

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA : Muhammad Redha
NRP : 5110100703
DOSEN WALI : Dr.Eng. Nanik Suciati, S.Kom., M.Kom.
DOSEN PEMBIMBING : 1. Dwi Sunaryono, S.Kom., M.Kom.
2. Ridho Rahman Hariadi, S.Kom.

2. JUDUL TUGAS AKHIR

“Implementasi Sistem Pendeteksi Gerakan dengan *Motion Detection* pada Kamera Video Menggunakan AForge .NET.”

3. LATAR BELAKANG

Pada saat ini keamanan adalah salah satu hal yang sangat penting. Dengan kemajuan teknologi dalam bidang keamanan sekarang ini, pengamatan suatu objek menjadi lebih praktis. Untuk mengamati suatu objek tidak perlu dilakukan pengamatan secara langsung dan terus menerus. Cukup meletakkan suatu kamera yang mengarah pada objek yang diinginkan dan objek tersebut dapat diamati pada layar monitor. Permasalahan yang timbul selama ini kamera hanya dapat menangkap suatu objek tetapi tidak dapat memberikan informasi tentang pergerakan objek tersebut. Hal ini sangat berguna jika diaplikasikan ke dalam sistem pendeteksi gerakan dengan *motion detection*. Ketika ada objek yang bergerak, maka secara otomatis aplikasi akan memberikan informasi objek asing yang terdeteksi tersebut.

Sistem *motion detection* merupakan suatu sistem yang digunakan untuk menganalisis gerakan pada video pengawasan untuk memudahkan pendeteksian gerakan dari suatu kejadian penting. Sistem *motion detection* yang akan dibangun harus dapat mengidentifikasi adanya perubahan dari kondisi awal yang sudah didefinisikan. Hal

ini sangat penting, karena apabila terjadi perubahan, maka aplikasi harus bisa mengkonfirmasi atau memberi informasi perubahan tersebut melalui informasi deteksi. Dengan demikian dapat diketahui jika ada objek yang bergerak. Hasil dari informasi pendeteksian gambar atau objek bergerak tersebut dapat digunakan sebagai media konkrit untuk melihat adanya perubahan.

Pemrograman dilakukan dengan menggunakan bahasa pemrograman C# dan *library framework* AForge .NET pada Microsoft Visual Studio 2010. AForge .NET adalah sebuah *framework* yang dirancang khusus untuk memberikan filter pemrosesan gambar pada bahasa pemrograman C#. Perancangan dimulai dengan membuat program pendeteksi gerakan. Setelah program dibuat, percobaan dilakukan terhadap aplikasi agar mendapatkan hasil yang optimal.

Hasil yang diharapkan semoga dengan adanya sistem ini nantinya dapat memberikan manfaat untuk memudahkan pendeteksian gerakan dari suatu kejadian penting pada video pengawas untuk meningkatkan sistem keamanan menjadi lebih baik.

4. RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana mendeteksi adanya gerakan pada video.
2. Bagaimana suatu sistem bisa memberikan informasi adanya gerakan pada sebuah video.
3. Bagaimana suatu sistem bisa menganalisis pengecualian perubahan gerakan kecil yang terjadi secara alami.
4. Bagaimana membangun aplikasi yang bisa memilih sebagian area yang memerlukan pendeteksian dari keseluruhan kawasan deteksi.

5. BATASAN MASALAH

Adapun batasan ruang lingkup permasalahan dari pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. Mengintegrasikan kamera video dengan perangkat lunak pendeteksi gerakan.
2. Wilayah deteksi hanya terbatas pada suatu area yang tidak terlalu luas (sebuah ruangan) dan kondisi ruangan yang relatif tetap.
3. Kamera hanya memantau perubahan yang terjadi pada area yang terbatas tanpa mendefinisikan perubahan tersebut. Bila ada objek baru yang bergerak, maka pendeteksi akan aktif.
4. Bahasa pemrograman yang digunakan adalah C# dengan *framework* .NET.

6. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari pembuatan Tugas Akhir ini adalah sebagai berikut:

1. Membangun sebuah sistem pendeteksi gerakan dengan *motion detection* pada kamera video.
2. Mengimplementasikan penerapan *framework* AForge .NET pada sistem pendeteksi gerakan.
3. Menganalisis pendeteksian gerakan pada video.

7. MANFAAT TUGAS AKHIR

Tugas akhir ini dibuat dengan harapan dapat memberikan manfaat di bidang informatika dalam penerapan sistem *motion detection* untuk pendeteksian gerakan dari suatu kejadian penting pada video pengawas agar meningkatkan sistem keamanan menjadi lebih baik.

8. TINJAUAN PUSTAKA

a. *Motion Detection*

Motion Detection adalah proses mendeteksi perubahan posisi dari suatu objek relatif terhadap sekitarnya atau perubahan lingkungan relatif terhadap suatu objek. *Motion Detection* dapat dicapai oleh kedua metode mekanik dan elektronik [1].

b. *Framework .NET*

Framework .NET merupakan sebuah perangkat lunak kerangka kerja yang berjalan utamanya pada sistem operasi Microsoft Windows. Saat ini *framework .NET* umumnya telah terintegrasi dalam distribusi standar Windows (mulai dari Windows Server 2003 dan versi-versi Windows yang lebih baru). Kerangka kerja ini menyediakan sejumlah besar pustaka pemrograman komputer dan mendukung beberapa bahasa pemrograman serta interoperabilitas yang baik sehingga memungkinkan bahasa-bahasa tersebut berfungsi satu dengan lain dalam pengembangan sistem. Perangkat lunak ini adalah kunci penawaran utama dari Microsoft dan dimaksudkan untuk digunakan oleh sebagian besar aplikasi-aplikasi baru yang dibuat untuk *platform* Windows [2].

Framework .NET dapat memenuhi beberapa tujuan berikut:

1. Menyediakan lingkungan pemrograman berorientasi objek, apakah kode objek disimpan dan dijalankan secara lokal, dijalankan secara lokal tetapi disebarluaskan melalui internet atau dijalankan secara *remote* (dijalankan dari suatu tempat).
2. Menyediakan lingkungan untuk menjalankan suatu kode yang meminimalkan konflik saat *software deployment* disebarluaskan dan *versioning* atau tentang versi.
3. Menyediakan lingkungan untuk menjalankan suatu kode yang menjamin keamanan saat kode dijalankan, termasuk kode yang dibuat oleh pihak yang tidak diketahui atau pihak ketiga yang setengah dipercaya.

4. Menyediakan lingkungan untuk menjalankan suatu kode yang dapat mengeliminasi masalah performa dari lingkungan *scripted* dan *interpreted*.
5. Membuat pengembang memiliki pengalaman yang konsisten dalam berbagai tipe aplikasi berbasis Windows dan aplikasi berbasis *web*.

Sebagai salah satu sarana untuk dapat memenuhi tujuan yang telah dipaparkan, maka dibuatlah berbagai macam bahasa pemrograman yang dapat digunakan dan dapat berjalan di atas *platform framework* seperti bahasa C#, VB.NET, C++, J#, Perl.NET dan lain-lain. Masing-masing bahasa tersebut mempunyai kelebihan dan kekurangannya masing-masing, namun yang pasti apapun bahasa pemrograman yang digunakan semua dapat saling berkomunikasi dan saling *compatible* satu sama lainnya dengan bantuan *framework .NET* [3].

c. AForge Framework

AForge Framework merupakan *framework open source* C# yang dirancang bagi para pengembang dan peneliti di bidang *computer vision* dan *artificial intelligence* yang meliputi pengolahan citra, jaringan saraf tiruan, algoritma genetika, logika *fuzzy*, *machine learning* dan robotika.

Framework ini terdiri dari beberapa *library* dan contoh aplikasi yang ditunjukkan oleh fitur-fitur sebagai berikut:

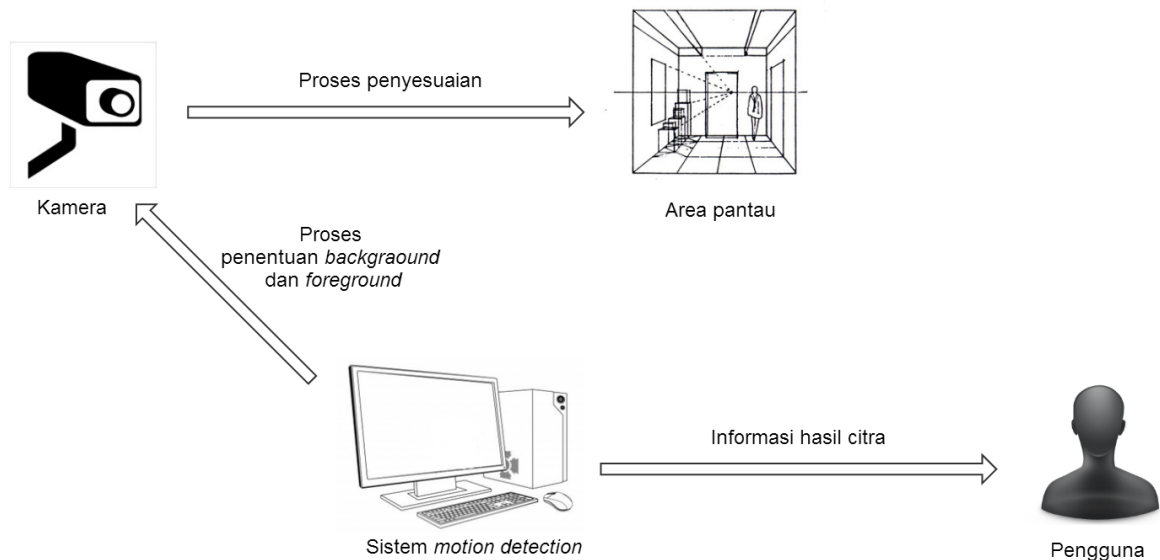
1. *Aforge.Imaging, library* dengan *image processing* dan *filtering*, untuk mem-*filter* gambar yang akan diproses.
2. *Aforge.Vision, computer vision library, library* untuk proses *motion detection* yang sederhana, membedakan nilai *threshold* dan perhitungan perbedaan piksel.
3. *Aforge.Neuro, neural network computation library*, untuk membuat *arsitektur neural network* (jaringan saraf tiruan).
4. *Aforge.Genetic, evolution programing library*, untuk ilmu komputasi.
5. *Afore.MachineLearning, machine learning library*, mesin pembelajaran.
6. *Aforge.Robotics, library providing support of some robotics kits*, memanipulasi perbedaan Lego Mindstom peralatan robotic, mendukung peralatan Lego Mindstrom RCX dan Lego Mindstrom NXT.
7. *Aforge.Video, set of libraries for video processing*, untuk video yang terintegrasi dengan Windows [4].

d. Pengertian C#

C# (dibaca “*See-sharp*”) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka *framework .NET*. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain dengan beberapa penyederhanaan. Menurut standar ECMA-334 *C# Language Specification*, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan simbol # (U+0023) [5].

9. RINGKASAN ISI TUGAS AKHIR

Motion detection melakukan pendekatan dengan membandingkan *frame* pada saat ini dengan *frame* sebelumnya. Mula-mula kamera video akan menangkap gambar dari ruangan yang sedang dipantau. Kemudian membandingkan warna yang terdapat pada *frame* saat ini dengan *frame* sebelumnya. Apabila terdapat perbedaan warna, maka objek tersebut terdeteksi sebagai gerakan. Sebuah fungsi akan dipanggil untuk proses pemberian informasi bahwa gerakan terdeteksi. Alur aplikasi dari sistem *motion detection* ditunjukkan pada Gambar 1.



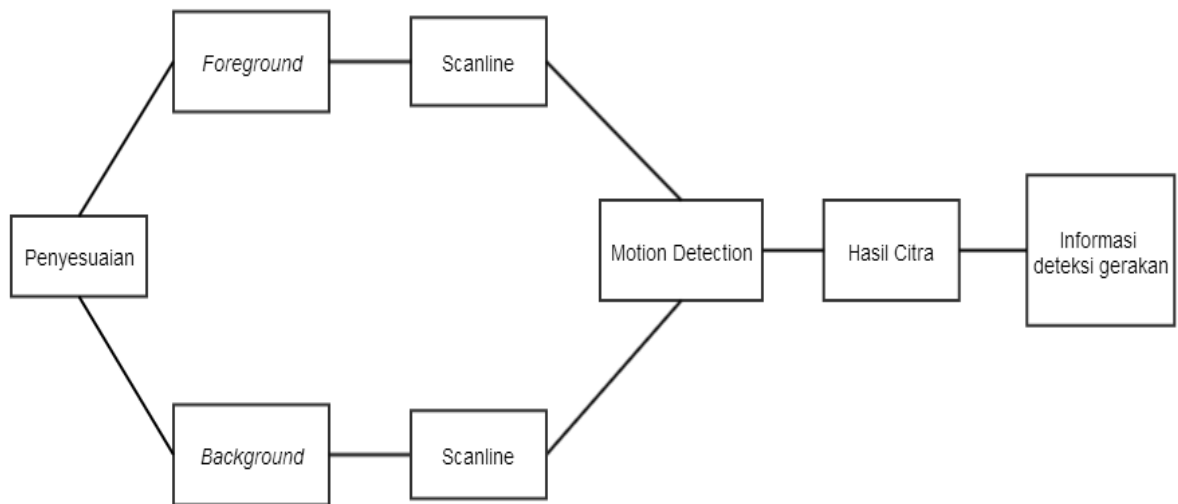
Gambar 1. Alur aplikasi

Pendekatan yang dilakukan adalah memisahkan gambar antara area *background* dengan area *foreground* pada area gerak untuk melacak adanya pergerakan. Asumsi dasar bahwa area *background* selalu tetap. Ini memungkinkan untuk mencapai segmentasi citra yang akurat. Dengan asumsi ini, dapat dilakukan penghitungan objek dari waktu ke waktu. Untuk setiap *frame* dari video, kita dapat mengurangi gambar *background* dari *frame* tersebut. Jumlah piksel dengan hasil mendekati nol diasumsikan sebagai *background* dan jumlah piksel yang lebih besar adalah *foreground* (objek). Meskipun pendekatan ini terlihat sederhana namun, pada kenyataannya lebih sulit karena *background* selalu berubah dari waktu ke waktu. Pencahayaan dapat berubah secara halus. Pendekatan yang lebih baik adalah pendekatan yang dapat beradaptasi dengan *background* secara perlahan [6].

Pada Tugas Akhir ini dibuat suatu sistem pendeteksian gerakan yang dapat beradaptasi terhadap perubahan gerakan kecil yang terjadi secara perlahan. Akan dibangun sebuah aplikasi yang dapat terintegrasi langsung terhadap kamera untuk mendeteksi perubahan gerakan pada video. Pemilihan atau penentuan area deteksi sangat penting dilakukan untuk menemukan area yang memerlukan pengamanan maksimum dan area yang hanya memerlukan pemantauan saja. Pengguna aplikasi bisa memilih sebagian area yang memerlukan pendeteksian saja dari keseluruhan area

pantau. Area yang memerlukan pengamanan maksimum hanya pada area dimana pendeteksian yang diperlukan tersebut berada, untuk area lainnya dapat diabaikan. Pemilihan area deteksi ini bertujuan untuk mengoptimalkan konfigurasi hasil citra deteksi sehingga sistem bisa beroperasi lebih baik.

Pendekatan *motion detection* dilakukan dengan mengidentifikasi adanya suatu gerakan dengan cara melakukan proses pengurangan nilai- nilai intensitas setiap piksel yang ada pada *background* terhadap nilai-nilai intensitas pada suatu *foreground* (objek) yang diambil secara kontinyu. Secara garis besar, gambaran proses pada sistem ini ditunjukkan pada Gambar 2.



Gambar 2. Gambaran umum sistem *motion detection*.

Berikut ini penjelasan poin-poin proses yang terdapat pada Gambar 2.

1. **Proses Penyesuaian**
Pada proses penyesuaian ini, cahaya yang ditangkap kamera video sampai pada kondisi normal, yaitu sebuah kondisi dimana tidak akan terjadi perubahan nilai secara signifikan dari intensitas cahaya pada posisi kamera. Dan kondisi yang tetap pada area yang dipantau.
2. **Proses Penentuan *Background***
Pada proses ini *background* yang didapat dari area yang dipantau nantinya akan di-*scanline*, yaitu proses pengambilan atau pencarian nilai intensitas yang ada pada suatu gambar (untuk setiap pikselnya) yang memiliki format *bitmap*.
3. **Proses Pengambilan *foreground* (objek)**
Setelah penentuan *background* langkah selanjutnya adalah pengambilan gambar yang dilakukan secara kontinyu persatuan waktu. Gambar yang diperoleh nantinya akan didefinisikan sebagai *foreground* (objek) seperti yang terlihat pada gambaran umum sistem. Seperti halnya pada *background*, *foreground* (objek) ini pun nantinya akan diproses untuk mendapatkan nilai intensitas dari masing-masing piksel.

4. *Proses Motion Detection*

Pada proses *motion detection*, *background* dan *foreground* (objek) akan di-*scanline*. Angka-angka yang diperoleh merepresentasikan tingkat intensitas cahaya yang ada pada *foreground* (objek). Nilai yang diperoleh nantinya akan disubstitusikan ke dalam sebuah persamaan untuk menentukan adanya gerakan dari objek yang sebelumnya sudah ada atau objek baru dari area yang dipantau. Setelah gerakan terdeteksi, maka sistem akan memberikan informasi pendeteksian gerakan yang terjadi.

10. METODOLOGI

a. Penyusunan proposal tugas akhir

Proposal Tugas Akhir ditulis untuk mengajukan ide atas pengerjaan Tugas Akhir. Proposal ini juga mengandung proyeksi dari ide Tugas Akhir yang diajukan. Proposal yang dibuat mengusulkan tentang sistem pendeteksi gerakan dengan *motion detection*.

b. Studi literatur

Pada tahap ini diperlukan adanya pengumpulan informasi dari beberapa sumber terkait dengan *motion detection* dan literatur yang diperlukan dalam proses perancangan dan implementasi sistem yang akan dibangun. Literatur yang digunakan adalah terkait dengan penerapan *motion detection* pada kamera video dan implementasi menggunakan AForge .NET.

c. Analisis dan desain perangkat lunak

Pada tahap ini akan dilakukan analisa awal dan dilakukan pendefinisian dari kebutuhan sistem yang dibangun untuk mengetahui masalah yang sedang dihadapi. Dari proses tersebut maka akan dilakukan perumusan rancangan sistem yang dapat memberikan solusi dari masalah yang dihadapi.

d. Implementasi perangkat lunak

Pada tahap ini dilakukan pembuatan perangkat lunak yang merupakan implementasi dari rancangan yang telah dibuat sebelumnya. Rencana pembuatan perangkat lunak ini akan diimplementasikan dengan menggunakan:

1. Bahasa pemrograman yang digunakan adalah C#.
2. *Framework* yang digunakan adalah AForge .NET.
3. *Tools* pemrograman yang digunakan adalah Visual Studio 2010.
4. *Tools* pemodelan yang digunakan adalah Microsoft Office Visio 2007.

e. Pengujian dan evaluasi

Pada tahap ini akan dilakukan pengujian terhadap perangkat lunak yang dibangun dengan melakukan berbagai gerakan yang akan dideteksi untuk mencoba jalannya aplikasi apakah telah sesuai dengan rancangan dan desain implementasi yang dibuat. Uji coba dan evaluasi perangkat lunak ini digunakan untuk mencari kesalahan program yang timbul, mengevaluasi jalannya program, dan melakukan perbaikan apabila terjadi adanya kesalahan program.

f. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
 - 1.1 Latar Belakang
 - 1.2 Permasalahan
 - 1.3 Batasan Tugas Akhir
 - 1.4 Tujuan
 - 1.5 Metodologi
 - 1.6 Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Uji Coba dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

11. JADWAL KEGIATAN

Jadwal kegiatan pelaksanaan Tugas Akhir dapat dilihat pada Tabel 1.

Tabel 1. Jadwal kegiatan.

Tahapan	2014															
	Februari				Maret				April				Mei			
Penyusunan Proposal																
Studi Literatur																
Perancangan sistem																

