di kalangan para pakar sekalipun. Sering terjadi kelas-

kelas komputer tidak dibagi menurut dasar pembagiannya, hingga artinya menjadi campur aduk. Misalnya, masyarakat sering sekali menyebut *PC* (*Personal Computer*) sama dengan *desktop*. Padahal, hal ini adalah dua hal yang berbeda acuannya. Sebuah *desktop* hampir pasti *PC*, akan tetapi *PC* sangat mungkin bukan *desktop*, tapi bisa saja *notebook*. Kami membagi jenis komputer berdasarkan tiga dasar, yaitu ukuran, karakteristik, dan jenis data. Pada kenyataannya dapat saja terjadi sebuah komputer dimasukkan dalam dua atau lebih kelas yang ada di dasar klasifikasi yang sama. Hal ini dimungkinkan oleh perkembangan jaman. Misalkan, dahulu semua *desktop* disebut *microcomputer*, tetapi karena perkembangan jaman, istilah *microcomputer* menjadi kurang spesifik (karena munculnya *notebook*, *handheld PC*, *desknote*).

2.2.3 Unit Webcam

Kamera Video Konferensi yang juga dikenal dengan Kamera Web (Webcam) adalah kamera yang dirancang penggunaannya pada World Wide Web. Kamera Web sekarang harganya turun dengan cepat yang menjadikannya sebagai suatu produk pasar massa. Pada kamera web Xcam2 yang bentuknya kecil, tetapi mempunyai kemampuan kamera video yang dapat diletakkan di mana saja dan dapat mentransmisikan video berwarna ke TV, VCR atau pada PC (Personal Computer).

Webcam (*Web Camera*) secara sederhana terdiri atas digital kamera yang tersambung dengan komputer. Webcam terintegrasi dengan

komputer dengan melalui port USB (sebelumnya kamera terhubung ke computer melalui port parallel). Cara kerja webcam tidak jauh berbeda dengan cara kerja kamera tradisional yang berbasis film, yaitu memilih obyek yang akan direkam dengan menggunakan jendela pengintai. Hal ini bisa kita aplikasikan dengan surat al-Baqarah ayat 110 dibawah ini:

وَأَقِيمُوا الصَّلَاةَ وَآتُوا الزَّكَاةَ وَمَا تُقَدِّمُوا لِأَنفُسِكُمْ مِنْ خَيْرٍ يَجِدُوهُ عِنْدَ اللَّهِ إِنَّ اللَّهَ بِمَا تَعْمَلُونَ بَصِيرٌ ﴿١١٠﴾

Dan Dirikanlah shalat dan tunaikanlah zakat. dan kebaikan apa saja yang kamu usahakan bagi dirimu, tentu kamu akan mendapat pahala nya pada sisi Allah. Sesungguhnya Allah Maha melihat apa-apa yang kamu kerjakan (QS. Al Baqarah/2:110).

Sesuai dengan potongan ayat diatas bahwasannya Allah melihat apa yang kita kerjakan walaupun kita sembunyi atau tanpa sepengetahuan orang lain. Hal ini yang membuat inspirasi untuk membuat sistem keamanan dengan menggunakan webcam. Sistem keamanan pasti ada kelebihan dan juga kekurangannya, walaupun ada kelebihan dalam sistem keamanan tersebut pastilah tidak dapat menandingi kekuasaan Allah. Apapun yang manusia kerjakan dan kapanpun pasti kelak akan di mintai pertanggung jawaban. Kamera keamanan juga bisa diimplementasikan dengan surat an-nissa ayat 15, bahwasanya hasil dari kamera keamanan juga bisa menjadi saksi, sebagaimana penjelasan saksi pada pada surat an-nissa ayat 15 tersebut.

Kemudian bayangan objek tersebut difokuskan oleh lensa ke sebuah peralatan peka cahaya yang dapat berupa sensor CCD (*Charge Coupled Device*) atau CMOS (*Complementary Metal-Oxide Semiconductor*). Setiap elemen sensor mengkonversi cahaya ke tegangan listrik yang sesuai dengan kecerahan (*brightness*) yang diteruskan ke sebuah ADC (*Analog to Digital Converter*) yang menerjemahkan fluktuasi tegangan dari CCD ke dalam kode biner. Keluaran dari ADC dikirimkan ke sebuah DSP (*Digital Signal Processor*) yang menyesuaikan kontras dan detail dari gambar, serta mengkompres gambar sebelum dikirim ke media penyimpanan. Semakin terang cahaya yang dihasilkan sensor CCD, semakin tinggi tegangan dan semakin besar resolusi piksel yang dihasilkan komputer menyebabkan semakin besar detail yang ditangkap oleh kamera.

2.2.4 Motor Stepper

Motor stepper atau biasa disebut motor langkah merupakan motor dengan rotor yang dapat berputar perlangkah, berputar penuh, berputar searah, ataupun berputar berlawanan dengan arah perputaran jarum jam. Motor langkah adalah salah satu komponen elektromekanik yang banyak digunakan, terutama yang berkaitan dengan peralatan *peripheral* komputer digital. Beberapa contoh penggunaan motor ini antara lain adalah untuk menggerakkan *head* dan memutar *floppy disk* pada pengendali disk (*disk drive*). Pada pencetak (*printer*) motor ini digunakan untuk menggerakkan *print head*, pada setiap pergantian karakter, dan menggerakkan kertas pada setiap pergantian baris. Selain untuk keperluan *peripheral* seperti tersebut

atas, motor langkah juga banyak digunakan untuk keperluan-keperluan dalam industri, seperti penggerak robot dan otomatisasi peralatan yang memerlukan gerakan-gerakan akurat.

Karakteristik yang membedakan motor langkah dengan motor biasa adalah :

- Motor langkah mempunyai momen gaya yang tinggi pada kecepatan putar yang rendah.
- Dalam keadaan diam, pada kumparan tetap ada arus, sehingga rotor dalam keadaan mencengkeram.
- Motor langkah dapat dioperasikan dengan kalang terbuka (*open loop*) dengan ketepatan posisi yang tinggi untuk kecepatan yang rendah, sehingga tidak ada langkah yang hilang selama perputaran.
- Pengoperasian motor langkah bersesuaian dengan teknik kontrol digital, sehingga dapat dihubungkan dengan komputer mikro.

Motor langkah dapat mempunyai langkah yang kecil, sehingga dapat dicapai resolusi yang besar. Ukuran sudut langkah tergantung pada spesifikasi masing-masing motor langkah, yaitu dari $0,9^\circ$ sampai 30° .

Hubungan antara sudut langkah θ dengan jumlah langkah, diberikan oleh persamaan :

$$S = 360/\theta$$

sedang kecepatan putar motor langkah diberikan dalam kaitan dengan jumlah langkah perdetik dan laju langkah, yaitu:

$$n = 60 \text{ f/s}$$

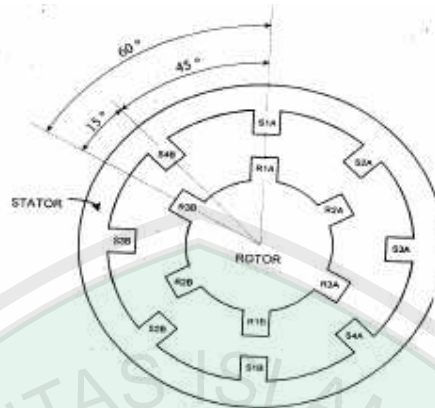
di mana :

n = jumlah putaran (rpm)

f = laju langkah (Hz)

S = jumlah langkah

Jenis motor langkah, berdasarkan sifat kemagnetan stator maupun rotornya, dibedakan menjadi dua yaitu jenis magnet tetap (PM, permanent magnet) dan jenis reluktansi berubah (VR, variable reluctance). Pada jenis PM, rotor terbuat dari magnet permanen. Gigi magnet kutub utara dan kutub selatan dibuat berselingan. Operasi motor langkah jenis PM ini dilakukan dengan pensaklaran secara bergantian. Karena langkah pada motor langkah jenis PM menggunakan magnet permanen, maka keadaan mencengkeram pada jenis ini lebih kuat daripada jenis VR. Sedangkan motor langkah jenis VR mempunyai gigi-gigi yang berpasangan dan saling berhadapan. Jumlah gigi pada stator dan rotor tidak sama, sehingga susunan magnet selalu berusaha meminimalkan jarak celah dalam bidang magnetik antara gigi rotor dan stator, sehingga motor langkah ini dapat dioperasikan. Kemagnetan pada motor langkah jenis VR ditimbulkan dengan memberi arus pada kumparan-kumparan secara berurutan.



Gambar 2.5 Motor langkah jenis VR 4 fase

Pada Gambar 2.5 di atas, stator mempunyai 8 gigi yang terdiri 4 pasang yakni gigi S1A berpasangan dengan S1B, gigi S2A berpasangan dengan S2B, dan seterusnya. Sedangkan rotor hanya mempunyai 6 gigi. Perputaran satu langkah terjadi bila kumparan S1A diberi arus, lalu gigi R1A pada rotor akan meluruskan diri dengan gigi S1A pada stator. Ini berarti bahwa pemberian arus pada kumparan S1A dan kemudian pada kumparan S2A mengakibatkan motor langkah bergerak satu langkah. Jika perpindahan arus dilanjutkan ke kumparan berikutnya secara berurutan, maka rotor akan berputar sesuai dengan jumlah, kecepatan, dan arah urutan tersebut.

Untuk memutar motor langkah terdapat 3 macam cara yaitu:

1. Dengan memberikan arus pada satu pasangan kumparan yang mempunyai lilitan yang berlawanan, yang akan menghasilkan langkah penuh (*Full Step*).

2. Dengan memberikan arus pada dua pasang kumparan yang berdekatan (S1A-S1B dan S2A-S2B), maka akan diperoleh kutub magnet yang sama. Cara ini juga akan memberikan langkah penuh.
3. Gabungan dari kedua cara di atas akan memberikan langkah setengah (*Half Step*).

Cara eksitasi motor langkah, baik yang *full step* maupun yang *half step*, diberikan pada tabel berikut ini.

Tabel 2.1 Mode operasi motor langkah 4 fase :

Langkah Penuh:

Langkah	Kumparan				Desimal
	L4	L3	L2	L1	
1	0	0	0	1	1
2	0	0	1	0	2
3	0	1	0	0	4
4	1	0	0	0	8
5	0	0	0	1	1
1	0	0	1	1	3
2	0	1	1	0	6
3	1	1	0	0	12
4	1	0	0	1	9
5	0	0	1	1	3

Langkah Setengah:

Langkah	Kumparan				Desimal
	L4	L3	L2	L1	
1	0	0	0	1	1
2	0	0	1	1	3
3	0	0	1	0	2
4	0	1	1	0	6
5	0	1	0	0	4
6	1	1	0	0	12
7	1	0	0	0	8
8	1	0	0	1	9
9	0	0	0	1	1

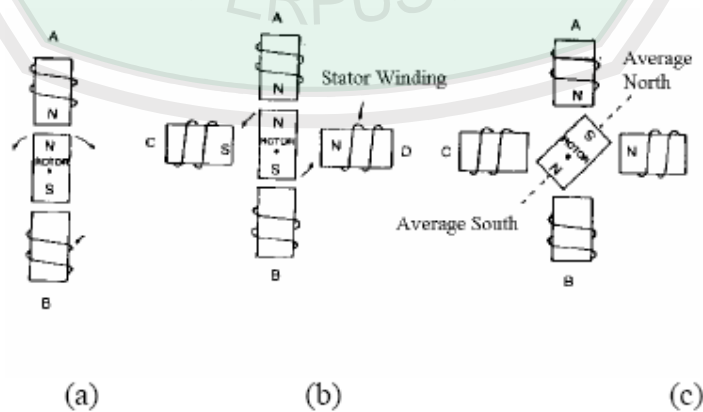
Arah perputaran motor langkah ditentukan sesuai dengan arah pergantian pasangan kumparan yang diberi arus (dieksitasi). Bila urutan pergantian awal searah dengan arah perputaran jarum jam (*clock wise, CW*), maka urutan sebaliknya akan menghasilkan putaran yang berlawanan dengan putaran jarum jam (*counter clock wise, CCW*) (Santoso,1999).

Motor langkah dengan jenis PM (*permanent magnet*) dapat dijelaskan sebagai berikut ini. Bentuk dasar dari sebuah motor langkah yang paling sederhana terdiri atas sebuah rotor, yang merupakan magnet permanen, dan sebuah stator yang dililiti kumparan, sehingga dapat membentuk magnet listrik. Jika stator diberi arus listrik, maka sisi-sisi rotor akan membentuk kutub-kutub magnet. Jika kutub magnet stator dan rotor sama (Gambar 2.2.a), kedua magnet akan saling tolak-menolak,

sehingga mengakibatkan rotor berputar. Perputaran ini dapat memiliki dua macam arah, tergantung pada faktor mekanik motor langkah itu sendiri. Besarnya perputaran adalah 180° .

Gambar 2.2.b memperlihatkan motor langkah yang terdiri atas dua buah stator dengan sebuah rotor. Prinsip kerja motor langkah ini sama dengan prinsip kerja motor langkah yang terdiri atas sebuah rotor dan stator. Jika arah arus listrik dan arah stator sedemikian rupa sehingga membentuk konfigurasi listrik magnet seperti pada Gambar 2.2.b, maka rotor akan berputar berlawanan arah dengan arah putaran jarum jam (*ccw*), sebesar 90° . Perputaran ini disebut perputaran langkah penuh (*full step*).

Jika magnet permanen dan magnet listrik membentuk konfigurasi seperti Gambar 2.2.c, motor akan berputar 45° cw. Perputaran ini disebut perputaran setengah langkah (*half step*). Jika motor langkah terdiri atas empat pasang stator, besar langkah penuhnya adalah 45° dan setengah langkahnya adalah $22,5^\circ$.



Gambar 2.6 Dasar-dasar motor langkah PM

2.3 Perangkat Lunak (Software)

2.3.1 Definisi Perangkat Lunak

Pengembangan perangkat lunak diarahkan pada realisasi sistem aplikasi yang mampu menunjang proses transaksi ekonomi yang cepat dan aman, serta pengambilan keputusan yang benar dan cepat. Harga yang terjangkau dan daya saing pada tingkat internasional merupakan salah satu kriteria yang dipersyaratkan, khususnya mendukung kebijakan substitusi impor. Perangkat lunak sistem operasi dengan kehandalan tinggi dan kebutuhan sumber daya memori maupun prosesor yang minimal serta fleksibel terhadap perangkat keras maupun program aplikasi yang baru, merupakan prioritas yang harus dikembangkan. Program aplikasi juga perlu dikembangkan, terutama yang terkait dengan sektor perekonomian, industri, pendidikan, maupun pemerintahan. Dalam mempercepat pengembangan dan pendayagunaan perangkat lunak, perlu pula ditinjau implementasi konsep *open source*. Penerapan konsep *open source* ini diharapkan mampu menggalakkan industri perangkat lunak dengan partisipasi seluruh lapisan masyarakat tanpa melakukan pelanggaran hak cipta.

2.3.2 Sistem Operasi (Operating System)

Pengertian dari sistem operasi dapat dilihat dari berbagai sudut pandang. Dari sudut pandang *user*, sistem operasi dapat dipandang sebagai alat untuk mempermudah penggunaan Komputer. Dalam hal ini sistem

operasi dirancang agar mudah digunakan, dengan sedikit memperhatikan performa dan mengabaikan utilisasi sumber daya. Selain itu dalam lingkungan *multi-user*, sistem operasi juga dapat dipandang sebagai alat untuk memaksimalkan penggunaan sumber daya komputer. Akan tetapi, di beberapa komputer, sudut pandang user dapat dikatakan hanya sedikit atau tidak ada sama sekali. Misalnya *embedded computer* pada peralatan rumah tangga seperti mesin cuci dan sebagainya mungkin saja memiliki lampu indikator untuk menunjukkan keadaan sekarang, tetapi sistem operasi ini dirancang untuk bekerja tanpa campur tangan user.

Dari sudut pandang sistem, sistem operasi dapat dipandang sebagai alat yang menempatkan sumber daya secara efisien (*Resource Allocator*). Sistem Operasi adalah manager bagi sumber daya, yang menangani konflik permintaan sumber daya secara efisien. Sistem operasi juga mengatur eksekusi aplikasi dan operasi dari alat I/O. Fungsi ini dikenal juga sebagai *Control Program*. Lebih lagi, Sistem operasi merupakan suatu bagian program yang berjalan setiap saat yang dikenal dengan istilah *kernel*.

Dari sudut pandang tujuan sistem operasi, sistem operasi dapat dipandang sebagai alat yang membuat komputer lebih nyaman digunakan. (*convenient*) untuk menjalankan aplikasi dan menyelesaikan masalah pengguna. Tujuan lain sistem operasi adalah membuat penggunaan sumber daya komputer menjadi efisien.

2.3.3 Bahasa Pemrograman

Borland Delphi atau yang disebut Delphi saja, merupakan sarana pemrograman pascal atau yang kemudian juga disebut bahasa pemrograman Delphi. Delphi merupakan generasi penerus dari Turbo Pascal. Turbo Pascal yang diluncurkan pada tahun 1983 dirancang untuk dijalankan pada system operasi DOS (yang merupakan system operasi yang paling banyak digunakan pada saat itu). Sedangkan Delphi yang diluncurkan pertama kali pada tahun 1995 dirancang untuk beroperasi dibawah system operasi windows.

Kebutuhan akan adanya program aplikasi yang bekerja dibawah system operasi windows serta memiliki antarmuka visual yang *user friendly* telah memancing minat banyak orang menggunakan bahasa pemrograman yang mampu menyediakan antar muka grafis (*Graphical User Interface*).

Borland Delphi merupakan salah satu bahasa pemrograman yang semenjak diturunkan pertama kali, langsung diminati dan dilirik para programmer. Hal ini disebabkan karena Delphi menyediakan fasilitas untuk pembuatan aplikasi dengan antar muka visual secara mudah dan dapat memberikan hasil yang memuaskan.

Delphi memiliki sarana yang tangguh untuk pembuatan aplikasi, mulai dari sarana pembuatan form, menu, toolbar, hingga kemampuan untuk menangani pengelolaan basis data yang besar. Kelebihan-kelebihan yang dimiliki Delphi antara lain panel pada Delphi, form dan komponen-

komponennya dapat dipakai ulang dan dikembangkan, mampu mengakses VBX, tersedia template aplikasi dan template form, memiliki lingkungan pengembangan visual yang dapat diatur sesuai kebutuhan, menghasilkan file terkompilasi yang berjalan lebih cepat serta kemampuan mengakses basis data dari bermacam-macam format.

2.4 Komunikasi Parallel

Komunikasi parallel yang digunakan adalah komunikasi parallel lewat kabel data dari printer (saat mengeluarkan data). Pada kondisi normal(tidak aktif) tegangan pada pin-pin ini adalah 0 volt, namun bila kita beri *high*, maka tegangannya adalah 5 volt.

2.4.1 Antarmuka Port Parallel

Komunikasi parallel adalah komunikasi yang mengirim data secara bersamaan. Port printer merupakan salah satu dari fasilitas saluran data secara parallel dengan menggunakan port LPT1 dengan DB25 pin. Port Paralel digunakan untuk menghubungkan rangkaian driver dengan PC. Port ini memiliki masukan hingga 8 bit atau keluaran 12 bit pada saat yang bersamaan, dengan hanya membutuhkan rangkaian eksternal sederhana untuk melakukan suatu tugas tertentu. Port paralel ini terdiri dari 4 jalur kontrol, 5 jalur status dan 8 jalur data. Biasanya port paralel ini dapat dijumpai sebagai port printer dalam bentuk konektor DB-25 betina (*female*). Dimana fungsi masing-masing pin dapat digolongkan menjadi 3 jalur data, yaitu :

- 1) Jalur control, memiliki arah *bidirectional*
- 2) Jalur status, memiliki satu arah yaitu arah input
- 3) Jalur data, memiliki dua arah, dapat juga berfungsi sebagai pengirim alamat dan data, masing-masing 8 bit, dimana keduanya melakukan transfer data dengan protocol handshaking serta diakses dengan register berbeda. Sisa dari pin adalah sebagai ground.

Tabel 2.2 Pin-pin konektor port LPT

Nomor Pin	Nomor Bit	Jalur	Sinyal
1	1	Kontrol	Strobe
2	1	Data	Data 1
3	2	Data	Data 2
4	3	Data	Data 3
5	4	Data	Data 4
6	5	Data	Data 5
7	6	Data	Data 6
8	7	Data	Data 7
9	8	Data	Data 8
10	7	Status	Ack
11	8	Status	Busy
12	6	Status	P Error
13	5	Status	Select
14	2	Kontrol	Auto FD
15	4	Status	Fault
16	3	Kontrol	Init
17	4	Kontrol	Select In
18-25			Ground



Gambar 2.4. Konektor DB 25

BAB III

DESAIN DAN PERANCANGAN SISTEM

Dalam bab ini akan dibahas mengenai perancangan dan pembuatan perangkat keras dari kamera pendeteksi gerak serta perangkat lunak pendukungnya. Perangkat lunak yang digunakan adalah software Program Delphi 5 untuk mengolah proses pencitraan. Proses pencitraan adalah pengolahan gambar digital dengan menggunakan metode deteksi gerakan dan metode analisis nilai RGB obyek untuk proses pendeteksian letak obyek. Sedangkan perangkat keras yang digunakan meliputi webcam, motor stepper, rangkaian driver dan PC, setelah itu dilanjutkan dengan pengujian dari sistem yang akan digunakan.

3.1 Perancangan Perangkat Keras

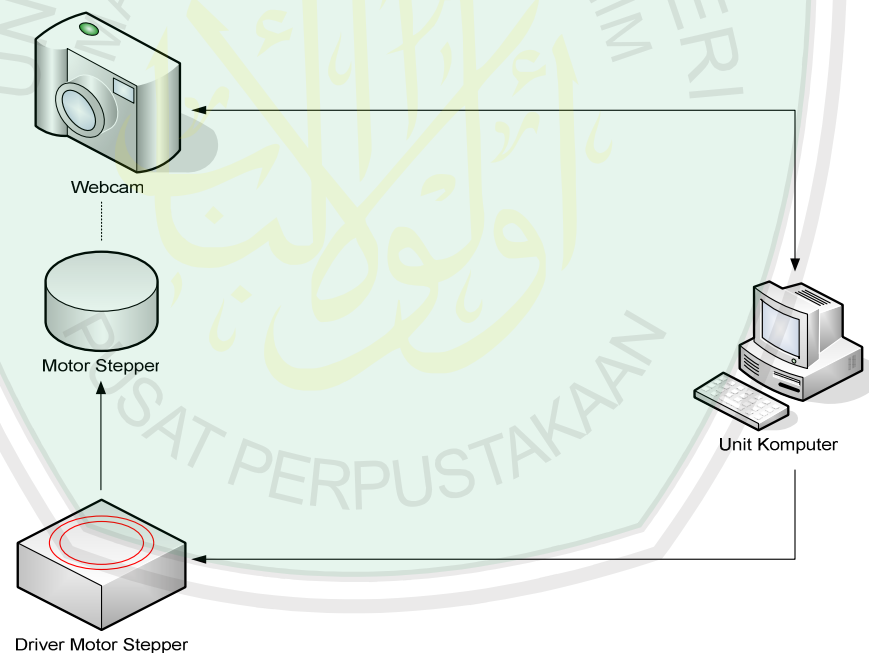
3.1.1 Pengendali Motor Stepper

Perangkat keras pengendali motor langkah dirancang untuk menggerakkan kamera. Untuk dapat menggerakkan kamera pada arah mendatar maka digunakanlah 1 unit motor langkah sebagai motor penggeraknya.

Pergerakan motor stepper akan diatur langsung oleh komputer. Sudut langkah dari motor stepper yang digunakan adalah $1,8^{\circ}$. Pada alat ini satu kali melakukan deteksi nilai RGB pada titik yang telah ditentukan input yang diberikan adalah 4, jadi sekali putar adalah $1,8^{\circ} \times 4 = 7,2^{\circ}$.

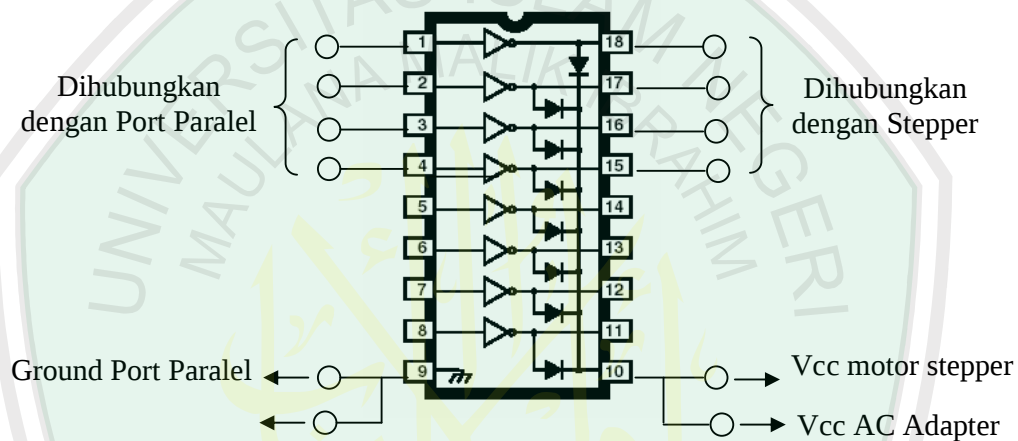
Dengan sudut putar $7,2^0$ diharapkan gerak putar stepper lebih cepat dari penyamplingan citra berikutnya. Selain itu pendeteksian terhadap obyek akan lebih detail dengan didukung dari pembacaan titik nilai RGB pada citra yang lebih banyak sehingga jarak antar piksel lebih dekat.

Obyek hanya dapat bergerak secara mendatar, ke kiri dan ke kanan karena motor stepper yang digunakan hanya satu sehingga bila terdapat obyek yang bergerak maka motor stepper hanya dapat mengikuti gerakan obyek ke kiri atau ke kanan.



Gambar 3.1 Diagram blok pengendali motor stepper

Rangkaian driver digunakan sebagai kontrol input yang akan diberikan pada motor stepper untuk menggerakkan motor stepper 4 phase. Rangkaian driver terdiri dari sebuah IC ULN2803, 4 buah header untuk menghubungkan input IC dengan port paralel, 4 buah header untuk menghubungkan output IC dengan stepper, sebagaimana disajikan pada Gambar 3.2.



Gambar 3.2 Rangkaian Driver

Rangkaian pengendali motor stepper pada Gambar 3.2 digunakan untuk motor stepper. Pengendalian gerak dari motor stepper akan dilakukan oleh perangkat lunak komputer pada unit pengendali motor stepper. Port paralel digunakan sebagai pengantarmukaan komputer dengan pengendali motor stepper.

3.1.2 Kamera

Kamera yang digunakan dalam sistem ini berfungsi untuk menangkap gambar yang akan dilihat oleh pemakai, dan juga sebagai

gambar yang akan dipindai oleh perangkat lunak untuk mendeteksi adanya gerakan.



Gambar 3.3 Logitech QuickCam
(Sumber : Logitech,2005)

Kamera yang akan digunakan dalam sistem ini adalah Logitech QuickCam. Informasi mengenai kamera ini dapat didapatkan dalam situs resminya di <http://www.Logitech.com> .

Spesifikasi kamera:

- Lensa VGA
- Kabel USB sepanjang 2,1 meter
- Manual fokus.
- Mampu menampilkan video pada resolusi 640 x 480 pixel
- Mampu menyimpan gambar berformat bitmap dengan resolusi 640 x 480.pixel
- Kecepatan frame 30 fps (pada sistem yang direkomendasikan)
- Mikropone yang sudah terpasang didalamnya (*built-in*).

Kamera ini sangat mudah dalam instalasi dan pemakaiannya, Kamera berintegrasi dengan komputer menggunakan port USB. Secara

mekanik, kamera akan terhubung dengan motor langkah untuk dapat berpindah posisi.

3.1.3 Unit Komputer

Selain pengendali motor stepper dan kamera, unit komputer juga merupakan bagian dari perangkat keras yang digunakan dalam sistem ini. Unit komputer ini juga dilengkapi pengeras suara. Spesifikasi perangkat komputer dapat dilihat pada tabel 3.4, yang dilengkapi dengan sistem operasi Windows XP Professional SP2 (5.1, Build 2600) dan DirectX 9.0c.

Spesifikasi	Keterangan
Prosesor	Intel P4-D 2,66GHz LGA
Motherboard	Asus P5RD-1
Memori	Cmos 512MB DDR1
Hardisk	Maxtor SATA-2
VGA	HIS ATI-X550 256MB, 128bit
Koneksi kamera	USB 2.0, Firewire, AV, Komposit S-video
Monitor	Samsung 15" Magic SynMaster
Speaker	Fender Bullet Reverb 38 watt

Tabel 3.1 Spesifikasi unit komputer

Spesifikasi unit komputer diatas adalah dalam pembuatan sistem ini. Akan tetapi sistem ini bisa dijalankan pada komputer standar minimum sebagaimana tercantum dibawah ini:

- Sistem operasi Windows® XP
- Pentium III 800MHz.
- Memory 128MB.
- 250 MB ruang kosong *hard drive*.

3.2 Perancangan Algoritma Deteksi Gerak

Sebelum melakukan tahap perancangan perangkat lunak, perlu dilakukan perancangan algoritma sebagai solusi permasalahan. Perancangan algoritma deteksi gerak meliputi algoritma deteksi gerak (secara umum), algoritma *screen capture*, dan algoritma deteksi gerak dengan teknik pengurangan beda antara 2 buah citra.

3.2.1 Algoritma Deteksi Gerak (Secara Umum)

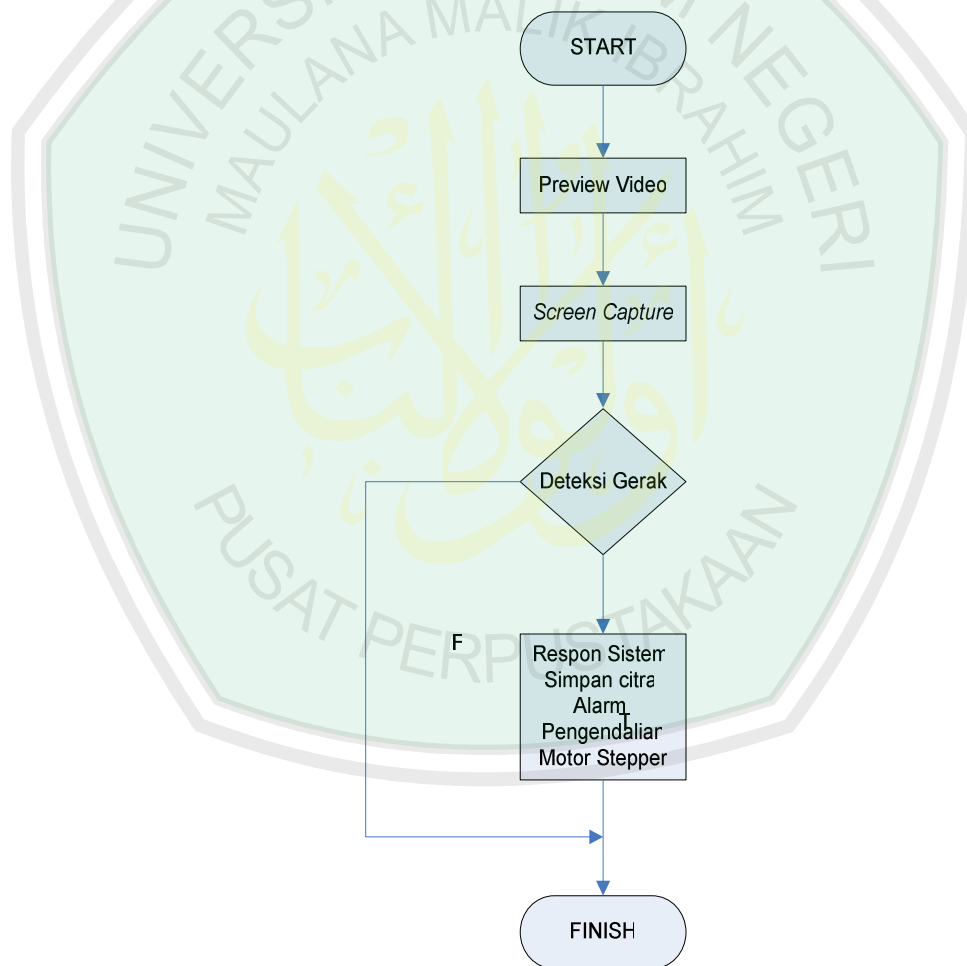
Terdapat berbagai cara dan pendekatan algoritma yang bisa dipakai untuk solusi deteksi gerak pada arus video berkelanjutan (*continuous video stream*). Dalam tugas akhir ini, untuk melakukan deteksi gerak digunakan teknik pengurangan pada bagian yang tidak bergerak dalam citra akan menghasilkan nilai RGB per piksel sama dengan nol. Langkah kerjanya adalah sebagai berikut:

1. Setelah mengaktifkan kamera video, dilakukan *preview* (tampilan) *streaming video* pada GUI. Kemudian dilakukan *screen capture* pada area tampilan video tersebut.
2. Melakukan deteksi gerak dengan pendekatan teknik pengurangan pada bagian yang tidak bergerak dalam citra akan menghasilkan nilai RGB per piksel sama dengan nol sedangkan bagian yang bergerak dalam citra memberikan nilai RGB per piksel tidak sama dengan nol, yaitu

dengan membandingkan citra terakhir yang di-*capture* dengan citra awal yang di-*capture*.

3. Merespon hasil deteksi dengan penyimpanan data citra atau pengaktifan alarm jika diperlukan.

Berdasarkan algoritma deteksi gerak maka dihasilkan diagram alir program (*flowchart*) sebagai berikut:

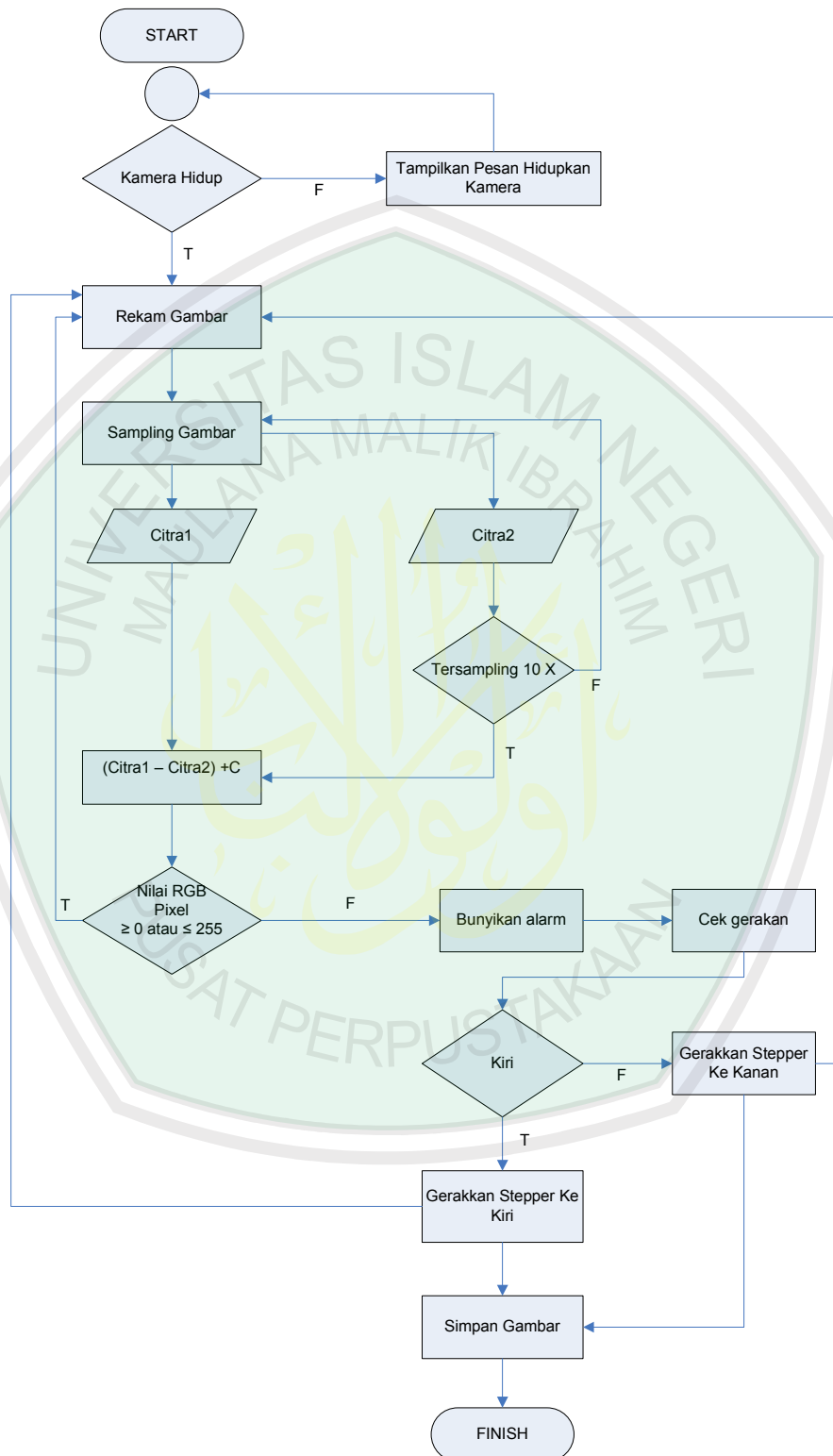


Gambar 3.4 *Flowchart* algoritma deteksi gerak secara umum

Berdasarkan cara kerja sistem operasi deteksi gerakan ini terdiri dari beberapa tahap, yaitu:

1. Mengaktifkan kamera (webcam).
2. Kamera mengambil obyek ditampilkan dalam bentuk video.
3. Hasil dari pengambilan obyek tersebut di sampling dan kemudian tampilan berbentuk gambar diam dalam format file bmp oleh delphi.
4. Proses penampilan gambar atau citra dilakukan berdasarkan pengaturan timer.
5. Timer menentukan kapan citra ditampilkan sebagai citra pertama dan kedua, proses ini dilakukan secara kontinyu.
6. Setelah citra terambil dilakukan proses perbandingan dengan menggunakan metode deteksi gerakan. Apabila ada perbedaan nilai di identifikasikan ada suatu gerakan.
7. Identifikasi ada perubahan atau tidak adalah dengan pemberian sinyal alarm.
8. Proses ini diikuti dengan gerak stepper yang mengikuti letak yang membedakan tersebut.
9. Proses ini terus berjalan sampai aplikasi dimatikan.

Dengan demikian sesuai cara kerja sistem operasi deteksi gerakan, maka menghasilkan diagram alir pada gambar 3.6 dibawah ini:



Gambar 3.5 Flowchart cara kerja sistem operasi deteksi gerak

3.2.2 Algoritma Capture Citra

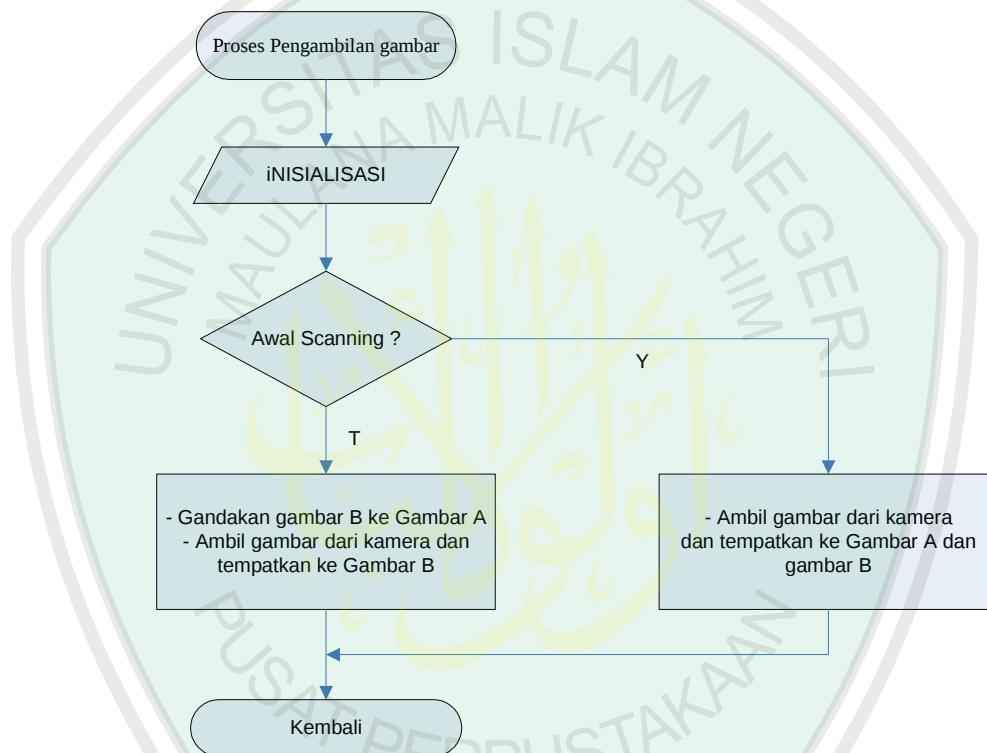
Terdapat beberapa cara untuk melakukan *capture* citra dari kamera video. Misalkan penggunaan *Dirrect-X* dan *AviCap*. Pada *AviCap*, koneksi kamera memanfaatkan *library* yang telah ada pada *Operating System* (OS) Windows berupa *avicap32.dll* dan *user32.dll*. Pada tugas akhir ini, digunakan komponen *VideoCap* (*Delphi package component*) untuk koneksi kamera serta pengaturan *driver*-nya. Sedangkan untuk proses *capture*, digunakan metode *windows device context* (*WindowDC*) memanfaatkan *library user32.dll*. *WindowDC* diatur untuk menangkap tampilan layar (*screen capture*) yang diarahkan pada posisi komponen *VideoCap*. Karena komponen *VideoCap* menampilkan *preview* gambar video dari kamera, maka hasil *screen capture*-nya adalah citra bitmap yang didapatkan dari hasil pencitraan kamera video.

Input dalam prosedur ini adalah variabel bitmap A dan bitmap B, yang mempunyai tipe bitmap. Awal scanning adalah variable bertipe *Boolean*. Dimana awal scanning akan bernilai benar pada saat unit pendeteksi gerak dijalankan untuk pertama kalinya, sesaat setelah kamera berhenti bergerak, atau pada saat reset dijalankan. Selain pada kondisi tersebut awal scanning bernilai salah.

Pada saat kondisi awal scanning bernilai benar maka prosedur ini akan mengambil bitmap dari kamera dan menempatkannya pada variabel bitmap A dan bitmap B, tetapi jika kondisi awal scanning bernilai salah

maka gambar yang *dicapture* dari kamera hanya akan ditempatkan dalam bitmap B, dimana isi dari Bitmap B telah disalin di bitmap A.

Diagram alir proses pengambilan gambar (*Capture Citra*) dapat dilihat pada Gambar 3.6. dibawah ini:



Gambar 3.6 Diagram alir proses pengambilan gambar

3.3 Perancangan Sistem Deteksi Gerak

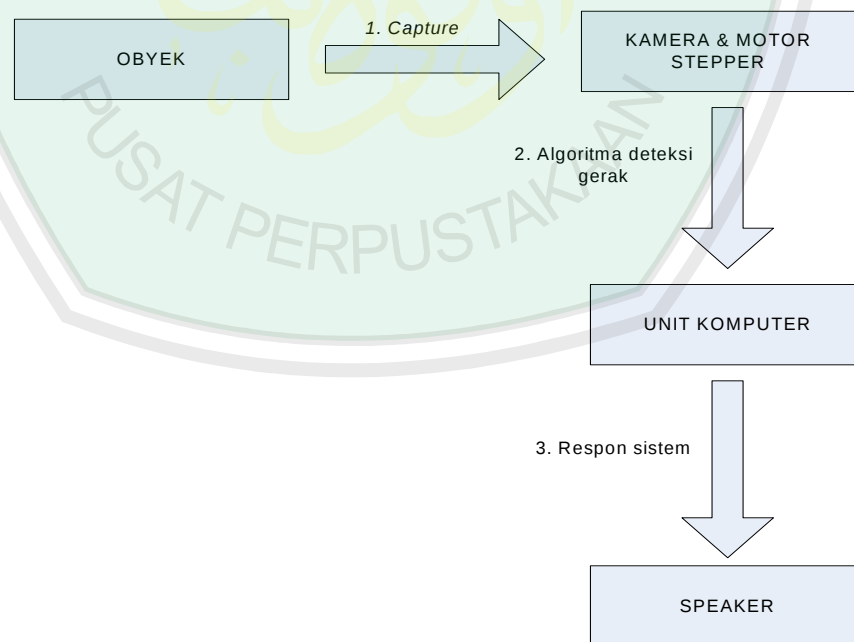
3.3.1 Tujuan Sistem Deteksi Gerak

Tujuan dari sistem deteksi gerak pada *streaming video* kamera keamanan yang akan dibangun adalah untuk mengimplementasikan algoritma deteksi gerak secara *real-time*, sesuai rancangan pada sub bab

sebelumnya. Sistem keamanan ini menggabungkan unit kamera video dengan perangkat komputer, yang diharapkan dapat digunakan sebagai sarana dan alat bantu keamanan dalam pendeteksian gerak. Sistem ini digunakan untuk mendeteksi gerakan secara *real-time*, dimana jika ada gerakan pada obyek yang diamati, pada saat yang bersamaan sistem harus mampu meresponnya dengan mengaktifkan suara peringatan maupun dengan menyimpan citra hasil deteksi sebagai data keamanan.

3.3.2 Arsitektur Sistem Deteksi Gerak

Sistem yang akan dibangun terdiri dari unit kamera video untuk pengambilan gambar, unit komputer sebagai pengolah data serta *speaker* untuk alarm keamanan. Arsitektur sistem secara umum dapat digambarkan sebagai berikut:



Gambar 3.11 Blok diagram sistem deteksi gerak

3.4 Perancangan Dan Pembuatan Perangkat Lunak

Perangkat lunak yang akan dibangun harus memenuhi ketentuan-ketentuan berikut:

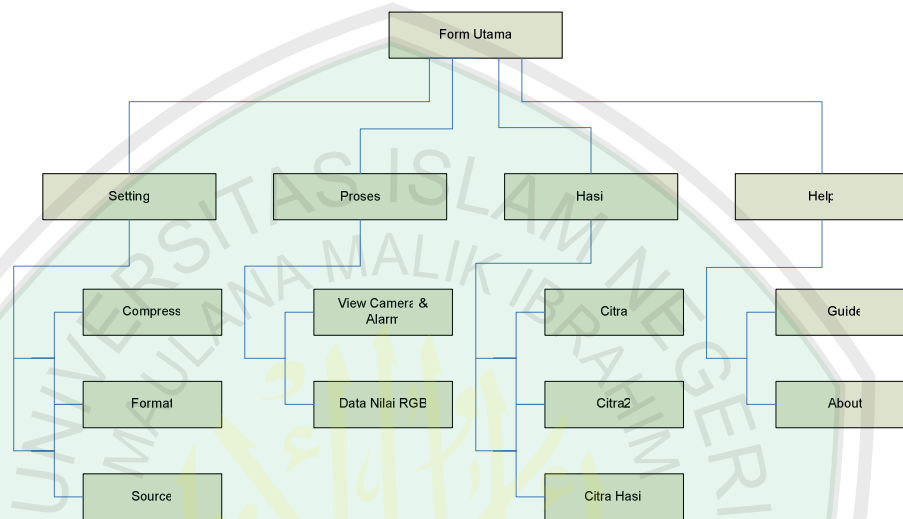
- Mampu menampilkan *streaming video* dari perangkat webcam ke panel aplikasi.
- Mampu melakukan *screen capture* dari *preview* streaming video pada panel aplikasi.
- Mampu melakukan proses deteksi gerak terhadap citra yang telah di-*capture* secara *real-time*.
- Mampu memberikan respon dari proses deteksi gerak berupa keputusan untuk menyimpan citra ke dalam data sistem dan mengaktifkan alarm jika perlu.

3.4.1 Perancangan Dan Pembuatan GUI

Tujuan dari perancangan GUI (*Graphical User Interface*) adalah untuk mendapatkan hasil kinerja maksimal dari interaksi manusia dan komputer. Tampilan GUI yang baik adalah tampilan yang memberikan kemudahan dan efisiensi bagi pengguna. GUI yang dibuat harus mendukung kontrol-kontrol sebagai berikut:

1. *Start* : untuk memulai sistem.
2. *Stop* : untuk menghentikan sistem.
3. *Close* : untuk keluar dari sistem.
4. *Setting* : untuk mengatur camera.
5. *Help* : berisi tentang keterangan program.

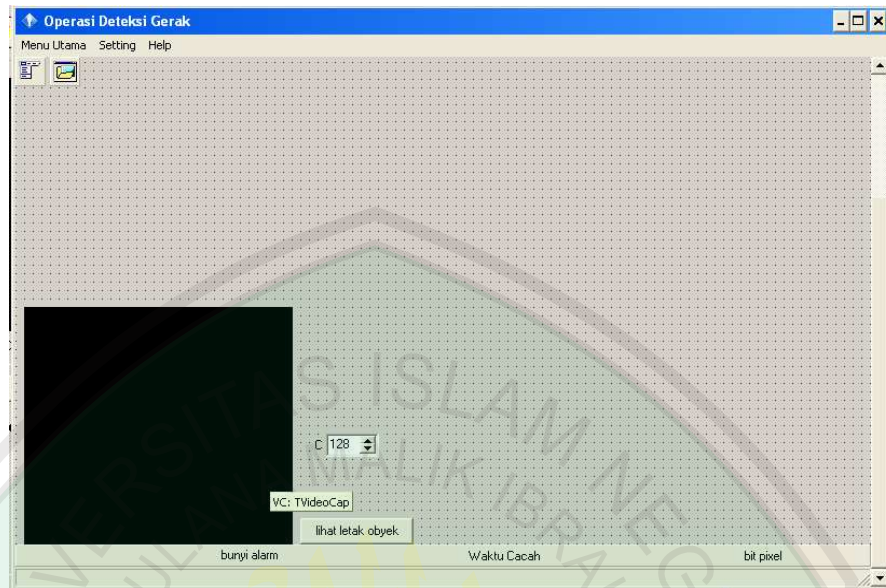
Berdasarkan uraian diatas, rancangan form aplikasi sistem deteksi gerak diperlihatkan dalam peta form berikut ini:



Gambar 3.12 Gambar peta form sistem deteksi gerak

A. Form Utama

Pada form utama terdapat kontrol sistem berupa tombol dan tampilan hasil deteksi serta analisisnya. Form utama memegang kendali terhadap form-form lainnya.



Gambar 3.13 Tampilan form utama

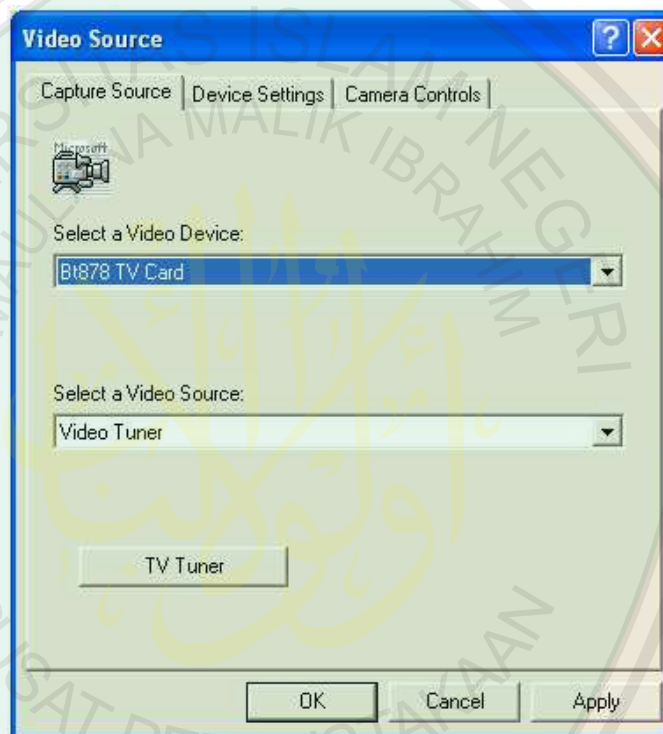
Kontrol GUI pada form utama:

- ❖ *Start*, berupa tombol deteksi dengan label “*Start*” yang berfungsi untuk memulai deteksi.
- ❖ *Stop*, berupa tombol deteksi dengan label “*Stop*” yang berfungsi untuk menghentikan proses deteksi.
- ❖ *Close*, berupa tombol keluar untuk keluar dari system.
- ❖ *Setting*, berupa *pagecontroller* camera yang ditampilkan secara *overlapping*, didalam *setting* terdapat *compress*, *format*, dan *source* ketiganya ini merupakan *setting* dari webcam.
- ❖ *Help*, berupa informasi atribut-atribut deteksi gerak.

B. Form Setting

Form *Setting* kamera merupakan panel halaman untuk pengaturan kamera secara *on-line* dari webcam. Tampilan *setting* kamera dapat dilihat pada gambar di bawah ini:

❖ Form pilihan kamera

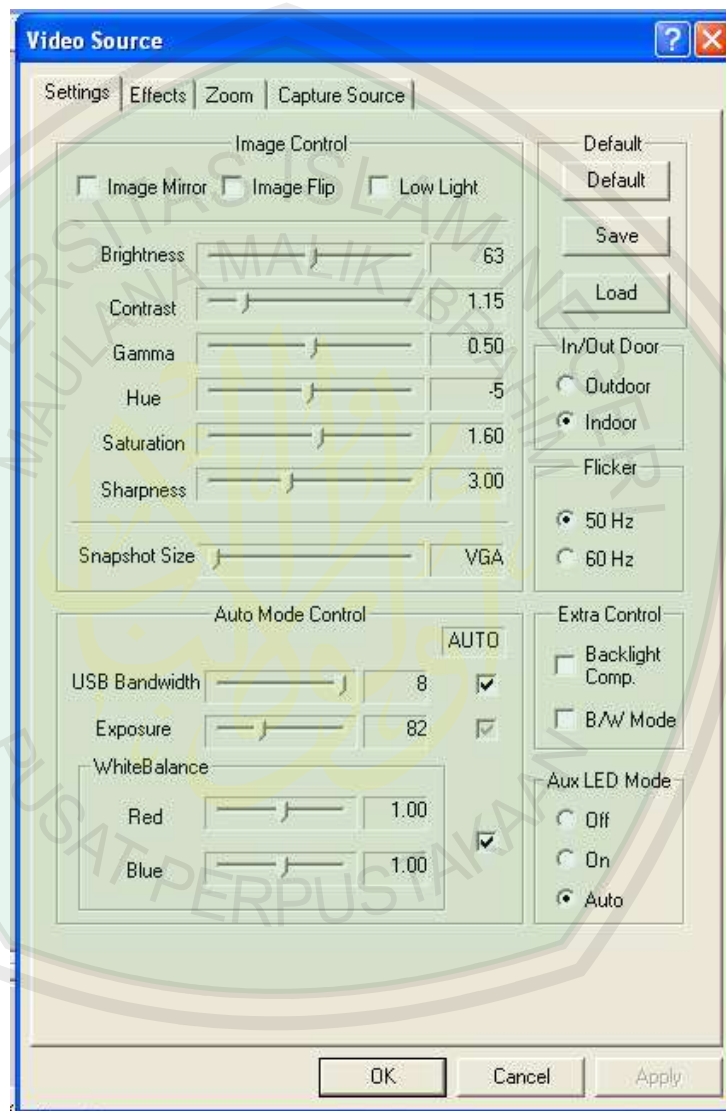


Gambar 3.14 Tampilan form pilihan kamera

❖ Form pengaturan kamera

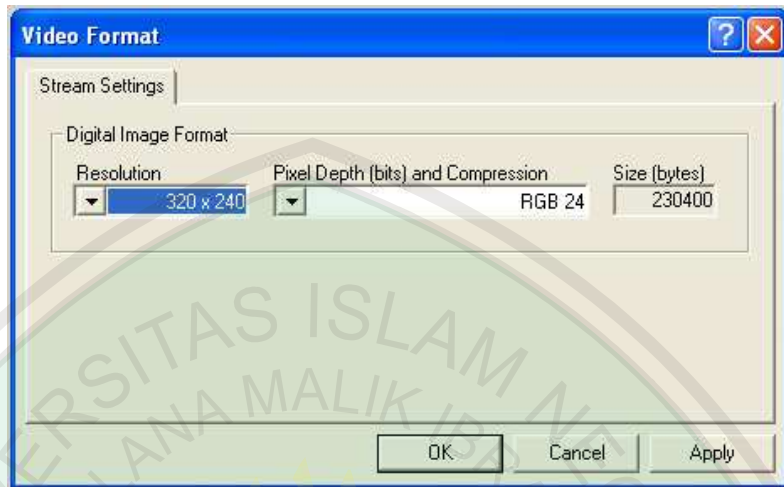
Menu digunakan untuk melakukan pengaturan kamera. Pencahayaan kamera seperti *brightness*, *contrast*, dan *saturation* dari kamera dapat diatur pada menu ini. Geser Slider pada nilai yang diinginkan, dan dapat dilihat langsung perubahan pada deteksi gerak dari

kamera. Tombol *default* dipergunakan untuk mengembalikan nilai *brightness*, *contrast*, dan *saturation* pada nilai *default*.



Gambar 3.15 Tampilan form pengaturan kamera

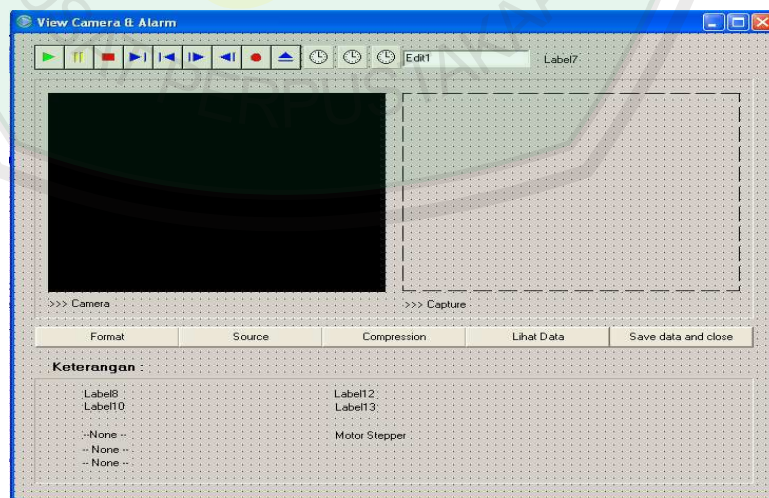
- ❖ Form pengaturan *streaming* video



Gambar 3.16 Tampilan form pengaturan *streaming* video

C. Form Proses

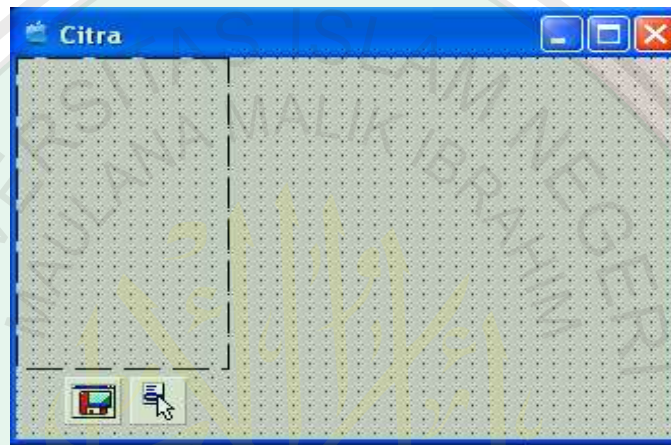
Form proses merupakan form yang mana *display* (tampilan) proses pendeteksian gerakan, dan juga menampilkan video *streaming*, didalam form proses ini meliputi respon sistem yang mana respon system ini berupa pengaktifan alarm. Berikut tampilan form proses:



Gambar 3.18 Tampilan form proses deteksi gerakan

D. Form Citra

Form citra merupakan form hasil dari pencitraan (proses) deteksi gerak, gambar yang di-*capture* akan ditampilkan pada form citra ini, mulai dari citra1, citra2, hingga citra hasil akan ditampilkan pada form citra ini.



Gambar 3.20 Tampilan form citra

BAB IV

HASIL DAN PEMBAHASAN

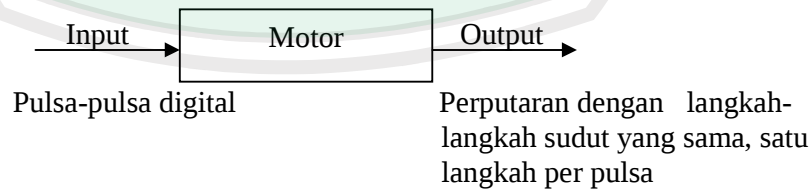
Dalam bab ini dibahas hasil yang didapat setelah dilakukan serangkaian kegiatan untuk mewujudkan sistem keamanan menggunakan kamera pendeteksi gerak dengan menggunakan webcam dan motor stepper.

4.1. Perangkat Keras

Perangkat keras yang digunakan dalam sistem ini berfungsi untuk menggerakkan kamera pada dua arah yaitu ke kanan dan ke kiri. Perangkat keras yang digunakan terdiri dari sebuah kamera, satu buah motor stepper dan rangkaian pengendalinya.

4.1.1. Aliran Data pada Motor Stepper

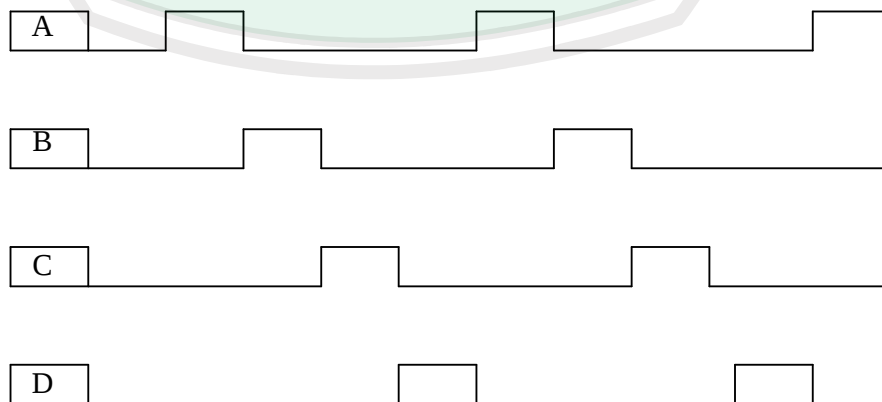
Motor Langkah (*stepper motor*) adalah motor yang menghasilkan putaran dengan sudut-sudut yang sama (atau kecepatan sudut yang konstan), sehingga motor dikatakan “melangkah”, untuk setiap pulsa yang diberikan sebagai inputnya, sebagaimana disajikan pada Gambar 4.1



Gambar 4.1 Motor Langkah

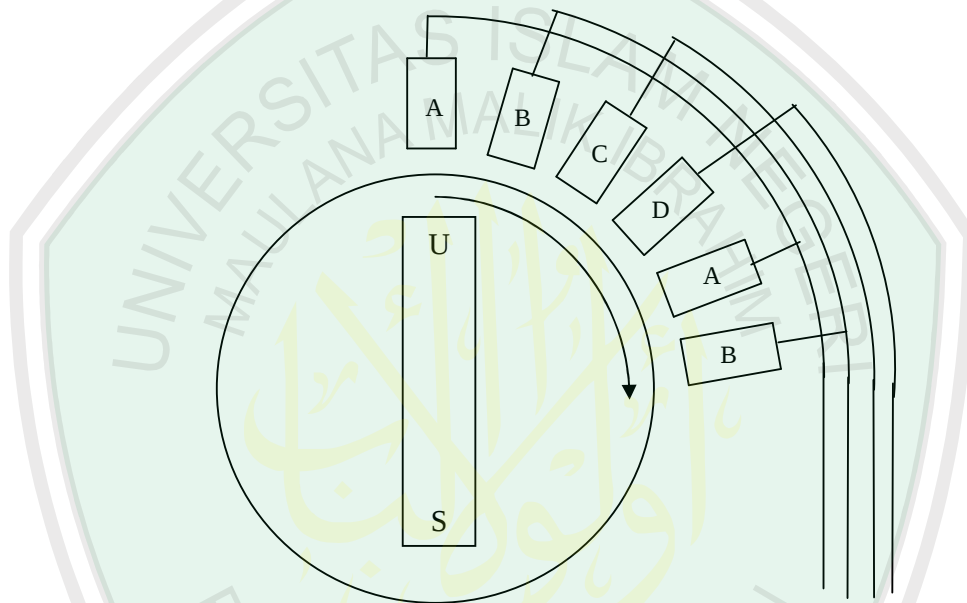
Motor stepper dapat digunakan untuk pengaturan posisi sudut yang akurat. Apabila satu pulsa input menghasilkan perputaran sejauh $1,8^{\circ}$ maka 20 pulsa yang sama akan menghasilkan perputaran sejauh $36,0^{\circ}$ dan

untuk mendapatkan satu putaran penuh sejauh 360° dibutuhkan 200 pulsa. Untuk menggerakkan sebuah motor stepper, setiap pasang kumparan stator harus disambungkan dan diputuskan dari aliran arus secara bergantian dalam urutan yang benar. Dengan demikian, input ke motor stepper berupa deretan pulsa harus menghasilkan output ke setiap pasang kumparan stator dalam urutan yang benar. Apabila kumparan stator dialiri arus sedemikian rupa, maka akan timbul medan magnet dan rotor akan berputar mengikuti medan magnet tersebut. Setiap pengalihan arus ke kumparan berikutnya menyebabkan magnet berputar menurut suatu sudut tertentu. Jumlah keseluruhan pengalihan menentukan sudut perputaran motor. Jika pengalihan arus dihentikan, maka rotor akan berhenti pada posisi terakhir. Jika kecepatan pengalihan tidak terlalu tinggi, maka slip akan dapat dihindari, sehingga tidak diperlukan umpan balik (*feedback*) pada pengendalian motor langkah ini. Motor stepper yang akan digunakan memiliki 4 fase (*pole* atau kutub), pengiriman pulsa dari komputer ke rangkaian motor langkah dilakukan secara bergantian, masing-masing 4 data (sesuai dengan jumlah phase-nya).



Gambar 4.2. Pemberian data/pulsa pada motor langkah

Pada saat yang sama, untuk tiap motor stepper, tidak boleh ada 2 (dua) masukan atau lebih yang mengandung pulsa sama dengan 1 (*high*), atau dengan kata lain, pada suatu saat hanya sebuah masukan yang bernilai 1 (satu) sedangkan lainnya bernilai 0 (nol), sebagaimana disajikan pada Gambar 4.3



Gambar 4.3 Diagram Motor Langkah

Hal pertama yang harus dilakukan dalam menggunakan alat ini adalah menghubungkan antarmuka port parallel dengan komputer, dan memastikan bahwa konektor kamera sudah terhubung dengan komputer.

4.2. Perangkat Lunak

Untuk mempermudah pengoperasiannya, maka dalam pengembangan perangkat lunak ini dirancang sedemikian rupa sehingga memudahkan dalam penggunaannya.

Bahasa pemrograman yang digunakan dalam sistem ini adalah Borland Delphi 5 beserta komponen-komponen yang dibutuhkan dalam pembuatan program deteksi gerak ini yakni, komponen videocap, driver webcam dan port I/O. Semuanya itu dapat dijalankan pada sistem operasi windows XP.

4.3. Penjelasan Program

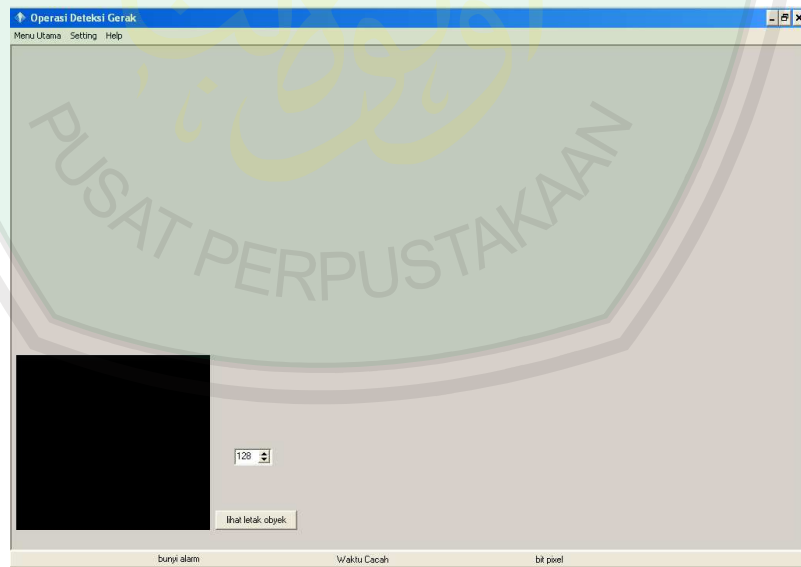
Langkah awal pembuatan program ini adalah dengan menginstall program dan software yang diperlukan dalam pembuatan program ini. Program yang diperlukan adalah Borland Delphi 5, yang berjalan pada sistem operasi windows. Proses komunikasi data antara motor stepper dengan PC tidak dapat dilakukan dengan bahasa *assembly* pada software delphi jika menggunakan operasi sistem *Windows Xp*. Untuk itu dibutuhkan penginstalan software tambahan terhadap operasi sistem tersebut untuk membantu komunikasi data ini yakni port I/O. Komunikasi data ini melalui port paralel. Selain itu juga penginstallan *component* tambahan pada borland delphi tersebut yaitu *component* videocap.

Setelah semua komponen komputer yang mendukung proses aplikasi terinstall, proses selanjutnya adalah penjelasan program. Penjelasan program merupakan petunjuk yang dapat digunakan dalam menjalankan suatu program. Pada Tugas Akhir ini akan dijelaskan mengenai penggunaan program yang ada, antara lain: Menu utama, Proses deteksi gerakan.

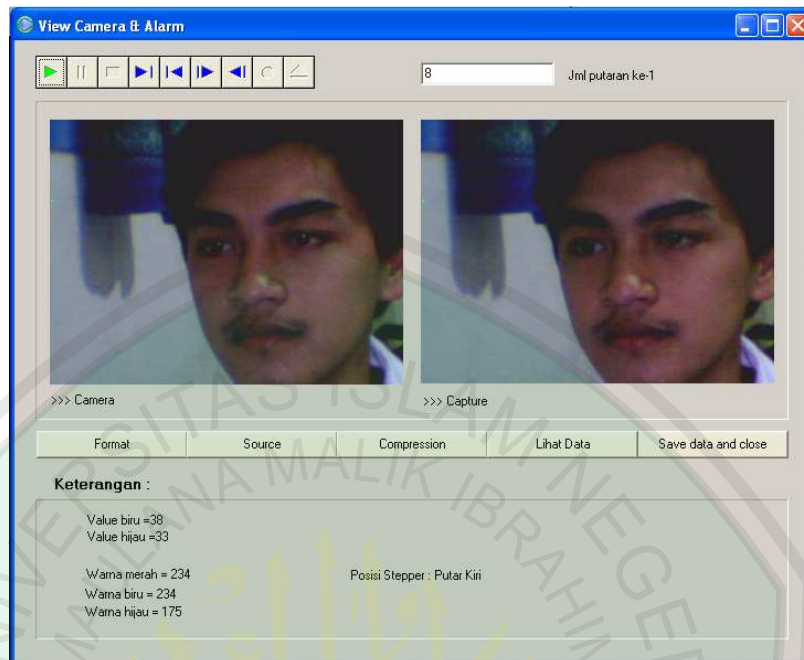
4.4. Sistem Kerja Software

Pada proses Deteksi Gerakan, bila terdapat perbedaan gerakan antara citra1 dan citra2 maka akan ditampilkan pada citra hasil. Perbedaan gerakan yang ditampilkan oleh citra hasil di indikasikan dengan pemberian sinyal alarm dan pergerakan motor stepper, di mana motor stepper tersebut akan bergerak mengikuti gerakan obyek.

Bunyi alarm memanfaatkan komponen media player pada delphi yang ditampilkan pada form1. Ketika form1 aktif, form utama di nonaktifkan kemudian dilakukan penyamplingan lagi pada form1 untuk melakukan pendeteksian letak obyek dengan menggunakan metode pembacaan nilai RGB pada titik-titik tertentu. Berikut *layout* form utama dan juga *layout* form1



Gambar 4.4 Layout Form Utama



Gambar 4.5 Layout Form 1

Pada sistem kerja alat ini setelah terdeteksi gerakan kemudian dilanjutkan dengan pendeteksian letak obyek yang diiringi bunyi alarm. Untuk mengetahui letak obyek tersebut maka obyek di sampling menjadi gambar diam untuk dideteksi nilai RGB (*Red, Green, Blue*) pada koordinat yang telah ditentukan.

Langkah awal sebelum menyampling adalah mengaktifkan Form1 terlebih dahulu yang berisikan media player, timer1, dan *component* video cap. Timer1 berisikan perintah untuk menyampling sekaligus mendeteksi letak obyek yang masuk. Berikut adalah Listing program untuk pengaktifan Form1 dan Timer1.

(prosedur pengaktifan form1)

```
procedure TForm1.FormActivate(Sender: TObject);
var
s:string;
```

```
begin  
memo1:=0;
```

(video cap di FormUtama di nonaktifkan)

```
formutama.vc.DriverOpen := false;  
formutama.vc.VideoPreview := false;  
/  
getdir(0,s);
```

(pemanggilan file alarm)

```
mp.FileName := s + '\lea.mp3';  
mp.Open;
```

(pengaktifan alarm)

```
mp.Play;  
mp.AutoRewind:=true;
```

(pengaktifan video cap di form1)

```
vc.VideoPreview := true;  
vc.DriverOpen := true;
```

(pengaktifan timer1 di form1)

```
timer1.Enabled := true;  
jmlPutar := 0;  
end;
```

(prosedur timer1)

```
procedure TForm1.Timer1Timer(Sender: TObject);  
var f,s : string;  
i, merah, hijau, biru, posisix : integer;  
begin  
label13.Caption:=floattostr(g);
```

(penyimpanan citra)

```
nama_file := formatdatetime('hh-mm-ss_dd-mm-yyyy',now())+'.bmp';  
vc.SingleImageFile := nama_file;  
vc.SaveAsDIB;  
getdir(0,s);  
f:= s+'\'+nama_file;;
```

(penampilan citra yang tersampling)

```
image1.Picture.LoadFromFile(f);
```

Proses pendeteksian letak obyek pada alat ini diikuti dengan gerak motor stepper yang dikontrol dalam timer2 dan tgerak untuk gerak ke kanan dan kiri. Berikut adalah Listing program untuk pemanggilan gerak motor stepper.

(pendeteksian letak obyek)

```
{warna RGB titik 20,120}  
merah := getRValue(image1.Canvas.Pixels[20,120]);
```

```

if (merahlebih besar dari230) then
begin
    label2.caption := 'Warna merah = '+inttostr(merah);
    label3.caption := 'Posisi Stepper : Putar Kiri';

```

(pemanggilan gerak stepper)

```

    timer2.Enabled:=true;
    Tgerak.Enabled :=false;
end;
biru := getBValue(image1.Canvas.PikseIs[20,120]);
label8.Caption := 'Value biru =' +inttostr(biru);
if (merahlebih besar dari230) then
begin
    label9.caption := 'Warna biru = '+inttostr(biru);
    label3.caption := 'Posisi Stepper : Putar Kiri';

```

(pemanggilan gerak stepper)

```

    timer2.Enabled:=true;
    Tgerak.Enabled :=false;
end;
hijau := getGValue(image1.Canvas.PikseIs[20,120]);
label10.Caption := 'Value hijau =' +inttostr(hijau);
if (merahlebih besar dari230) then
begin
    label11.caption := 'Warna hijau = '+inttostr(hijau);
    label3.caption := 'Posisi Stepper : Putar Kiri';

```

(pemanggilan gerak stepper)

```

    timer2.Enabled:=true;
    Tgerak.Enabled :=false;
end;
form2.Memo1.Lines.Add(inttostr(merah)+' '+inttostr(hijau)+' '+inttostr(biru));
inc(memo1);

```

(Prosedur timer2 untuk gerak stepper ke kiri)

```

procedure TForm1.TGerakTimer(Sender: TObject);
var
    x:byte;
    putar:array[1..4] of byte;

```

```

begin
    putar[1]:=$01;
    putar[2]:=$02;
    putar[3]:=$04;
    putar[4]:=$08;
    inc(i);
    x:=putar[i];

```

(menentukan alamat port paralel)

```

outport($378,x);
Edit1.Text:=inttostr(inport($378));
if i=4 then
begin
    i:=0;
    inc(jmlPutar);

```

```
label7.Caption := 'Jml putaran ke-' + inttostr(jmlPutar);
if jmlPutar = 1 then NonAktif;
end;
end;
```

(Prosedur Tgerak untuk gerak stepper ke kanan)

```
procedure TForm1.Timer2Timer(Sender: TObject);
var
x:byte;
putar:array[1..4] of byte;
begin
putar[1]:=$08;
putar[2]:=$04;
putar[3]:=$02;
putar[4]:=$01;
inc(i);
x:=putar[i];
```

(menentukan alamat port paralel)

```
outport($378,x);
Edit1.Text:=inttostr(inport($378));
if i=4 then
begin
i:=0;
inc(jmlPutar);
label7.Caption := 'Jml putaran ke-' + inttostr(jmlPutar);
if jmlPutar = 1 ;
end;
```

4.5. Mendeteksi nilai RGB pada titik tertentu

Pendeteksian letak benda dilakukan dengan metode pendeteksian nilai RGB pada koordinat tertentu, dengan parameter jika nilai R, G atau B lebih besar dari 230. Nilai 230 digunakan sebagai nilai parameter, dimana dengan menggunakan obyek bercahaya intensitas cahayanya mampu memberikan nilai lebih besar dari 230 pada salah satu dari nilai R, G atau B .

Pada tabel di bawah ini menunjukkan koordinat piksel yang dideteksi nilai RGBnya, dengan parameter nilai RGB pada piksel tersebut lebih besar dari 230 maka dapat diidentifikasi letak obyek disebelah kiri

atau kanan. Pada koordinat piksel (20,120) hingga (130,120) menunjukkan bahwa identifikasi letak obyek berada di sebelah kiri dan pada koordinat piksel (190,120) hingga (300,120) menunjukkan bahwa identifikasi letak obyek berada di sebelah kanan.

Misalnya, pada koordinat piksel (30,120) terdapat nilai RGB lebih besar dari 230 maka akan teridentifikasi letak obyek berada di sebelah kiri, sehingga motor stepper akan bergerak ke kiri mengikuti gerakan obyek. Sedangkan pada koordinat piksel (200,120) terdapat nilai RGB lebih besar dari 230 maka akan teridentifikasi letak obyek berada di sebelah kanan, sehingga motor stepper akan bergerak ke kanan mengikuti gerakan obyek.

Koordinat piksel (x,y)	Identifikasi letak obyek
20,120	Kiri
30,120	Kiri
40,120	Kiri
50,120	Kiri
60,120	Kiri
70,120	Kiri
80,120	Kiri
90,120	Kiri
100,120	Kiri
110,120	Kiri
120,120	Kiri
130,120	Kiri
140,120	Diam
150,120	Diam
160,120	Diam
170,120	Diam
180,120	Diam
190,120	Kanan
200,120	Kanan
210,120	Kanan
220,120	Kanan
230,120	Kanan
240,120	Kanan

250,120	Kanan
260,120	Kanan
270,120	Kanan
280,120	Kanan
290,120	Kanan
300,120	Kanan

Tabel 4.1 Koordinat piksel yang digunakan untuk mengidentifikasi letak obyek.

Dari tabel di atas dapat dilihat bahwa sumbu y selalu tetap yakni 120 dan sumbu x terpaut 10 pada masing-masing kondisi (kiri atau kanan). Nilai 120 diperoleh dari titik koordinat tengah sumbu y pada citra, yang mana ukuran citra adalah 320 x 240. Pada koordinat piksel (140,120) hingga (180,120) menunjukkan bahwa identifikasi pemanggilan gerak stepper tidak dijalankan yang artinya stepper diam dan program akan kembali merekam gambar.

4.1.2 Deteksi gerakan yang disebabkan oleh obyek yang dimasukkan

Pengambilan data dilakukan dengan 2 cara, yaitu memasukkan obyek pada saat posisi awal obyek berada di sebelah kiri yang dilakukan sebanyak 6 kali percobaan dan memasukkan obyek pada saat posisi awal obyek berada di sebelah kanan yang juga dilakukan sebanyak 6 kali percobaan. Dalam pengambilan data pendeteksian gerakan diperoleh dari obyek yang dimasukkan dan ada pula yang diperoleh dari gerakan obyek lain (misalnya: benda lain yang bergerak) yang menyebabkan deteksi gerakan aktif.

Apabila obyek berada pada koordinat tertentu dan salah satu dari nilai R, G atau B pada koordinat tersebut lebih besar dari 230, maka motor stepper akan bergerak mengikuti letak obyek. Pada tabel yang disajikan di bawah ini berisi data yang memenuhi syarat untuk memanggil prosedur stepper. Misalnya: pada tabel 4.2, pada urutan penyamplingan citra ke-126 terdapat nilai RGB yang lebih besar dari 230 di koordinat piksel (20,120), (30,120) dan (40,120) pada obyek yang bergerak dari arah kiri. Pada tabel 4.3, pada urutan penyamplingan citra yang ke-2 terdapat nilai RGB yang lebih besar dari 230 di koordinat piksel (240,120) dan (250,120) pada obyek yang bergerak dari arah kanan.

No	Urutan penyamplingan citra	Koordinat piksel
1	122	20,120
2	123	30,120
3	126	(20,120),(30,120),(40,120)
4	127	(50,120),(60,120),(70,120)
5	128	(70,120),(80,120),(90,120)
6	129	(90,120),(100,120),(110,120)
7	130	100,120
8	131	(110,120),(120,120),(130,120)
9	132	(130,120),(140,120),(150,120)
10	133	(140,120),(150,120),(170,120)
11	134	(170,120),(180,120),(190,120)
12	135	(180,120),(190,120)
13	136	(190,120),(200,120)
14	137	(150,120),(160,120),(180,120),(190,120)
15	138	(140,120),(150,120)
16	139	(120,120),(130,120),(140,120)
17	140	(110,120),(130,120)
18	141	(120,120),(130,120)
19	142	140,120
20	143	(150,120),(160,120)
21	144	(150,120),(160,120)
22	145	(150,120),(160,120)
23	146	(160,120),(170,120)
24	147	(160,120),(170,120)
25	148	(170,120),(180,120)
26	149	(170,120),(180,120)

27	150	(180,120),(190,120)
28	151	(180,120),(190,120)
29	152	(190,120),(200,120)
30	153	(190,120),(200,120)
31	154	(170,120),(180,120),(190,120)
32	155	(160,120),(170,120)(180,120)
33	156	(150,120),(160,120),(170,120)
34	157	(150,120),(160,120)
35	158	160,120
36	159	170,120
37	160	170,120
38	161	170,120
39	162	(170,120),(180,120)
40	163	180,120
41	164	180,120
42	165	(180,120),(190,120)
43	166	(180,120),(190,120)
44	167	(170,120),(180,120),(190,120)
45	168	(170,120),(180,120)
46	169	(150,120),(160,120)
47	170	(150,120),(160,120)
48	171	160,120
49	172	160,120
50	173	160,120
51	174	160,120
52	175	160,120
53	176	160,120
54	177	(160,120),(170,120)
55	178	170,120
56	179	170,120
57	180	170,120
58	181	(170,120),(180,120)
59	182	180,120
60	183	180,120
61	184	(180,120),(190,120)
62	185	190,120
63	186	(170,120),(180,120),(190,120)
64	187	(170,120),(180,120),(190,120)
65	188	(160,120),(170,120),(180,120)
66	189	150,120
67	190	160,120
68	191	160,120
69	192	160,120
70	194	160,120

Tabel 4.2 Koordinat piksel dengan nilai RGB lebih besar dari 230 pada

obyek yang bergerak dari arah kiri

No	Urutan penyamplingan citra	Koordinat piksel
1	1	(250,120),(260,120)
2	2	(240,120),(250,120)
3	3	240,120
4	4	(210,120),(220,120),(230,120),(240,120)
5	5	(200,120),(210,120),(220,120)
6	6	(180,120),(190,120),(200,120)
7	7	(160,120),(170,120),(180,120)
8	8	(160,120),(170,120),(180,120)
9	9	(140,120),(160,120),(160,120)
10	10	140,120
11	11	(130,120),(140,120)
12	12	130,120
13	13	(120,120),(130,120)
14	14	120,120
15	15	(110,120),(120,120)
16	16	(120,120),(130,120)
17	17	(140,120),(150,120)
18	18	(150,120),(160,120)
19	19	150,120
20	20	150,120
21	21	140,120
22	22	140,120
23	23	130,120
24	24	130,120
25	25	130,120
26	26	(130,120),(140,120)
27	27	(130,120),(140,120)
28	28	(140,120),(150,120)
29	29	140,120
30	30	(130,120),(140,120)
31	31	130,120
32	32	(120,120),(130,120)
33	33	(120,120),(130,120),(140,120)
34	34	(130,120),(140,120),(150,120)
35	35	(130,120),(140,120),(150,120)
36	36	140,120
37	37	140,120
38	38	140,120
39	39	140,120
40	40	(120,120),(130,120),(140,120)
41	41	(130,120),(140,120),(150,120)
42	42	(150,120),(160,120)
43	43	(150,120),(160,120)
44	44	(150,120),(160,120)
45	45	150,120
46	46	(140,120),(150,120)
47	47	(140,120),(150,120)

48	48	140,120
49	49	140,120
50	50	(130,120),(140,120)
51	51	130,120
52	52	130,120
53	53	130,120
54	54	(130,120),(140,120),(150,120)
55	55	160,120
56	56	180,120
57	57	(170,120),(180,120)
58	58	(170,120),(180,120)
59	59	170,120
60	60	170,120
61	61	170,120
62	62	(160,120),(170,120)
63	63	160,120
64	64	160,120
65	65	(150,120),(160,120)
66	66	(150,120),(160,120)
67	67	(150,120),(160,120)
68	68	150,120
69	69	(140,120),(150,120)
70	70	(140,120),(150,120)
71	71	140,120
72	72	130,120
73	73	140,120
74	74	(120,120),(130,120)
75	75	(120,120),(130,120)
76	76	(120,120),(130,120),(140,120)
77	77	(120,120),(130,120),(140,120)
78	78	(130,120),(140,120),(150,120)
79	79	140,120
80	80	(130,120),(140,120)
81	81	(130,120),(140,120)
82	82	130,120
83	83	130,120
84	84	(140,120),(150,120)
85	85	(150,120),(160,120),(170,120)
86	86	(130,120),(140,120),(170,120)

Tabel 4.3 Koordinat piksel dengan nilai RGB lebih besar dari 230 pada obyek yang bergerak dari arah kanan.

Cara pembacaan data di atas berdasarkan kolom urutan. Urutan yang dimaksudkan adalah urutan penyamplingan citra setelah masuk

pendeteksian letak obyek yang kemudian dianalisis nilai RGBnya pada masing-masing koordinat yang telah ditentukan. Jika salah satu nilai komponen RGB mengandung nilai lebih besar dari 230 maka pemanggilan prosedur gerak stepper aktif. Apabila dalam satu urutan terdapat lebih dari satu koordinat yang memenuhi syarat maka pembacaan dalam program adalah pada urutan koordinat yang terbesar. Ini disebabkan oleh sub rutin program dalam pemanggilan prosedur gerak stepper. Dalam delphi pemanggilan prosedur jika parameter pemanggilan prosedur tersebut lebih dari satu maka pemanggilan yang dijalankan adalah urutan sub rutin program yang paling terakhir

4.1.3 Pembahasan

4.1.3.1 Skenario Pengujian Program Jarak Benda dengan Kamera

Tabel 4.5 Hasil uji coba

NO	Percobaan	Panjang gelombang cahaya	Keterangan
1	Pertama	0.1 nm	Jarak benda 10 cm
2	Kedua	1 nm	Jarak benda 1 m
3	Ketiga	2 nm	Jarak benda 2 m
4	Keempat	3.5 nm	Jarak benda 3.5 m
5	Kelima	5 nm	Jarak benda 5 m

Skenario yang pertama adalah pengujian program dilaksanakan pada sebuah kamar, sebagaimana umumnya sebuah kamar, kamar dilengkapi sebuah jendela, yang mana apabila pada siang hari keadaan

kamar akan terlihat terang. Pengujian pertama ini dilakukan pada malam hari, setelah tombol start dijalankan kemudian jarak antara obyek yang dideteksi dengan kamera cukup dekat yakni 10 cm. Dari skenario yang pertama ini dapat menghasilkan kesimpulan bahwa obyek dapat diterima dengan baik oleh kamera, dan gambar dapat disimpan, sedangkan motor stepper juga dapat bekerja dengan normal sesuai dengan harapan yakni bergerak ke kanan dan ke kiri mengikuti benda yang bergerak. Sedangkan alarm juga berfungsi sebagaimana mestinya yakni ketika terdeteksi gerakan alarm akan dipanggil dan akan berbunyi, bunyi alarm akan berhenti sampai tidak ada pergerakan lagi, dan juga bunyi alarm dapat dihentikan dengan manual, karena pada program ini dilengkapi menu media player sebagai pengontrol file alarm, begitupun seterusnya berulang-ulang sampai program deteksi gerak dihentikan, dan menghasilkan panjang gelombang cahaya 0.1 nm. Adapun untuk memperoleh panjang gelombang cahaya, maka bisa menggunakan rumus perhitungan panjang gelombang cahaya berikut ini :

$$\lambda = V/f \Rightarrow \lambda = V \cdot T$$

λ = Panjang gelombang cahaya (nm).

V = Laju gelombang (m/s).

f = frekuensi (Hz).

T = Periode (s).

Sedangkan untuk memperoleh nilai f adalah :

$$F = 1/T$$

Dan juga untuk memperoleh nilai dari laju gelombang (V) adalah :

$$V = s/t$$

t = waktu cahaya untuk menempuh sampai ke benda (s).



Tabel 4.6 matriks gambar 4.7 dengan satuan 20 x 15 pixel

	69	109	135	149	152	146	131	117	142	152	151	150	149	148	141	139	132	126	119
	73	77	77	90	104	113	139	129	136	165	163	160	162	158	152	149	145	138	131
	57	27	35	58	62	54	67	109	147	178	175	170	171	166	162	159	154	147	142
	60	75	100	117	101	70	73	93	164	191	186	181	178	173	170	166	161	154	148
	123	180	195	211	198	172	118	108	178	198	195	189	185	182	177	173	169	160	153
	159	202	211	216	212	208	145	128	177	205	202	196	192	190	184	178	174	164	156
	187	201	190	216	235	228	154	144	185	206	206	202	198	196	190	183	179	169	160
	159	155	156	214	236	217	131	142	189	208	208	207	204	200	194	187	183	173	164
	159	155	175	240	234	213	146	146	196	210	209	207	205	203	197	190	185	176	167
	185	178	189	244	229	207	192	159	193	207	208	206	203	201	196	189	184	176	167
	178	162	171	224	219	206	191	157	191	204	205	204	202	200	194	187	181	174	166
	157	109	116	124	124	138	149	152	189	198	199	199	197	193	189	183	177	173	166
	83	40	73	69	58	67	87	133	181	189	192	191	190	185	183	179	173	169	164
	36	39	40	46	44	43	66	130	174	179	182	181	180	177	176	171	166	163	157

Skenario yang kedua adalah jarak benda yang bergerak dengan kamera 1 meter. Adapun proses dan langkah pengujian sama dengan skenario yang pertama dan disimpan dalam bentuk dan format yang sama juga. Hasil dari skenario yang kedua ini motor stepper dan alarm dapat berfungsi sebagaimana mestinya. Sedangkan panjang gelombang cahaya menghasilkan 1 nm. Berikut gambar objek dengan jarak 1 meter :



Gambar 4.9 Gambar benda dengan kamera berjarak 1 meter.

Tabel 4.8 matriks gambar 4.9 dengan satuan 20 x 15 pixel

	31	40	137	70	29	106	68	52	41	53	126	24	103	172	130	135	238	216	223
30	39	132	72	23	72	78	41	71	65	153	89	67	173	144	139	235	233	236	
30	38	132	113	38	22	35	77	182	177	136	131	87	161	171	155	228	239	242	
31	38	121	116	53	19	26	113	200	204	166	114	171	152	141	162	224	237	242	
31	37	107	98	56	20	36	129	196	201	209	198	183	171	148	159	232	240	241	
32	35	79	60	42	29	44	134	184	199	209	215	179	189	149	159	244	242	241	
33	32	55	43	25	33	44	138	181	198	204	220	191	191	150	164	247	242	241	
30	27	72	43	19	24	40	134	168	178	193	207	208	180	137	168	248	241	241	
54	48	60	27	21	27	35	76	72	83	103	163	213	190	137	169	236	229	232	
132	114	64	20	26	30	29	54	49	57	62	152	225	186	120	169	221	218	220	
141	110	43	24	28	31	30	51	47	51	56	146	223	169	104	163	252	241	236	
123	66	28	29	30	31	30	40	42	44	47	146	222	142	89	156	237	227	223	
52	36	29	31	32	29	33	39	36	42	40	122	171	137	67	151	224	214	208	
31	31	30	32	33	28	52	65	31	36	36	107	139	111	52	145	198	183	179	

Tabel 4.9 matriks gambar 4.10 dengan satuan 20 x 15 pixel

	35	45	140	84	50	136	103	132	216	195	136	157	173	166	132	140	238	216	223	
	31	41	136	85	45	106	133	162	227	203	118	115	179	173	144	142	236	233	236	
	30	38	134	123	62	64	103	200	249	246	181	154	175	160	172	157	228	239	242	
	32	39	126	125	79	76	99	177	236	165	87	49	139	159	135	160	225	237	242	
	33	39	113	109	91	77	78	161	148	82	76	34	95	187	107	147	235	240	241	
	37	36	84	74	83	76	63	157	93	126	157	98	82	196	122	155	245	242	241	
	38	32	60	59	58	66	60	177	170	147	150	128	120	198	137	163	247	242	241	
	33	26	74	56	43	45	53	176	205	155	150	144	171	174	109	168	249	241	241	
	57	50	66	40	39	42	51	166	204	142	129	128	185	184	110	170	238	229	232	
	135	118	71	29	35	37	42	162	210	86	55	134	220	179	106	171	221	218	220	
	142	113	48	25	30	34	38	145	191	94	49	172	225	158	95	168	251	242	236	
	127	68	30	26	27	30	38	136	164	102	54	204	207	130	82	161	238	228	223	
	55	37	30	26	27	28	42	130	130	68	76	183	154	129	63	156	225	215	209	
	31	31	30	26	25	24	58	175	174	114	110	170	131	106	49	149	199	185	180	

Tabel 4.10 matriks gambar 4.11 dengan satuan 20 x 15 pixel

[illegible]

Skenario yang kelima adalah jarak benda yang bergerak dengan kamera antara 5 m. Waktu uji coba yang kelima ini pada siang hari. Pada skenario ini bisa didapatkan hasil bahwa apabila sudah terdeteksi gerakan, benda yang terdeteksi gerakan akan di-*capture* dan disimpan dengan baik. Dan alarm dapat berfungsi sebagaimana mestinya. Akan tetapi motor stepper tidak dapat berfungsi dengan baik, hal ini dikarenakan intensitas cahaya yang ditangkap oleh kamera kurang, karena prosedur yang dipakai untuk menggerakkan motor stepper adalah dengan perhitungan nilai dari intensitas nilai RGB pada sebuah gambar yang di-*capture*. Sedangkan panjang gelombang cahaya yang didapatkan adalah 5 nm.

Adapun format penyimpanan gambar sama seperti skenario yang pertama, baik format gambar ataupun nama file gambar yang disimpan. Berikut hasil gambar dari skenario kedua deteksi deteksi gerak.



Gambar 4.12 Hasil gambar setelah terdeteksi gerakan pada jarak 5 m

	146	149	152	156	158	159	161	163	164	162	176	191	188	184	179	178	180	183	185				
	149	152	156	158	159	161	168	172	169	165	181	195	192	187	183	182	184	187	190				
	151	154	158	160	162	163	109	96	152	170	186	198	194	190	186	186	188	191	193				
	152	156	158	160	168	130	43	35	68	169	190	200	196	191	188	188	189	203	215				
	153	156	158	160	166	144	117	90	88	170	190	200	195	190	186	184	188	207	221				
	152	156	158	162	163	154	129	106	152	172	190	200	194	187	184	181	186	196	200				
	151	154	161	137	66	54	119	79	115	170	193	198	191	183	183	181	187	196	199				
	150	153	161	84	17	26	124	77	22	82	189	194	186	175	170	171	182	196	201				
	147	150	153	61	36	74	121	73	29	44	181	188	179	167	161	164	174	204	217				
	143	151	121	25	95	169	83	52	30	44	174	180	171	161	156	162	170	185	189				
	141	150	97	20	56	75	31	31	30	42	166	172	163	151	144	158	165	175	176				
	138	143	132	85	30	26	31	32	30	41	161	166	159	143	135	154	160	170	172				
	134	137	146	93	28	32	32	32	30	41	157	162	155	137	130	150	156	165	167				
	129	133	137	62	27	32	32	32	30	41	154	157	149	131	127	146	152	161	162				



Gambar 4.13 Hasil gambar yang bergerak kiri

	148	150	154	156	158	160	161	163	164	162	176	190	188	183	178	178	180	182	184		
	150	160	158	158	160	162	163	165	166	166	181	194	191	187	183	182	185	186	189		
	61	99	151	162	162	164	165	167	168	168	186	197	195	190	187	186	189	190	193		
	30	18	100	169	164	165	166	168	169	170	189	199	196	192	189	189	191	202	215		
	55	30	109	168	163	164	166	168	170	171	192	199	195	193	190	189	191	206	221		
	93	67	148	164	163	163	166	168	170	171	194	198	193	193	188	185	191	196	199		
	56	62	154	168	163	163	165	167	168	170	194	196	190	192	188	184	190	195	199		
	68	28	59	112	154	162	162	165	166	169	192	192	188	188	184	183	187	194	201		
	70	28	27	23	106	165	159	162	162	165	187	186	183	178	175	177	180	203	217		
	38	32	32	25	79	161	158	160	160	162	180	176	173	170	168	171	173	182	189		
	32	32	32	27	63	153	156	157	156	162	176	168	165	164	162	165	168	171	175		
	32	32	32	27	64	150	153	153	152	161	174	165	161	156	155	160	162	166	171		
	32	32	32	28	55	144	149	148	147	158	168	160	156	143	147	157	156	159	166		
	32	32	32	30	43	135	145	144	143	154	163	156	150	131	140	153	149	152	160		

Pada pengujian menghitung nilai efisiensi terang cahaya kali ini, jarak yang digunakan sama dengan pengujian jarak benda dengan kamera. Perlu diketahui bahwa cahaya bersifat membuat terang ruangan atau obyek. Ternyata bahwa sifat terang cahaya ini bergantung pada panjang gelombang cahaya tersebut. Sedangkan fluk cahaya adalah energi cahaya yang dibawa suatu cahaya yang diubah menjadi bentuk terang. Tiap-tiap cahaya mempunyai kemampuan sendiri-sendiri yang dinyatakan dalam efisiensi terang. Untuk mendapatkan nilai efisiensi terang dapat menggunakan rumus perhitungan dibawah ini :

685 lumen /watt = Efisiensi terang
 f = frekuensi (Hz)
 V = Laju gelombang (m/s)
 λ = Panjang gelombang (nm)

Sedangkan untuk memperoleh nilai dari laju gelombang (V) adalah :

$$V = s/t$$

s = Jarak antara sumber cahaya ke benda (m).

t = waktu cahaya untuk menempuh sampai ke benda (s).

Keterangan :

- Pada terang maksimum efisiensi terang 685 lumen/watt.
- $f(\lambda)$ d (λ) dianggap 1 karena tidak diketahui frekuensi terhadap panjang gelombangnya.
- V dikeluarkan karena tidak mengandung fungsi panjang gelombang.

Tabel 4.5 Hasil uji coba perhitungan nilai efisensi cahaya

NO	Percobaan	Fluk cahaya	Keterangan
1	Pertama	68,5 lumen	Jarak benda 10 cm
2	Kedua	685 lumen	Jarak benda 1 m
3	Ketiga	1370 lumen	Jarak benda 2 m
4	Keempat	2397.5 lumen	Jarak benda 3.5 m
5	Kelima	3425 lumen	Jarak benda 5 m

Pengujian pertama dengan jarak 10 cm benda yang diuji sama dengan skenario pada pengujian jarak benda :

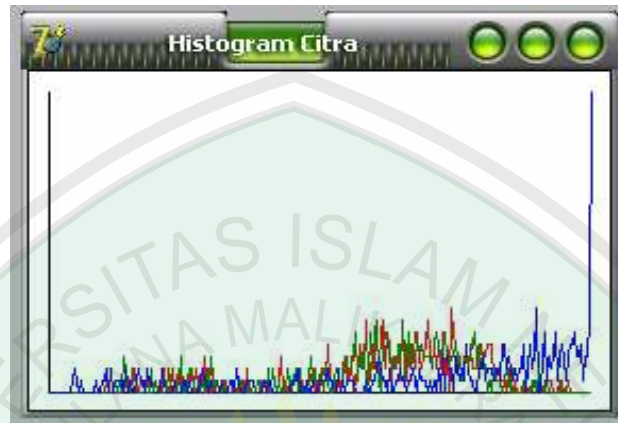
$$F = 685 \int V f(\lambda) d(\lambda)$$

$$F = 685.V \int f(\lambda) d(\lambda)$$

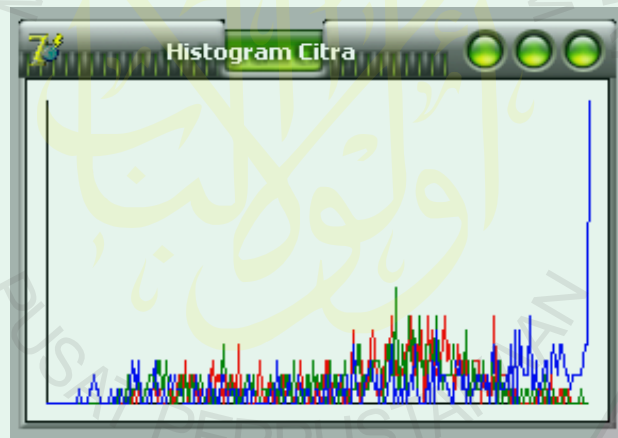
$$= 685.s/t \int 1$$

$$F = 685 . 0,1 = 68,5 \text{ lumen}$$

Jadi pengujian pertama ini luminous fluknya menghasilkan 68.5 lumen, dan menghasilkan gambar tabel histogram dibawah :



Gambar 4.14 Histogram pada gambar 4.7



Gambar 4.15 Histogram pada gambar 4.8

Pengujian kedua dengan jarak 1 m benda yang diuji sama dengan skenario pada pengujian jarak benda :

$$F = 685 \int V f(\lambda) d(\lambda)$$

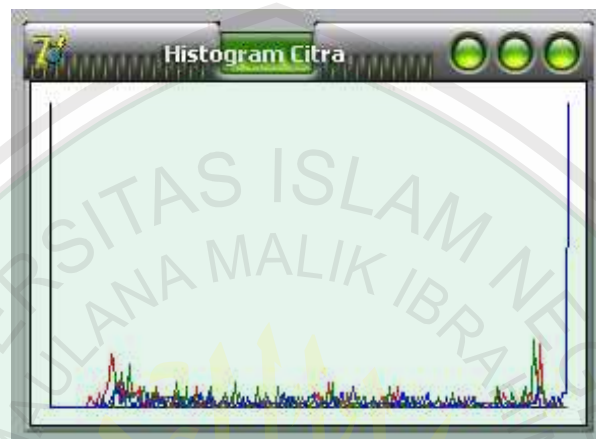
$$F = 685.V \int f(\lambda) d(\lambda)$$

$$= 685.s/t \int 1$$

$$F = 685 .1$$

= 685 lumen

Jadi pengujian kedua ini luminous fluknya menghasilkan 685 lumen, dan menghasilkan gambar tabel histogram dibawah :



Gambar 4.16 Histogram pada gambar 4.9

Pengujian ketiga dengan jarak 2 m benda yang diuji sama dengan skenario pada pengujian jarak benda :

$$F = 685 \int V f(\lambda) d(\lambda)$$

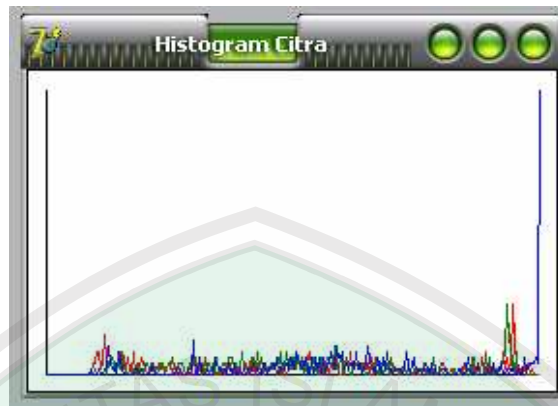
$$F = 685.V \int f(\lambda) d(\lambda)$$

$$= 685.s/t \int 1$$

$$F = 685 .2$$

$$= 1370 \text{ lumen}$$

Jadi pengujian ketiga ini luminous fluknya menghasilkan 1370 lumen, dan menghasilkan gambar tabel histogram dibawah :



Gambar 4.17 Histogram pada gambar 4.10

Pengujian keempat dengan jarak 3.5 m benda yang diuji sama dengan skenario pada pengujian jarak benda :

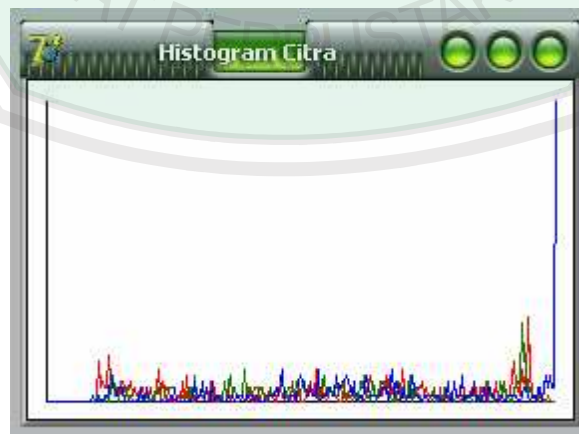
$$F = 685 \int V f(\lambda) d(\lambda)$$

$$F = 685.V \int f(\lambda) d(\lambda)$$

$$= 685.s/t \int 1$$

$$F = 685 .3,5 = 2397,5 \text{ lumen}$$

Jadi pengujian keempat ini luminous fluknya menghasilkan 397,5 lumen, dan menghasilkan gambar tabel histogram dibawah :



Gambar 4.18 Histogram pada gambar 4.11

Pengujian kelima dengan jarak 5 m benda yang diuji sama dengan skenario pada pengujian jarak benda :

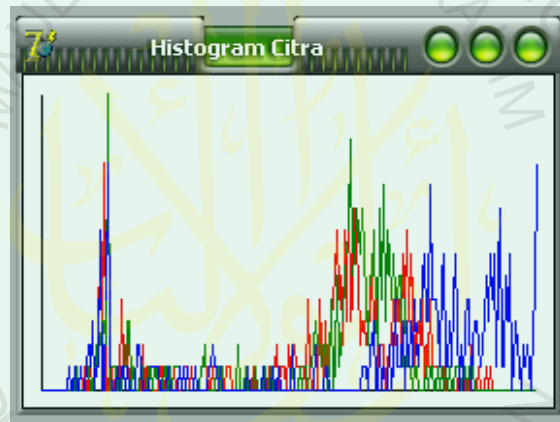
$$F = 685 \int V f(\lambda) d(\lambda)$$

$$F = 685.V \int f(\lambda) d(\lambda)$$

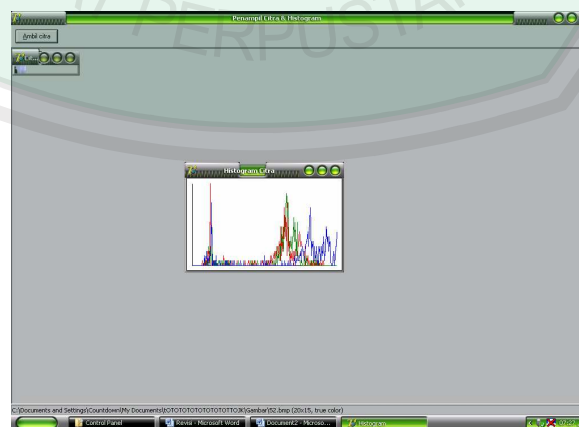
$$= 685.s/t \int 1$$

$$F = 685 . 5 = 3425 \text{ lumen}$$

Jadi pengujian kelima ini luminous fluknya menghasilkan 3425 lumen, dan menghasilkan gambar tabel histogram dibawah :



Gambar 4.19 Histogram pada gambar 4.12



Gambar 4.20 Histogram pada gambar 4.13

Pada bagian deteksi letak obyek rancang bangun alat ini parameter yang digunakan adalah analisis nilai RGB pada koordinat yang ditentukan terhadap citra yang tersampling dengan, parameter nilai RGB lebih besar dari 230. Pendeteksian letak obyek oleh kamera dilakukan dengan cara memanfaatkan gerak motor stepper yang di antarmukakan (diinterfacekan) terhadap PC melalui port procedur.

Pada tabel 4.1 dapat di lihat apabila obyek dimasukkan dari sebelah kiri maka koordinat yang terbaca lebih awal adalah koordinat dengan sumbu x yang memenuhi syarat untuk menggerakkan procedur motor stepper ke kiri yaitu antara koordinat piksel (20,120) hingga (130,120). Begitupun sebaliknya jika obyek dimasukkan dari sebelah kanan maka pembacaan koordinat yang pertama terbaca adalah antara koordinat piksel (190,120) hingga (300,120).

Seperti yang telah dijelaskan sebelumnya jika pada waktu yang sama dalam penyamplingan terdapat nilai yang memenuhi syarat pada koordinat yang berbeda maka pemanggilan procedure stepper yang diaktifkan adalah koordinat yang lebih besar. Suatu misal nilai yang memenuhi syarat pada koordinat piksel (130,120) dan (140,120) maka yang di aktifkan adalah pemanggilan procedure gerak stepper pada koordinat piksel (140,120). Hal ini disebabkan oleh sub rutin program yang dibuat dimulai dari koordinat piksel (20,120) hingga (300,120).

Untuk mendapatkan nilai RGB lebih besar dari 230 maka obyek yang digunakan harus bercahaya. Metode ini berdasarkan dari cara kerja

webcam itu sendiri di mana semakin besar intensitas cahaya yang masuk nilai tegangan akan semakin besar dan hasil tegangan tersebut bila dikonversikan terhadap citra akan menghasilkan nilai RGB yang lebih besar pada citra tersebut. Oleh sebab itu jika obyek cahaya tersebut terletak pada koordinat yang ditentukan maka nilai RGB pada koordinat tersebut akan menghasilkan nilai lebih besar dari 230.

Proses penyamplingan citra dan gerak stepper diatur oleh timer. Interval yang diberikan terhadap proses penyamplingan citra dan gerak stepper berbeda. Pada gerak stepper interval yang diberikan adalah 1ms sedangkan pada penyamplingan citra adalah 50ms. Keadaan ini diharapkan stepper dapat selesai menjalankan perintah yang diberikan sebelum citra berikutnya tersampling.

Dalam rancang bangun alat ini stepper bergerak sebesar $7,2^0$ pada setiap satu kali proses prosedur gerak stepper tersebut dipanggil. Karena dalam prosedur ini input yang diberikan pada stepper adalah 4. Dengan perputaran sudut yang kecil dan interval timer yang lebih cepat dibandingkan dengan penyamplingan citranya, selang waktu penganalisan *software* terhadap koordinat yang telah ditentukan dengan gerak stepper akan seimbang.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

1. Webcam dapat dimanfaatkan sebagai sistem keamanan dengan menggunakan metode deteksi gerakan. Metode ini digunakan untuk mendeteksi adanya perbedaan gerakan pada obyek.
2. Agar motor stepper dapat di perintah melalui PC diperlukan rangkaian driver yang berfungsi untuk menggerakkan motor stepper dengan perintah software tertentu. Pada rangkaian driver terdapat sebuah IC, dimana input dari IC tersebut dihubungkan dengan port paralel yang terdapat pada PC dan output dari IC dihubungkan dengan motor stepper.
3. Webcam dapat mendeteksi letak obyek dengan cara menganalisis nilai RGB pada koordinat yang ditentukan terhadap citra yang tersampling, dengan parameter nilai RGB lebih besar dari 230.
4. Hasil dari pengujian program deteksi gerak ini adalah program dapat berjalan sebagaimana mestinya akan tetapi program ini kurang maksimal apabila obyek yang ditangkap jarak antara kamera dengan obyek tersebut terlalu jauh, yakni motor stepper tidak dapat bergerak sebagaimana mestinya, hal ini dikarenakan intensitas cahaya yang ditangkap oleh kamera kurang maksimal, karena prosedur untuk menggerakkan motor stepper menggunakan nilai RGB.

B. Saran

Untuk penelitian berikutnya penulis menyarankan adanya perbaikan-perbaikan sebagai berikut:

1. Obyek yang digunakan tidak hanya terbatas pada obyek bercahaya saja.
2. Agar dapat menggunakan obyek yang tidak bercahaya dibutuhkan kamera dengan kepekaan dan resolusi yang lebih tinggi dari kamera yang digunakan dalam penelitian ini.
3. Motor stepper yang digunakan dapat berputar 180^0 , dan menggunakan dua buah motor stepper agar gerak stepper bergerak tidak hanya horizontal saja melainkan dapat bergerak secara vertikal.

DAFTAR PUSTAKA

- Achmad, balza dan firdausy kartika. 2005. *Teknik Pengolahan Citra Digital Menggunakan Delphi*. Yogyakarta: Ardi Publishing.
- Ahmad Usman, 2005, *Pengolahan Citra Digital dan Teknik Pemrogramannya*, Yogyakarta: Graha Ilmu.
- Abdullah Amin, 2004, *Integrasi Sains-Islam: Mempertemukan Epistemologi Islam dan Sains*, Yogyakarta: Pilar Religia.
- Gunawan feri, 2005. *Membuat Sistem Pendeteksi Ruangan Dengan Sinar Laser*. Jakarta: Elex Media Komputindo.
- Husni, 2004. *Pemrograman Database Dengan Delphi*. Yogyakarta: Graha Ilmu.
- Kosasih, Asep. 2006. *Pendekatan praktis belajar algoritma & pemrograman dengan bahasa delphi 5.0*. Bandung: Yrama Widya.
- Madcom, 2003. *seri panduan pemrograman borland delphi 7*. Yogyakarta: Andi Publishing.
- Nugroho, Dedi Tri Putra Adi. 2006. *Cara Mudah Belajar Delphi*. Semarang: Neomedia Press.
- Pranata Antony, 2003, *Pemrograman Borland Delphi 6 Edisi 4*. Yogyakarta: Andi
- Sutanata, Edhy. 2004, *Algoritma Teknik Penyelesaian Permasalahan Untuk Komputasi*. Yogyakarta: Graha Ilmu.
- Suyanto, M. 2003, *Multimedia Alat Untuk Meningkatkan keunggulan bersaing*. Yogyakarta: Andi.
- Tim Penelitian Dan Pengembangan. 2006. *Teknik Antarmuka Mikrokontroler Dengan Komputer Berbasis Delphi*. Jakarta: Salemba Infotek.
- Widodo Romi Budi dan Irawan Joseph Dedy. 2007. *Interfacing Paralel Dan Serial Menggunakan Delphi*. Yogyakarta: Graha Ilmu.
- Brian marshall, 2005. *how webcam work*. [http:// www.howstuffwork.com/](http://www.howstuffwork.com/) Akses 21 April 2008
- , 2005, *Port USB*, [http:// www.pctechguide.com/](http://www.pctechguide.com/) Akses 20 Maret 2008